

# **Optimización de Estrategias de Control Difusas y Convencionales para Procesos MIMO Mediante el Uso del Recocido Simulado**

por

**Ing. Rubén Anton Aguilar Rivera**

**Tesis**

Presentada al Programa de Graduados en Ingeniería  
como requisito parcial para obtener el grado académico de

**Maestro en Ciencias**

especialidad en

**Automatización**

**Instituto Tecnológico y de Estudios Superiores de Monterrey**

**Campus Monterrey**

Mayo de 2006

# Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Monterrey

División de Ingeniería y Arquitectura

Programa de Graduados en Ingeniería

Los miembros del comité de tesis recomendamos que la presente tesis de Rubén Anton Aguilar Rivera sea aceptada como requisito parcial para obtener el grado académico de **Maestro en Ciencias**, especialidad en:

**Automatización**

**Comité de tesis:**

---

Dr. José de Jesús Rodríguez Ortiz

Asesor de la tesis

---

Dr. Horacio Martínez Alfaro

Sinodal

---

Dr. Rogelio Soto Rodríguez

Sinodal

---

Dr. Federico Viramontes Brown

Director del Programa de Graduados

Mayo de 2006

A Dios, que todo me ha dado,  
a mis padres, por su amor,  
e inevitable,  
inconfundible  
e indiscutiblemente,  
a la musa de mi inspiración...

## Reconocimientos

Agradezco al Departamento de Mecatrónica y Automatización del Tecnológico de Monterrey, Campus Monterrey, por el apoyo que me brindó al permitirme entrar al programa de becas laborando como asistente de docencia en sus instalaciones.

Agradezco al Dr. José de Jesús Rodríguez Ortiz, asesor de esta tesis, por el consejo y guía que me ofreció durante el tiempo que duró el desarrollo del presente trabajo.

Agradezco al Dr. Horacio Martínez Alfaro y al Dr. Rogelio Soto Rodríguez por desempeñar su papel como sinodales en el comité de tesis y los consejos y correcciones que emitieron con la intención de perfeccionar el documento presentado.

Agradezco a todos los estudiantes de la maestría, a los que están y los que estuvieron, por el apoyo que me brindaron como compañeros de trabajo, como compañeros de equipo y como amigos durante el periodo de posgrado. De igual manera agradezco al resto de las personas con las que tuve la oportunidad de compartir diferentes experiencias a lo largo de esta etapa.

Finalmente, agradezco al lector que se ha tomado el tiempo para revisar este documento, con el deseo que su contenido pueda servirle a él y a otros que su camino los ha conducido hasta aquí.

RUBÉN ANTON AGUILAR RIVERA

*Instituto Tecnológico y de Estudios Superiores de Monterrey*  
*Mayo 2006*

# Optimización de Estrategias de Control Difusas y Convencionales para Procesos MIMO Mediante el Uso del Recocido Simulado

Rubén Anton Aguilar Rivera, M.C.  
Instituto Tecnológico y de Estudios Superiores de Monterrey, 2006

Asesor de la tesis: Dr. José de Jesús Rodríguez Ortiz

El presente trabajo hace investigación en el campo del control de procesos MIMO haciendo uso de esquemas basados en sistemas difusos. Para ello se hace un compendio de la lógica difusa que se aterriza en el diseño de diferentes estructuras de controladores basados en esta técnica. Se proponen tres estructuras principales: Un controlador difuso multivariable que se encarga tanto de la manipulación como del acoplamiento, una estrategia basada en controladores univariados apoyados por un bloque desacoplador convencional y la estrategia basada en una estructura PID difusa apoyada también en bloques desacopladores. A su vez, se proponen lineamientos para el diseño de los controladores difusos MIMO.

Se estudia la posibilidad de utilizar el recocido simulado como una técnica que permite el ajuste de los controladores sin la necesidad de amplio conocimiento del proceso o del efecto de la variación de los parámetros del controlador. Se justifica su aplicación y se describe la manera en que logró aplicar para las estrategias mencionadas. Se describe como este procedimiento puede aplicarse también a controladores convencionales como es el PID.

Se hace una comparación de desempeño frente a la estrategia convencional de PID con desacoplador. Se describe la naturaleza de estas pruebas para evaluar el máximo rendimiento que puede ofrecer el controlador. A su vez, también se presenta el esquema de pruebas necesario para evaluar la robustez de los sistemas, entendiéndola en este caso como insensibilidad ante el cambio en los parámetros del modelo del proceso.

Tras la presentación y análisis de los resultados obtenidos, se presentan las conclusiones alcanzadas y las recomendaciones pertinentes.

# Índice general

<b>Reconocimientos</b>	<b>v</b>
<b>Resumen</b>	<b>vi</b>
<b>Índice de tablas</b>	<b>x</b>
<b>Índice de figuras</b>	<b>xiv</b>
<b>Capítulo 1. Introducción</b>	<b>1</b>
1.1. Establecimiento del problema . . . . .	1
1.2. Juitificación . . . . .	1
1.3. Objetivos . . . . .	2
1.4. Metodología . . . . .	3
1.5. Descripción del documento de tesis . . . . .	3
<b>Capítulo 2. Antecedentes</b>	<b>5</b>
2.1. Breve resumen del estado del arte . . . . .	5
2.2. Línea de investigación . . . . .	7
<b>Capítulo 3. Estructura del controlador difuso</b>	<b>9</b>
3.1. Introducción . . . . .	9
3.2. Lógica Difusa . . . . .	10
3.2.1. Definiciones básicas . . . . .	10
3.2.2. Propiedades de los conjuntos difusos . . . . .	10
3.2.3. Operaciones básicas con conjuntos difusos . . . . .	12
3.3. Descripción del controlador difuso . . . . .	14
3.3.1. Estructura general del controlador difuso . . . . .	14
3.3.2. Normalización . . . . .	15
3.3.3. Fuzificador . . . . .	15
3.3.4. Conjuntos difusos de entrada . . . . .	16
3.3.5. Ganancias de Normalización . . . . .	18
3.3.6. Conjuntos difusos de salida . . . . .	20

3.3.7. Máquina de inferencia . . . . .	24
3.3.8. Base de reglas . . . . .	26
3.3.9. Defuzificador . . . . .	30
3.4. Otras estructuras de controlador difuso . . . . .	31
3.4.1. Controlador difuso univariable de 9 reglas . . . . .	31
3.4.2. Controladores difusos univariables para emulación de PID . . . . .	34
3.5. Conclusión . . . . .	37
<b>Capítulo 4. Estrategia de Sintonización</b>	<b>41</b>
4.1. El problema de optimización . . . . .	41
4.2. Recocido Simulado . . . . .	42
4.2.1. Aspectos prácticos del algoritmo . . . . .	44
4.2.2. Algoritmo de sintonización . . . . .	49
4.2.3. Ejemplo de aplicación del algoritmo . . . . .	51
4.2.4. Ejemplo Comparativo . . . . .	52
4.3. Conclusión . . . . .	53
<b>Capítulo 5. Estrategia de comparación</b>	<b>60</b>
5.1. Introducción . . . . .	60
5.2. Descripción de la estructura de PID con desacoplador . . . . .	60
5.2.1. Ganancias Relativas . . . . .	61
5.2.2. Diseño del desacoplador . . . . .	62
5.3. Sintonización de la estrategia de control . . . . .	63
5.3.1. Sintonía obtenida mediante programación genética . . . . .	64
5.3.2. Ejemplo de diseño de PID con desacoplador . . . . .	65
5.4. Mejoramiento del desempeño del controlador mediante el uso del recocido simulado . . . . .	66
5.5. Conclusión . . . . .	67
<b>Capítulo 6. Presentación y análisis de resultados</b>	<b>69</b>
6.1. Introducción . . . . .	69
6.2. Diseño de las pruebas . . . . .	69
6.3. Resumen de sintonizaciones utilizadas . . . . .	72
6.4. Análisis de los resultados . . . . .	78
<b>Capítulo 7. Conclusiones y recomendaciones</b>	<b>85</b>
7.1. Conclusiones . . . . .	85
7.2. Recomendaciones para investigaciones futuras . . . . .	87
<b>Bibliografía</b>	<b>89</b>

<b>Apéndice A. Tablas de resultados en pruebas de desempeño</b>	<b>92</b>
A.1. Presentación de resultados. Caso de dos signos distintos . . . . .	92
A.2. Presentación de resultados. Caso de un signo distinto . . . . .	98
<b>Apéndice B. Gráficas de pruebas de desempeño</b>	<b>103</b>
<b>Apéndice C. Interfaz Humano-Máquina y código programado para control multivariable</b>	<b>109</b>
C.1. Introducción . . . . .	109
C.2. Interfaz gráfica con el usuario . . . . .	109
C.2.1. Pantalla principal . . . . .	110
C.2.2. Pantalla de proceso . . . . .	111
C.2.3. Pantalla de evaluación de respuesta del sistema . . . . .	112
C.2.4. Pantallas de configuración de controlador . . . . .	114
C.3. Código del programa . . . . .	117
C.3.1. Código del lazo de control . . . . .	118
C.3.2. Código en Matlab para seleccionar reglas de controlador difuso multivariable . . . . .	125
<b>Vita</b>	<b>132</b>



## Índice de tablas

3.1. Pares de salida mF y mC para caso de 1 signo distinto. . . . .	22
3.2. Pares de manipulaciones permitidos. . . . .	23
3.3. Pares de salidas mF y mC para 2 signos. . . . .	23
3.4. Pares de manipulaciones permitidos . . . . .	24
3.5. Pares de salida mF y mC para caso sin signos . . . . .	25
3.6. Pares de manipulaciones permitidos, caso de todos los signos positivos .	25
3.7. Valores atribuidos a las reglas difusas . . . . .	27
3.8. Posiciones iniciales de conjuntos difusos de entrada de controlador difuso univariable. . . . .	32
3.9. Posiciones iniciales de conjuntos difusos de salida del controlador difuso univariable. . . . .	33
3.10. Ley de control para procesos univariables . . . . .	33
3.11. Base de reglas de controlador univariable . . . . .	33
3.12. Configuración inicial de PID difuso. Posiciones de centros de entradas y salidas. . . . .	37
4.1. Ganancias de entrada y salida del controlador difuso . . . . .	52
4.2. Parámetros de desempeño del controlador difuso . . . . .	52
4.3. Función de costo para las pruebas realizadas en ambas sintonías de controlador de 16 reglas. . . . .	53
4.4. Configuración de controladores difusos de 16 reglas. Sintonía original y la obtenida mediante algoritmo de sintonización. . . . .	53
4.5. Configuración de controladores difusos de 16 reglas. Conjuntos difusos de salida. . . . .	53
5.1. Sintonía de estrategia PID con desacoplador . . . . .	66
5.2. Comparación de sintonía obtenida con programación genética y la mejorada por el recocido simulado . . . . .	66
6.1. Sintonización de controlador difuso multivariable. Caso 1 signo. Conjuntos difusos de entrada. . . . .	72

6.2. Sintonización de Controlador difuso multivariable. Caso 2 signos. Conjuntos difusos de entrada. . . . .	73
6.3. Sintonización de controlador difuso multivariable. Caso 1 signo. Conjuntos difusos de salida. . . . .	73
6.4. Sintonización de Controlador difuso multivariable. Caso 2 signos. Conjuntos difusos de salida. . . . .	73
6.5. Sintonización de controlador difuso multivariable. Caso 1 signo. Ganancias . . . . .	74
6.6. Sintonización de Controlador difuso multivariable. Caso 2 signos. Ganancias. . . . .	74
6.7. Sintonización de Controladores difusos univariables . Caso 1 signo. Conjuntos difusos de entrada . . . . .	74
6.8. Sintonización de controlador difuso univariable. Caso 1 signo. Conjuntos difusos de salida. . . . .	75
6.9. Sintonización de controlador difuso univariable. Caso 1 signo. Ganancias. . . . .	75
6.10. Sintonización de Controladores difusos univariables . Caso 2 signos. Conjuntos difusos de entrada. . . . .	75
6.11. Sintonización de Controladores difusos univariables . Caso 2 signos. Conjuntos difusos de salida. . . . .	76
6.12. Sintonización de Controladores difusos univariables . Caso 2 signos. Ganancias. . . . .	76
6.13. Sintonización de PID (RS) . Caso 1 signo. . . . .	76
6.14. Sintonización de Controlador PID. Caso 2 signos. Ganancias. . . . .	77
6.15. Sintonización de PID de 3 difusos. Caso 1 signo. . . . .	77
6.16. Sintonización de PID de 3 difusos . Caso 2 signos. . . . .	77
6.17. Resultados de Pruebas de robustez. Función de costo para caso de 2 signos. . . . .	78
6.18. Resultados de Pruebas de robustez. Función de costo para caso de 2 signos (Continuación) . . . . .	79
6.19. Resultados de Pruebas de robustez. Función de costo para caso de 1 signo . . . . .	79
6.20. Resultados de Pruebas de robustez. Función de costo para caso de 1 signo (Continuación) . . . . .	79
A.1. Prueba de Desempeño. Indicadores de desempeño en condiciones ideales (Lazo1) . . . . .	92
A.2. Prueba de Desempeño. Indicadores de desempeño en condiciones ideales (Lazo2). . . . .	93
A.3. Prueba de Robustez. Aumento de 50 % en las ganancias del proceso. (Lazo1). . . . .	93
A.4. Prueba de Robustez. Aumento de 50 % en las ganancias del proceso. (Lazo2). . . . .	93

A.5. Prueba de Robustez. Disminución de 50 % en las ganancias del proceso. (Lazo1). . . . .	94
A.6. Prueba de Robustez. Disminución de 50 % en las ganancias del proceso (Lazo2). . . . .	94
A.7. Prueba de Robustez. Aumento de 50 % en las constantes de tiempo. (Lazo1). . . . .	94
A.8. Prueba de Robustez. Aumento de 50 % en las constantes de tiempo (Lazo2).	94
A.9. Prueba de Robustez. Disminución de 50 % en las constantes de tiempo (Lazo1). . . . .	95
A.10. Prueba de Robustez. Disminución de 50 % en las constantes de tiempo (Lazo2). . . . .	95
A.11. Prueba de Robustez. Aumento de 50 % en el tiempo muerto. (Lazo1). .	95
A.12. Prueba de Robustez. Aumento de 50 % en el tiempo muerto. (Lazo2). .	95
A.13. Prueba de Robustez. Disminución de 50 % en el tiempo muerto. (Lazo1).	96
A.14. Prueba de Robustez. Disminución de 50 % en el tiempo muerto. (Lazo2).	96
A.15. Prueba de Robustez. Aumento de 50 % en los parámetros. (Lazo1). . .	96
A.16. Prueba de Robustez. Aumento de 50 % en los parámetros. (Lazo2). . .	96
A.17. Prueba de Robustez. Disminución de 50 % en los parámetros. (Lazo1). .	97
A.18. Prueba de Robustez. Disminución de 50 % en los parámetros. (Lazo2). .	97
A.19. Prueba de Robustez. El peor caso. (Lazo1). . . . .	97
A.20. Prueba de Robustez. El peor caso. (Lazo2). . . . .	98
A.21. Pruebas En condiciones ideales. 1 signo. (Lazo1). . . . .	98
A.22. Pruebas En condiciones ideales. (1 signo, Lazo2). . . . .	98
A.23. Pruebas de Robustez. Aumento de 50 % en las ganancias del proceso. (1 signo, Lazo1). . . . .	99
A.24. Pruebas de Robustez. Aumento de 50 % en las ganancias del proceso. (1 signo, Lazo2). . . . .	99
A.25. Pruebas de Robustez. Disminución de 50 % en las ganancias del proceso. (1 signo, Lazo1). . . . .	99
A.26. Pruebas de Robustez. Disminución de 50 % en las ganancias del proceso. (1 signo, Lazo2). . . . .	99
A.27. Pruebas de Robustez. Aumento de 50 % en las constantes de tiempo. (1 signo, Lazo1). . . . .	99
A.28. Pruebas de Robustez. Aumento de 50 % en las constantes de tiempo. (1 signo, Lazo2). . . . .	100
A.29. Pruebas de Robustez. Disminución de 50 % en las constantes de tiempo. (1 signo, Lazo1). . . . .	100
A.30. Pruebas de Robustez. Disminución de 50 % en las constantes de tiempo. (1 signo, Lazo2). . . . .	100

A.31.Pruebas de Robustez. Aumento de 50 % en el tiempo muerto. (1 signo,Lazo1). . . . .	100
A.32.Pruebas de Robustez. Aumento de 50 % en el tiempo muerto. (1 signo,Lazo2). . . . .	100
A.33.Pruebas de Robustez. Aumento de 50 % en todos los parámetros. (1 signo,Lazo1). . . . .	101
A.34.Pruebas de Robustez. Aumento de 50 % en todos los parámetros. (1 signo,Lazo2). . . . .	101
A.35.Pruebas de Robustez. Disminución de 50 % en todos los parámetros. (1 signo,Lazo1). . . . .	101
A.36.Pruebas de Robustez. Disminución de 50 % en todos los parámetros. (1 signo,Lazo2). . . . .	101
A.37.Pruebas de Robustez. El peor caso. (1 signo,Lazo1). . . . .	102
A.38.Pruebas de Robustez. El peor caso. (1 signo,Lazo2). . . . .	102

## Índice de figuras

2.1. Diagrama de proceso multivariable de dos entradas y dos salidas. . . . .	5
2.2. Diagrama de proceso de estación de Nivel y Temperatura . . . . .	8
3.1. Algunas funciones de membresía: a) $\Gamma$ , b) $\Lambda$ , c) L, d) Sigleton. . . . .	19
3.2. Diagrama para representar las reglas del controlador . . . . .	29
3.3. Fuzificador Centro de Gravedad . . . . .	30
3.4. Fuzificador de centro promedio . . . . .	31
3.5. Comportamiento del error y su derivada durante la acción de control. .	34
3.6. Estructura de Controlador Tipo PID difuso. . . . .	35
3.7. Reglas para el caso de 1 signo. Salida 1. . . . .	38
3.8. Reglas para el caso de 1 signo. Salida 2. . . . .	38
3.9. Reglas para el caso de 2 signo. Salida 1. . . . .	39
3.10. Reglas para el caso de 2 signo. Salida 2. . . . .	39
3.11. Reglas para el caso sin signo. Salida 1. . . . .	40
3.12. Reglas para el caso sin signo. Salida 2. . . . .	40
4.1. Diagrama de Flujo del Recocido Simulado . . . . .	55
4.2. Conjuntos difusos de salida del controlador. . . . .	56
4.3. Base de reglas para la salida 1 . . . . .	56
4.4. Base de reglas para la salida2 . . . . .	56
4.5. Ajuste de los conjuntos difusos de entrada . . . . .	57
4.6. Respuesta del sistema sintonizado mediante RS. . . . .	57
4.7. respuesta del sistema sintonizado mediante RS. . . . .	58
4.8. Controlador de 16 reglas. Comparación de sintonía en servocontrol, lazo 1. Sólido: original. Punteado: Obtenido mediante RS. . . . .	58
4.9. Controlador de 16 reglas. Comparación de sintonía en servocontrol, lazo 2. Sólido: original. Punteado: Obtenido mediante RS. . . . .	58
4.10. Controlador de 16 reglas. Acoplamiento de lazo 2 debido a cambio en lazo 1. Sólido: original. Punteado: Obtenido mediante RS. . . . .	59
4.11. Controlador de 16 reglas. Acoplamiento de lazo 1 debido a cambio en lazo 2. Sólido: original. Punteado: Obtenido mediante RS. . . . .	59

5.1.	Diagrama de bloques del lazo de control multivariable . . . . .	61
5.2.	Definición de áreas A y B de la respuesta de primer orden . . . . .	65
5.3.	Respuesta de PID en lazo 1. Continua: Genético. Punteada: Propuesto por RS. . . . .	67
5.4.	Respuesta de PID en lazo 2. Continua: Genético. Punteada: Propuesto por RS. . . . .	68
6.1.	Calidad de éxito de controlador de 2 signos. . . . .	81
6.2.	Calidad de éxito de controlador de 1 signo. . . . .	82
B.1.	Respuesta de caso ideal. Lazo 1 . . . . .	103
B.2.	Respuesta de caso ideal. Lazo 2 . . . . .	103
B.3.	Respuesta al aumento de ganancias. Lazo 1 . . . . .	104
B.4.	Respuesta al aumento de ganancias. Lazo 2 . . . . .	104
B.5.	Respuesta a la disminución de ganancias. Lazo 1 . . . . .	104
B.6.	Respuesta a la disminución de ganancias. Lazo 2 . . . . .	105
B.7.	Respuesta al aumento en las constantes de tiempo. Lazo 1 . . . . .	105
B.8.	Respuesta al aumento en las constantes de tiempo. Lazo 2 . . . . .	105
B.9.	Respuesta a la disminución en las constantes de tiempo. Lazo 1 . . . . .	106
B.10.	Respuesta a la disminución en las constantes de tiempo. Lazo 2 . . . . .	106
B.11.	Respuesta al aumento en los tiempos muertos. Lazo 1 . . . . .	106
B.12.	Respuesta al aumento en los tiempos muertos. Lazo 2 . . . . .	107
B.13.	Respuesta a la disminución en los tiempos muertos. Lazo 1 . . . . .	107
B.14.	Respuesta a la disminución en los tiempos muertos. Lazo 2 . . . . .	107
B.15.	Respuesta al aumento en todos los parámetros. Lazo 1 . . . . .	108
B.16.	Respuesta al aumento en todos los parámetros. Lazo 2 . . . . .	108
C.1.	Pantalla principal de la interfaz Hombre-Máquina . . . . .	111
C.2.	Pantalla de introducción de datos del proceso multivariable. . . . .	113
C.3.	Pantalla de evaluación de función de costo. . . . .	114
C.4.	Pantalla de configuración de controlador difuso multivariable. . . . .	116
C.5.	Pantalla de configuración de estrategia de controlador difuso univariable con desacoplador. . . . .	117
C.6.	Pantalla de configuración de estrategia de controlador PID de 3 difusos con desacoplador. . . . .	118

## Capítulo 1

### Introducción

#### 1.1. Establecimiento del problema

En la industria los procesos de múltiples entradas y salidas (MIMO) son sumamente comunes. El control de esta clase de procesos no sería diferente al caso univariable si no existieran acoplamientos de una entrada con dos o más salidas. A causa de estos efectos de perturbación generados por la misma acción de control, es necesario tomar un enfoque diferente en el que las manipulaciones calculadas deben someterse a las restricciones que el acoplamiento impone.

Para la solución de este problema se han utilizado diversas técnicas: Controladores convencionales apoyados en bloques de cancelación, controladores basados en el concepto de espacio de estados, controladores no lineales, etc. Una de las tantas posibilidades que pueden utilizarse es la de los controladores implementados mediante sistemas difusos. El diseño y sintonía de éstos es y ha sido un tema importante de investigación.

Debido al prometedor potencial que encierran los controladores difusos y a las propiedades que se les ha atribuido de alta robustez y control ante la no linealidad, se considera de interés el indagar acerca de las estrategias de control difuso multivariable así como en métodos para sintonizarlas sin la necesidad de la presencia de un experto o de contar con información exhaustiva del proceso o del mismo controlador.

#### 1.2. Juitificación

Los sistemas difusos son un caso bastante interesante. Se les ha atribuido propiedades como alta robustez y capacidad de control frente a no linealidades, pero para que funcionen óptimamente se debe contar con un experto, cuyo conocimiento del control del proceso en particular es convertido de una base de reglas de manipulación en una ecuación no lineal que puede ser programada para ejecutarse automáticamente.

Sin embargo, el principal problema es que tal experto, por lo general, no existe y es necesario hacer los ajustes del controlador basados en alguna otra fuente de informa-

ción. El método más común que ha sido utilizado para hacer funcionar las aplicaciones controladas por sistemas difusos es el método de prueba y error, dado que, por lo general, el mismo controlador es lo suficientemente complejo como para poder hacer un ajuste preciso de manera intuitiva.

Respecto a la sintonización de los controladores difusos se ha hecho una investigación muy amplia, pero es una realidad que todavía resta mucho por hacer al respecto. En la bibliografía se pueden encontrar diferentes fuentes donde se reportan numerosos métodos para ajustar un controlador. Estos oscilan entre formas ordenadas de aplicar el método de prueba y error, reglas basadas en algún criterio, como puede ser pruebas de estado estable o respuesta a la frecuencia, hasta métodos basados en herramientas más sofisticadas como redes neuronales o algoritmos genéticos. De igual manera, la estructura misma del controlador encierra un número igual de posibilidades. Sin embargo, es en el potencial latente de los sistemas difusos donde reside su verdadero valor.

Por tales motivos, se ha pensado que la mejor estrategia a seguir es la búsqueda de un método que permita la sintonización de los controladores con la menor cantidad de información. Algoritmos basados en métodos estocásticos, donde se generan opciones al azar y son seleccionadas bajo un cierto criterio parecen ser la opción más adecuada cuando no se tiene mayor información del proceso que la que se puede obtener mediante las pruebas convencionales.

### 1.3. Objetivos

Los objetivos del presente trabajo pueden resumirse en los siguientes puntos:

- Diseño de estructuras basadas total o parcialmente en sistemas difusos con el fin de generar controladores que puedan ofrecer de manera simultánea el control de las variables de proceso y el combate del acoplamiento del sistema.
- Revisión de la literatura en búsqueda de un método que permita la sintonización del controlador difuso sin la necesidad de tener información exhaustiva del proceso o de la naturaleza del lazo de control; la selección de una técnica adecuada y la implementación aterrizada al problema de la sintonización de los parámetros de las estructuras propuestas.
- Evaluación del desempeño de los controladores diseñados y ajustados estableciendo una técnica convencional como base comparativa para determinar su calidad y robustez. Se entiende este concepto como insensibilidad al cambio de parámetros en el modelo del proceso.



## 1.4. Metodología

Para realizar objetivos anteriores se realizaron las siguientes acciones:

- Investigación respectiva a los conceptos de lógica difusa que se aterrizaron en la proposición de tres estrategias de control para procesos multivariables acoplados basadas en estos conceptos. (Estructura de un difuso multivariable, difusos univariados con desacoplador y PID difuso con desacoplador).
- Definición de los lineamientos del algoritmo de sintonización (Recocido Simulado) para su adecuado funcionamiento. Entre ellos deben contemplarse la función de costo, la configuración del problema, el vecindario, la calendarización del enfriamiento, etc.
- Optimización de las estrategias de control difusas y convencionales mediante el algoritmo de sintonización. Elegir a aquellas sintonías que ofrezcan el menor costo para cada estrategia.
- Diseño, aplicación y análisis de pruebas que determinaron el desempeño de las estrategias propuestas tanto en situaciones ideales como en situaciones donde existe cambio de parámetros del modelo y que dejan ver la robustez de la estructura evaluada.
- Comparaciones entre los resultados obtenidos por las estructuras propuestas y los resultados que se pueden obtener bajo las mismas condiciones con una técnica convencional. (PID con desacoplador)
- Análisis final de los resultados y deducción de conclusiones.

## 1.5. Descripción del documento de tesis

En el capítulo 1 se sientan las bases de la investigación. Se presenta el problema a resolver y una justificación de su importancia. Se establecen los objetivos y la metodología a seguir.

En el capítulo 2 se presentan los antecedentes. Se hace un breve resumen del estado del arte referente a lógica difusa y aplicaciones al control multivariable. Además se muestran los avances conseguidos en la línea de investigación que se sigue en el trabajo realizado.

En el capítulo 3 se discute brevemente los fundamentos de la lógica difusa y los componentes generales de un sistema difuso. A su vez, se describe la estructura de las estrategias de control basadas en esta teoría y que posteriormente se someterán a análisis.

En el capítulo 4 se trata lo referente a la técnica de optimización, describiendo su funcionamiento y las decisiones que se tomaron para implementar el algoritmo. Se muestran algunos ejemplos de su efectividad aplicándola a los controladores anteriormente descritos.

En el capítulo 5 se hace un breve resumen de la técnica de comparación y se muestra como puede mejorarse su desempeño mediante la aplicación de la técnica de optimización.

En el capítulo 6 se describe la metodología que se siguió para el diseño y aplicación de las pruebas. Se exponen los resultados más importantes y se hacen deducciones a partir de análisis de los mismos.

Finalmente, en el capítulo 7, se condensan las conclusiones alcanzadas y las recomendaciones pertinentes.

## Capítulo 2

### Antecedentes

#### 2.1. Breve resumen del estado del arte

Se entiende por un sistema multivariable aquél que tiene múltiples entradas o múltiples salidas. Además, debe de cumplirse que las entradas afecten en alguna proporción a las variables de proceso, es decir, que exista cierta interacción o acoplamiento. En caso contrario, el sistema puede ser dividido en varios procesos univariados y aplicar para su control las técnicas que son convenientes para estos casos.

En este trabajo en particular se analizó el caso más simple de control multivariable, el cual consiste en aquél que posee dos entradas y dos salidas únicamente, pero que existe interacción de una variable en la otra. El esquema básico del sistema descrito puede apreciarse en la figura 2.1. En ella se representan como  $G_{11}$  y  $G_{22}$  a las variables de interés del proceso, aquellas funciones de transferencia que relacionan a la variable controlada con su respectiva variable manipulada. Por otro lado,  $G_{12}$  y  $G_{21}$  representan los efectos de interacción. El problema del control multivariable consiste en mantener en la referencia deseada a las variables de proceso y eliminar los efectos que se presentarán debido a la interacción existente.

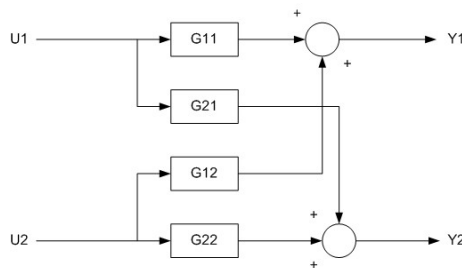


FIGURA 2.1: Diagrama de proceso multivariable de dos entradas y dos salidas.

La presencia de los bloques  $G_{12}$  y  $G_{21}$  son los que justifican aplicar una estrategia de control más compleja que el mero lazo retroalimentado. Debe tomarse en cuenta que el efecto de estos bloques puede ser entendido como perturbaciones al sistema, y como tales, el lazo de control debe de ser capaz de contrarrestar en caso de que se presenten.

Aún así, es deseable y conveniente en la mayoría de las aplicaciones disminuir estos disturbios lo máximo posible. Por tal motivo, se han realizado investigaciones y en la bibliografía se han expuesto diferentes soluciones para resolver esta problemática. En [4] se explica la manera en que se puede analizar un sistema multivariable para poder implementar la estrategia de control convencional apoyado en desacopladores. En [5] se muestra el procedimiento para implementar un controlador digital desacoplante. En [17] se describe las bases para el diseño de controladores basados en el concepto de espacio de estado, técnica que resulta ser muy poderosa al aplicarla a sistemas multivariables de alto grado de complejidad.

Como ya fue mencionado. Es la intención de este trabajo realizar estudios acerca de la aplicación de los sistemas difusos, lo cual constituye una posibilidad que puede resultar de sumo interés. En general, el campo referente a la naturaleza y aplicación de la lógica difusa, ha sido objeto de una extensa investigación.

De la bibliografía [19, 6] puede saberse de los orígenes de los sistemas difusos. Estos fueron inicialmente concebidos por Zadeh en los años 60, quien estudió las posibilidades que puede ofrecer esta extensión de la lógica clásica para implementar lo que se conoce como razonamiento aproximado. No fue hasta los años 70 en que se comenzó a realizar controladores difusos para resolver problemas teóricos (tales como el péndulo invertido) y fue en los años 80 cuando se tuvo un verdadero auge al resolver problemas industriales como el caso de la procesadora de cemento o el control del tren bala, así como numerosas aplicaciones en aparatos para el uso diario.

En lo que se refiere a los avances que se han tenido en ingeniería de control, puede decirse que se han obtenido numerosas reglas para configurar el controlador y se han propuesto variados esquemas y estrategias en los que éstos intervienen para mejorar el desempeño de los sistemas automáticos. Sistemas difusos como el principal controlador del lazo retroalimentado [6] son sumamente comunes; sin embargo, esquemas tales como el control supervisorio [11] (en los que se calculan diferentes valores de referencia con el fin de aclarar la respuesta) han sido también un objeto de interés para los investigadores. Esquemas en los que lo difuso actúa como supervisor para cambiar los parámetros de un controlador convencional es también una alternativa común. En [21] se explica como aplicar la técnica, conocida como “Gain Scheduling”. En [13] se muestra una metodología similar que ofrece la capacidad de autosintonización del controlador. Esquemas donde se hace sinergia entre los sistemas difusos y otras técnicas de inteligencia artificial también están disponibles. Por ejemplo, en [2] se muestra la manera en que un esquema basado en sistemas difusos y redes neuronales puede ser concebido para el buen control de un motor sincrónico. Otros esquemas similares pueden encontrarse en la literatura. En [12] no solo se muestra un esquema semejante, sino también un algoritmo para su auto sintonización.

Lo referente a la sintonía del controlador difuso también ha sido ampliamente estudiado. [22] muestra una metodología parcial para realizar la sintonización de los

de tipo multivariable, además de presentar recomendaciones para el ajuste restante a prueba y error. Por otra parte, en [6] se expone una metodología para la sintonización de controladores difusos univariables.

En intentos de saltar la dificultad que presenta la falta de métodos formales para sintonizar esta clase de controladores, algoritmos basados en diversos métodos han sido presentados. En [10] se muestra una metodología para conseguir una sintonización utilizando algoritmos genéticos. Numerosos métodos basados en el concepto de gradiente han sido propuestos, tales como los mencionados en [26, 3]. En el primero, se aplica un algoritmo basado en el gradiente para determinar las reglas del controlador, en el segundo, se determina con un algoritmo basado en el mismo concepto los tamaños de las bases de los conjuntos difusos de entrada de un controlador difuso univariable para que presente efecto PID sobre el proceso a controlar.

Concerniente también a las técnicas de sintonización, se tienen reportadas diversas aplicaciones de la técnica de recocido simulado utilizado para diversos problemas de optimización. En [16] se muestra la manera de aplicar este método para optimizar las dimensiones de los elementos de sistemas mecánicos referente a variables como velocidades y aceleraciones. En [20] se muestra su efectividad al ajustar un controlador en espacio de estados en base al estimador óptimo cuadrático.

## 2.2. Línea de investigación

En lo que se refiere a la línea de investigación seguida, se han realizado diferentes estudios para implementar el control difuso en esquemas diversos y probar su efectividad. Esfuerzos iniciales fueron reportados por Ruiz y Rodríguez en [24, 25], donde se obtuvo el modelo de un proceso de nivel y temperatura y se diseñaron controladores difusos multivariables que fueron comparados frente a técnicas convencionales. Un diagrama representativo del proceso modelado aparece en la figura 2.2.

En reportes posteriores, se trabajó con diversas simulaciones, algunas que utilizaban los modelos reportados en [24] y algunos otros tomados de la bibliografía. Villa [5] hace uso de un modelo multivariable reportado en [22], para aplicarla a sistemas difusos de tres conjuntos difusos en cada entrada y se sintonizaron los controladores por el método de prueba y error. De igual manera, se comparó contra técnicas convencionales donde se estableció cierta mejoría bajo ciertas condiciones. El trabajo mostrado por Bretado [1] describe la manera en que la técnica de control difuso multivariable fue aplicada al control de presión y flujo de inyección de oxígeno en una estación de recalentamiento químico existente en el ámbito industrial de la empresa AHMSA. En este caso se identificó el proceso para obtener un modelo en espacio estado y se sintonizó al controlador de manera empírica, obteniendo buenos resultados.

El trabajo de Díaz [23], muestra la manera de obtener un controlador difuso uni-

variable que tuviera el poder de emular un controlador PID. Los resultados fueron utilizados por Palma [3], donde se utilizó un algoritmo basado en el gradiente para obtener una mejor sintonización del llamado PID difuso emulador (PIDDE). En este último caso se utilizó las reglas de sintonización reportadas en [15] para realizar la comparación del desempeño de las estrategias de control.

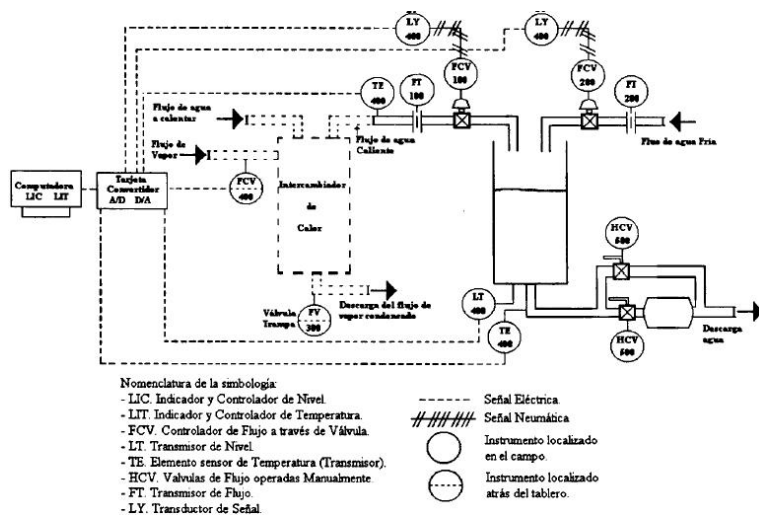


FIGURA 2.2: Diagrama de proceso de estación de Nivel y Temperatura

## Capítulo 3

# Estructura del controlador difuso

### 3.1. Introducción

Los controladores difusos basados en conocimiento (FKBC por sus siglas en inglés) son un grupo particular de los sistemas basados en conocimiento (KBS), los cuales se caracterizan por utilizar la experiencia de un experto para realizar la toma de sus decisiones. En el caso de un controlador difuso, la heurística del operador de una cierta planta o el conocimiento de un manual puede ser plasmada en un algoritmo que puede ser ejecutado por una computadora, siguiendo las líneas de razonamiento trazadas por la fuente.

La propiedad de realizar la transmutación del conocimiento humano en una ecuación no lineal es el principal argumento que justifica el esfuerzo en investigación en los sistemas difusos, dado que de esta manera, será posible dar soluciones de control para sistemas de los que no se tiene una descripción matemática satisfactoria o cuya representación es simplemente demasiado compleja como para darle una aplicación práctica.

Como en el caso general de los KBS, los FKBC almacenan el conocimiento que los caracteriza en la forma de una base de reglas, las cuales ofrecen una salida de control específica ante ciertas condiciones de las señales de entrada. Por lo general, Las variables que más comúnmente se utilizan para los controladores difusos consisten en las señales de error y derivada del error. Para el caso de las salidas de éstos generalmente se escoge la manipulación o el cambio en la manipulación. Haciendo una analogía con los controladores convencionales, esto cambia sus propiedades de control proporcional derivativo a control proporcional integral.

Las estructuras de los sistemas difusos son diversas, dado que se conforman de varios elementos importantes y al combinarlos entre sí se obtienen un sinnúmero de variaciones. Sin embargo, pocas de ellas son realmente útiles para realizar acción de control. El objetivo de esta sección consiste en dar una breve descripción de los elementos que conforman al sistema difuso y especificar cuales de ellos se utilizarán en las estructuras de los controladores de las estrategias que se proponen.

## 3.2. Lógica Difusa

### 3.2.1. Definiciones básicas

La teoría de conjuntos difusos es una extensión de la teoría de conjuntos clásica, donde cada elemento posee un grado de pertenencia que se encuentra dentro del intervalo  $[0,1]$ . Esta característica marca la diferencia, dado que normalmente un elemento del universo de discurso simplemente pertenece a un conjunto o no pertenece a él. Por lo tanto, para poder describir un conjunto difuso no es suficiente hacer una lista de sus elementos constitutivos. Es necesario también indicar el grado de pertenencia de cada uno de ellos. El grado de pertenencia se obtiene evaluando al elemento en una función de membresía que se escribe para el elemento  $x$  de un conjunto  $A$  del universo  $U$  como  $A(x)$ . Siendo así podemos enlistar el contenido del conjunto  $A$  de la manera siguiente.

$$A = \{(x, \mu_A(x)) \mid x \in U\} \quad (3.1)$$

Existen otras formas útiles de escribir el contenido de los conjuntos difusos. Cuando el universo  $U$  es continuo se puede utilizar la notación integral.

$$A = \int_U \mu_A(x)/x \quad (3.2)$$

Asimismo, para el caso de un universo discreto se puede utilizar la notación sumatoria.

$$A = \sum_U \mu_A(x)/x \quad (3.3)$$

Donde se hace el comentario que ni la integral ni la sumatoria se refieren a las operaciones de suma e integración, sino al listado del total de todos los elementos del conjunto. De igual forma, el símbolo  $/$  no debe ser interpretado como división.

### 3.2.2. Propiedades de los conjuntos difusos

La teoría de los conjuntos difusos es una extensión de la teoría clásica, por lo tanto, es natural que algunas de las propiedades sean compartidas entre ambas clases de conjuntos. Sin embargo, algunos conceptos son exclusivos de los conjuntos difusos. Se presentan algunas de las propiedades y conceptos básicos que se utilizan para caracterizarlos.

#### **Soporte**

El soporte consiste en el rango del dominio en el que el conjunto tiene elementos que le pertenecen en grado significativo. El resultado de obtener el soporte de un conjunto difuso es un conjunto nítido que enlista a todos los puntos  $x$  que pertenecen



al conjunto difuso  $A$ . Denotando la operación de obtención del soporte como  $\text{supp}()$  puede enunciarse lo explicado mediante la siguiente expresión.

$$\text{supp}(A) = \{x \in U \mid \mu_A(x) \geq 0\} \quad (3.4)$$

### Corte $\alpha$

El concepto de corte  $\alpha$  es una extensión del de soporte. Se define como el conjunto nítido formado por elementos de  $A$  cuyo valor de pertenencia es mayor o igual  $\alpha$ . También existe el llamado corte severo (strong cut), el cual desprecia el caso igual a  $\alpha$ .

$$A_\alpha = \{x \in U \mid \mu_A(x) \geq \alpha\} \quad (3.5)$$

### Centro

El concepto de centro de un conjunto difuso se entiende como el valor medio  $x_c$ , en el cual el valor de pertenencia es máximo. Por ejemplo, en un conjunto difuso de forma triangular, el centro del conjunto es el centro del triángulo; para una función de membresía trapezoidal, el centro es el valor medio del lado menor del trapecio. Cuando se tienen funciones de tipo rampa donde la parte más alta permanece en el valor máximo para el resto del dominio, el punto de quiebre es considerado como el centro del conjunto difuso.

### Altura

Por altura se entiende como máximo valor de pertenencia que puede tener un conjunto difuso determinado. Generalmente este valor máximo es de 1.

### Punto de cruce

El punto de cruce es el lugar donde si se tienen los conjuntos difusos  $A$  y  $B$ , con valores de pertenencia  $A(x)$  y  $B(y)$ , tenemos que  $x = y$ . La literatura recomienda que este punto debe ocurrir donde las funciones de pertenencia es de 0.5 para aplicaciones de control. [6]

### Forma convexa

Este concepto se refiere a la única existencia de un máximo o un rango de máximos en el conjunto difuso, de tal manera que no se presenten cambios en la concavidad de  $A(x)$ . Formalmente se tiene que un conjunto difuso  $A$  donde  $x_1, x_2 \in \mathbb{R}^n$  y  $A(x_1) \geq 0$ ,  $A(x_2) \geq 0$ , es convexo en  $\mathbb{R}^n$  si y solo si

$$\mu_A[\lambda x_1 + (1 - \lambda)x_2] \geq \min(\mu_A(x_1), \mu_A(x_2)) \quad (3.6)$$

donde  $\lambda \in [0,1]$ .

### 3.2.3. Operaciones básicas con conjuntos difusos

Dado que los FKBC trabajan con conjuntos difusos, es necesario el conocer las operaciones que se pueden realizar con ellos para poder aplicarlas como elementos del sistema de control. Dado que la lógica difusa es una extensión de la lógica clásica, es lógico pensar que todas las operaciones clásicas pueden aplicarse a los conjuntos difusos. Sin embargo, por su propia naturaleza, a estos últimos se les pueden aplicar otras operaciones que se mostrarán a continuación.

#### Complementos difusos

Siendo  $c: [0,1] \rightarrow [0,1]$  que convierte el conjunto  $A$  en su complemento  $\bar{A}$  debe de cumplir con los siguientes axiomas.

Axioma 1:  $c(0)=1$  y  $c(1)=0$ .

Axioma 2: Para todo  $a, b \in [0,1]$  si  $a \geq b$ , entonces  $c(a) \geq c(b)$ .

Respetando las condiciones anteriores se obtiene la definición clásica del complemento

$$\mu_{\bar{A}} = 1 - \mu_A(x) \quad (3.7)$$

Sin embargo, ésta no es la única definición posible, dado que cualquier otra que cumpla las condiciones anteriores puede ser tomada como complemento. Así otra propuesta para la operación complemento es forma de Sugeno definida por

$$c_\lambda(a) = \frac{1 - a}{1 + \lambda a} \quad (3.8)$$

donde  $\lambda \in (-1, \infty)$ .

#### Unión difusa

La operación de unión, conocida también como norma S o co-norma T también es aplicable a los sistemas difusos. Puede definirse  $s: [0,1] \times [0,1] \rightarrow [0,1]$  que cumple con las axiomas siguientes.

Axioma 1 :  $s(1,1) = 1$ ;  $s(0, a) = s(a, 0) = a$ .

Axioma 2:  $s(a,b) = s(b,a)$

Axioma 3: Si  $a \geq a'$ ,  $b \geq b'$ , entonces  $s(a,b) \geq s(a', b')$ .

Así, la definición clásica que cumple con todas estas condiciones es la operación  $\max(a,b)$ .

$$\mu_{A \cup B}(x) = s(\mu_A(x), \mu_B(x)) = \max(\mu_A(x), \mu_B(x)) \quad (3.9)$$

Igualmente, existen otras definiciones posibles que han sido propuestas en la literatura. Un ejemplo es la suma algebraica

$$s_{SA}(a, b) = a + b - ab \quad (3.10)$$

### Intersección difusa

La tercera operación básica que se puede realizar con conjuntos difusos es la intersección, conocida en la teoría de la lógica difusa como la norma  $T$ . De manera similar al caso anterior, existe una definición que es aplicada en la teoría clásica de conjuntos y es extendida al caso difuso. Además, existen otras formas que se pueden proponer respetando los axiomas que definen a la operación.

Se entiende por norma  $T$  a la operación  $t : [0,1] \times [0,1] \rightarrow [0,1]$  que satisface los axiomas:

Axioma 1:  $t(0,0) = 0$ ;  $t(a,1) = a$ ;  $t(1,b) = b$

Axioma 2:  $t(a,b) = t(b,a)$

Axioma 3: Si  $a \leq \bar{a}$  y  $b \leq \bar{b}$ , entonces  $t(a,b) \leq t(\bar{a}, \bar{b})$

Axioma 4:  $t(t(a,b),c) = t(a,t(b,c))$

De esta manera, la operación básica que se define como norma  $T$  es la operación mínimo.

$$\mu_{A \cap B}(x) = t(\mu_A(x), \mu_B(x)) = \min(\mu_A(x), \mu_B(x)) \quad (3.11)$$

Otra de tantas definiciones que cumplen con los axiomas es, por ejemplo, el producto algebraico

$$t_{PA}(a, b) = ab \quad (3.12)$$

### Operadores promedio

Observando las definiciones anteriores y realizando una comparación de los resultados de las diferentes normas  $S$  y  $T$  para los mismos valores de pertenencia de  $A(x)$  y  $B(x)$ , podremos notar que existen operadores límite que siempre son los mismos. Así, para la norma  $S$  los resultados se encontrarán siempre en el intervalo  $[\max(a,b), s_{SD}(a,b)]$  y para los operadores de la norma  $T$  los límites quedan determinados por el intervalo  $[\min(a,b), t_{PD}(a,b)]$ .

$s_{SD}$  y  $t_{PD}$  denotan las operaciones de suma y producto drástico, las cuales están definidas mediante las expresiones 3.13 y 3.14.

$$s_{SD}(a, b) = \begin{cases} a, & b = 0 \\ b, & a = 0 \\ 1, & b \neq 0, a \neq 0 \end{cases} \quad (3.13)$$

$$t_{PD}(a, b) = \begin{cases} a, & b = 1 \\ b, & a = 1 \\ 0, & b \neq 1, a \neq 1 \end{cases} \quad (3.14)$$

Por lo tanto, existe un intervalo  $(\min(a,b), \max(a,b))$  que no es alcanzado por ninguno de los operadores mencionados hasta ahora. Debido a ello se llegó a la conclusión de que deberían existir operadores cuyos resultados quedaran dentro de ese intervalo y se les dio el nombre de operadores promedio. Ejemplos de ellos se pueden encontrar en la literatura [19].

### 3.3. Descripción del controlador difuso

#### 3.3.1. Estructura general del controlador difuso

En el contexto general, Los sistemas difusos se pueden dividir en tres tipos: en los sistemas difusos puros, los sistemas difusos Takagi-Sugeno y los sistemas difusos con fuzificador y defuzificador.

Los Sistemas difusos puros trabajan exclusivamente con variables lingüísticas tanto de entrada como de salida. En este tipo de sistema difuso, las reglas que lo conforman se refieren a una relación entre conjunto difusos. Una de las formas más comunes consiste en

$$IF \ x \ es \ A \ THEN \ B \ es \ C \quad (3.15)$$

Debido a que los sensores con los que se cuenta disponibles en la industria y en el campo de investigación arrojan valores numéricos (nítidos), los sistemas difusos puros son incompatibles con el mundo real; destinándolos exclusivamente al campo de la investigación teórica.

debido a estas limitaciones, se hicieron propuestas que permitieran superarlas satisfactoriamente. La estructura del controlador propuesto por Takagi-Sugeno tiene como salida una función del valor de entrada. Esto queda ejemplificado en la expresión

$$IF \ x \ es \ A \ THEN \ B \ es \ y = cx \quad (3.16)$$

Los sistemas difusos prácticos, utilizan generalmente el esquema de fuzificador y defuzificador. Estos consisten en utilizar un sistema difuso puro adicionando una serie de técnicas que se utilizan para convertir el valor nítido a la pertenencia de un conjunto difuso para que pueda ser procesado por la máquina de inferencia y posteriormente vuelto a un valor numérico una vez que se haya procesado la salida. El esquema con fuzificador y defuzificador es el que se utilizará en todas las estructuras de FKBC propuestas en este documento.

### 3.3.2. Normalización

El rango en que puede ser utilizado el algoritmo del FKBC puede ser muy variado; principalmente depende de la aplicación. Algunos criterios que normalmente se utilizan para determinar el universo de discurso donde se colocan los conjuntos difusos son los siguientes:

- Los rangos de operación del proceso.
- El rango de valores del tipo de variable int.
- El rango de valores del tipo de variable float.
- El rango  $[-1,1]$ .
- El rango en que se miden las variables de proceso  $[0,100]$ .

De utilizarse un rango normalizado respecto a alguno de los criterios anteriores, éste puede ser ajustado mediante una ganancia multiplicativa externa al rango de operación que llegue a ser conveniente. De igual manera, esto será aplicado al bloque de la salida. Se tiene la ventaja de este modo que se cuenta con un parámetro de ajuste similar a lo que es la ganancia proporcional, lo cual al aplicar el algoritmo de sintonía ayudará a mejorar el desempeño del controlador.

### 3.3.3. Fuzificador

Como se ha mencionado anteriormente, los sistemas meramente difusos no tienen una aplicación práctica, dado que las variables se pueden medir y controlar están expresadas en valores nítidos y no en términos lingüísticos. Para poder procesar en lenguaje natural la información de sensores y actuadores, es necesario realizar una interfase que permita fuzificar (convertir a la pertenencia de uno o varios conjuntos difusos) y traducir al resultado procesado en una cantidad nítida.

Los fuzificadores son los bloques que convierten el valor de entrada en valores de pertenencia de uno o varios conjuntos difusos. Existen tres tipos de fuzificadores que se utilizan con regularidad, los cuales consisten en el fuzificador tipo singleton, el fuzificador tipo triangular y el fuzificador tipo gaussiano. Se define el fuzificador singleton como:

$$\mu_{A'}(x) = \begin{cases} 1, & x = x^* \\ 0, & x \neq x^* \end{cases} \quad (3.17)$$

El fuzificador triangular se define por la expresión:

$$\mu_{A'}(x) = \begin{cases} \prod(1 - \frac{|x_i - x_i^*|}{b_i}), & |x_i - x_i^*| < b_i \\ 0, & \text{otro caso} \end{cases} \quad (3.18)$$

El fuzificador Gaussiano está definido por la expresión:

$$\mu_{A'}(x) = \left\{ \prod (e^{-\frac{(x_i - x_i^*)^2}{a_i}}) \right\} \quad (3.19)$$

El fuzificador tipo singleton es el de mayor velocidad de cálculo, dado que es solo el valor de 1 colocado en el valor nítido de entrada  $x^*$ . Debido a que cualquier otro valor será menor o igual a la unidad, el resultado de la fuzificación consiste en el valor de pertenencia de  $x^*$  para los conjuntos difusos involucrados.

Los fuzificadores triangular y gaussiano son más complejos y por lo mismo tienen una menor velocidad de cálculo. Sin embargo, puede simplificarse en cierta medida los cálculos si las funciones de pertenencia de entrada del FKBC son del mismo tipo que el fuzificador. Sin embargo, se puede demostrar que estos últimos dos tipos ofrecen la capacidad de rechazo al ruido que el primer fuzificador no posee.

A sabiendas que la velocidad de cálculo es primordial en un sistema de control y que existen otro tipo de medidas posibles contra el ruido de la señal de entrada, se ha preferido el fuzificador tipo singleton para el desarrollo del FKBC.

### 3.3.4. Conjuntos difusos de entrada

Parte importante del proceso de un sistema de control consiste en la definición de las entradas y salidas del sistema. El caso de los controladores difusos no es la excepción. En el caso del diseño de FKBC es necesario a su vez la definición de los conjuntos difusos de entrada y salida, dado que son en base a estos que la máquina de inferencia tomará las decisiones de sus acciones de control.

Al igual que los sistemas convencionales, los FKBC trabajan con las señales de error y derivadas del error de las variables de proceso respecto a sus puntos de referencia. Suponiendo que el FKBC será programado en un sistema de cómputo o un microcontrolador, es más adecuado definir estas señales en su forma discreta como procede

$$e(k) = y_{ref} - y(k) \quad (3.20)$$

$$\delta e(k) = y_{ref} - y(k) \quad (3.21)$$

donde  $y_{ref}$  se refiere a la referencia del controlador e denota al error, y se refiere la variable de proceso y e al cambio en el error.

Se ha determinado que es más conveniente diseñar al controlador para que su salida sean los cambios en el flujo de cada una de las variables, dado que de esta manera se garantizará que el controlador tenga efecto integral y se elimine por completo el error de estado estable. Esto se demuestra al ver como la forma integral al derivarla convierte a la manipulación en la derivada de esta variable.

$$u(t) = K_p e(t) + K_i \int_t e(t) dt \quad (3.22)$$

$$\frac{du(t)}{dt} = K_p \frac{de(t)}{dt} + K_i e(t) \quad (3.23)$$

En forma discreta se obtiene la estructura del controlador propuesto:

$$\Delta u(k) = K_p \Delta e(t) + K_i e(t) \quad (3.24)$$

$$\Delta u(k) = u(k) - u(k - 1) \quad (3.25)$$

Una vez definidas las variables de entrada y de salida deben de definirse los conjuntos difusos que las representarán. Especialmente en el caso de utilizar un solo controlador difuso para manipular tanto el nivel como la temperatura, se debe tener cuidado con la cantidad conjuntos de entrada dado que el número de reglas depende directamente de ello. Para este caso, donde se ha establecido al error y la derivada del error de cada una de las variables controladas, las reglas serán proporcionales a

$$R = n^I \quad (3.26)$$

donde R es la cantidad de reglas, n es la cantidad de conjuntos difusos e I es la cantidad de entradas del sistema. Por ejemplo, Si tomamos tres conjuntos difusos de entrada tendremos una base de 81 reglas y si utilizamos 5, una base de 625 reglas.

Para los conjuntos difusos de entrada se distribuyeron en el espacio normalizado un total de tres conjuntos difusos para cada valor de entrada, a los cuales se les dio los valores lingüísticos de NEGATIVO (N), CERO (Z) y POSITIVO (P). Para distribuirlos inicialmente, antes de la aplicación de auto sintonía se tienen dos posibilidades: una es seguir las recomendaciones reportadas en [6], donde se busca que el traslape entre ambos conjuntos difusos sea de 0.5; la otra opción consiste en utilizar la distribución de los conjuntos difusos mencionados en [14] para tener una aproximación a una buena sintonización.

La forma de los conjuntos difusos es también de relativa importancia. En la bibliografía se han reportado numerosas formas para definirlos. En este caso se buscaron formas simples que faciliten el cálculo de las diferentes operaciones y que cuenten con la suficiente flexibilidad para dar la libertad suficiente al algoritmo de autosintonía.

Los tres tipos de funciones utilizadas en el controlador difuso son la función  $\Lambda$ , la función  $\Gamma$  y la función L.

La función  $\Gamma$  está definida por la expresión

$$\Gamma(x, \alpha, \beta) = \begin{cases} 0, & x < \alpha \\ \frac{x-\alpha}{\beta-\alpha}, & \alpha \leq x \leq \beta \\ 1, & x > \beta \end{cases} \quad (3.27)$$

La función  $\Lambda$  está definida a su vez por la expresión

$$\Lambda(x, \alpha, \beta, \gamma) = \begin{cases} 0, & x < \alpha \\ \frac{x-\alpha}{\beta-\alpha}, & \alpha \leq x \leq \beta \\ \frac{\gamma-x}{\gamma-\beta}, & \beta \leq x \leq \gamma \end{cases} \quad (3.28)$$

La función  $L$  está definida a su vez por la expresión

$$L(x, \alpha, \beta) = \begin{cases} 1, & x < \alpha \\ \frac{\beta-x}{\beta-\alpha}, & \alpha \leq x \leq \beta \\ 0, & x > \beta \end{cases} \quad (3.29)$$

Otra función sumamente utilizada en los controladores difusos es la función singleton, aunque se aplica especialmente para el caso de los conjuntos difusos de salida. La función singleton se define mediante la expresión

$$S(x, \alpha) = \begin{cases} 1, & x = \alpha \\ 0, & x \neq \alpha \end{cases} \quad (3.30)$$

donde 3.27, 3.28 y 3.29 se utilizarán para los conjuntos  $N$ ,  $Z$  y  $P$  de cada entrada, respectivamente. Las forma de estas funciones se muestra en la figura 3.1.

### 3.3.5. Ganancias de Normalización

A parte de los valores que puedan tomar los conjuntos difusos en particular, siempre es posible escalar los valores de entrada o de salida multiplicándolos por algún número que sea conveniente al caso. Normalmente esto se hace con el fin de mantener las condiciones del controlador constantes, dado que de lo contrario habría que hacer cambios durante la ejecución que podrían resultar tediosos.

Un ejemplo de lo que se pretende explicar es el caso del cambio en referencia. El diseñador puede considerar un caso en particular de cambio en referencia como puede ser el escalón unitario y ajustar el controlador para que ante esa entrada el sistema de la mejor respuesta posible. Sin embargo, si en la práctica es necesario hacer cambios de referencia de diferente magnitud, es obvio entonces que las posiciones de los conjuntos difusos no serán las mismas para que el sistema pueda responder como en el momento en que se diseñó la aplicación.

Esto puede fácilmente resolverse colocando una ganancia en cada una de las entradas del controlador que divida las señales de error y sus derivadas entre la magnitud



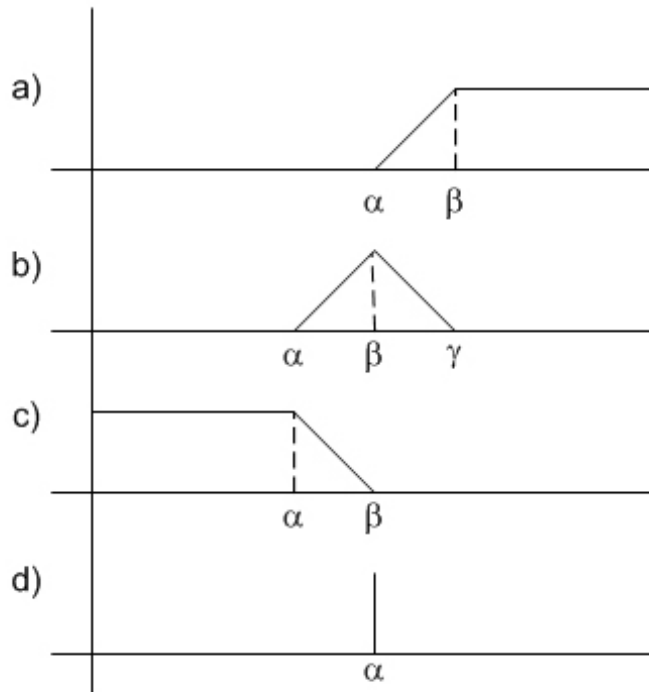


FIGURA 3.1: Algunas funciones de membresía: a)  $\Gamma$ , b)  $\Lambda$ , c)  $L$ , d) Sigleton.

del cambio en referencia y colocar otros en las salidas que multipliquen la magnitud de la manipulación por la amplitud del cambio en referencia. De esta manera, con la ayuda de estas ganancias adaptables, el controlador operará siempre bajo las mismas condiciones y la respuesta será la misma.

Debe de tomarse en cuenta, como en todos los controladores, que esta propiedad se verá afectada por la linealidad del proceso, dado que cambios en la linealidad se pueden entender como cambios en el modelo a diferentes puntos de operación. Sin embargo, esta es una dificultad a la que todo controlador deberá enfrentarse en el momento de controlar un proceso real.

Esta propiedad no tiene por que ser exclusivamente utilizada para el fin explicado. Es posible ahorrar tiempo y esfuerzo en el ajuste del controlador si simplemente se multiplica o divide por una ganancia en la entrada o salida del controlador que si se manipulan todos los conjuntos. Las reglas a seguir son sencillas y pueden enlistarse de la forma siguiente:

Suponiendo una ganancia de valor  $a$ , donde  $a > 1$ :

- Multiplicar por  $a$  es equivalente a estrechar las bases del conjunto a  $1/a$ , lo cual hará más sensible el controlador.
- Dividir entre  $a$  es equivalente a ensanchar las bases del conjunto hasta el punto

a, lo cual hará menos sensible al controlador.

De esta manera si la intención es solamente modificar las bases del controlador, no es necesario mover los conjuntos, sino simplemente multiplicar por el valor deseado. El único caso en que es necesario moverlas es cuando se desea que los conjuntos difusos tengan una amplitud desigual a lo largo del universo de discurso. En casos, así, dado que la ganancia queda a cargo del multiplicador inicial, es suficiente mover estas bases dentro de un espacio normalizado, convencionalmente 1.

En el trabajo se utilizarán ambas posibilidades de variación, bajo la clara idea de que los cambios son equivalente e intercambiables.

### 3.3.6. Conjuntos difusos de salida

Para los conjuntos difusos de salida se tiene una mayor libertad para designarlos, dado que la cantidad de reglas es independiente del número de los mismos. La precaución que se debe procurar es no utilizar más de los necesarios, es decir, tener la certeza que sean utilizados todos los que han sido definidos.

Una manera de determinar los conjuntos difusos es basados en la metodología mencionada en la bibliografía [22], la cual ofrece, para el caso del controlador multivariable, la posición de los conjuntos difusos de salida basados en la respuesta estática del sistema. La metodología consiste en obtener las manipulaciones a partir de las salidas en estado estable normalizadas.

Las salidas en estado estable de un sistema pueden obtenerse mediante la expresión

$$u(k)K = y(\infty) \quad (3.31)$$

$u(k)$  representa a manipulación,  $K$  es la ganancia del sistema y  $y(\infty)$ , las salidas en estado estable. Si redefinimos estos valores escalares como matrices para contemplar todos los casos posibles que pueden presentarse en un sistema multivariable y despejamos para la manipulación se obtiene

$$\mathbf{Y} = \mathbf{K}\mathbf{U}(\infty) \quad (3.32)$$

$$\mathbf{U} = \mathbf{K}^{-1}\mathbf{Y}(\infty) \quad (3.33)$$

Dado que son dos salidas, existen un total de nueve combinaciones posibles, puesto que cada una de ellas puede tender a subir, a bajar o a quedarse en su valor actual. Es fácil obtener los valores de manipulación que hacen posibles cada una de las combinaciones de las salidas de estado estable al definir  $\mathbf{Y}(\infty)$  como se muestra en 3.34.

$$\mathbf{Y}(\infty) = \begin{bmatrix} -1 & -1 & -1 & 0 & 0 & 0 & 1 & 1 & 1 \\ -1 & 0 & 1 & -1 & 0 & 1 & -1 & 0 & 1 \end{bmatrix} \quad (3.34)$$

Finalmente se asigna una etiqueta a cada uno de los conjuntos difusos, cuyas posiciones centrales consistirán en las que se obtuvieron mediante 3.33. Se utilizarán funciones triangulares definidas por 3.28 y 3.27 y 3.29 para los extremos. Otra posibilidad es definir a todos ellos mediante la función singleton. Las etiquetas se establecen como:

$$\mathbf{U} = \begin{bmatrix} mF_1 & mF_2 & mF_3 & mF_4 & mF_5 & mF_6 & mF_7 & mF_8 & mF_9 \\ mC_1 & mC_2 & mC_3 & mC_4 & mC_5 & mC_6 & mC_7 & mC_8 & mC_9 \end{bmatrix} \quad (3.35)$$

Donde se ha definido a  $mF_1$  como la etiqueta para el menor de los valores y  $mF_9$  para el máximo de todos; lo mismo aplica para  $mC$ . Es de observarse que no necesariamente se obtendrán los conjuntos en orden alguno, dado que la magnitud de las manipulaciones depende del proceso en cuestión.

Debido a que las manipulaciones obtenidas por este método solo permiten que las salidas tiendan en estado estable a los valores de 1, 0 y -1, se consideró interesante investigar si era posible obtener otros valores diferentes. Para ello se aplicaron todas las combinaciones posibles de las manipulaciones mostradas en 3.35 y se obtuvieron los valores hacia los que convergen las salidas del proceso. En base a esta información será posible definir posteriormente las reglas del controlador difuso. Los valores se obtuvieron haciendo uso de 3.32. En este caso  $\mathbf{U}$  tendrá la forma

$$\mathbf{U} = \begin{bmatrix} mF_1 & \dots & mF_1 & mF_2 & \dots & mF_2 & \dots & mF_9 & \dots & mF_9 \\ mC_1 & \dots & mC_9 & mC_1 & \dots & mC_9 & \dots & mC_1 & \dots & mC_9 \end{bmatrix} \quad (3.36)$$

En este caso, deben seleccionarse los pares plausibles, es decir que no generen un efecto desbalanceado de valor de estado estable en la salida del proceso. Esto depende del caso particular, pero puede esbozarse que estos pares deben de cumplir con la norma

$$||mF(i) - mC(j)|| < \epsilon \quad (3.37)$$

Para observar el efecto de los cambios de signos de cada una de las funciones de transferencia, se consideró aplicar la metodología extendida a los tres casos de signos distintos: Un signo diferente, dos signos diferentes, todos los signos iguales.

Sin embargo, debe tenerse en cuenta que solo las combinaciones que consiguen el movimiento de una sola variable son las que tienen propiedades desacoplantes. La selección de otros pares de combinaciones puede ser de utilidad para aplicaciones donde el acoplamiento no represente una perturbación significativa.

### Caso de un signo diferente

Se aplicó la metodología al caso de un signo diferente tomando las ganancias estáticas de la ecuación de la planta, obteniendo así la expresión

$$\mathbf{K} = \begin{bmatrix} 6.4 & 2,1 \\ -0.4 & 0.35 \end{bmatrix} \quad (3.38)$$

Aplicando 3.32 se obtuvieron los valores de convergencia que se resumen en la tabla 3.1.

TABLA 3.1: Pares de salida mF y mC para caso de 1 signo distinto.

	$mF_7$	$mF_4$	$mF_1$	$mF_8$	$mF_5$	$mF_2$	$mF_9$	$mF_6$	$mF_3$
$mC_1$	-1 -1	3.36 -0.27	7.72 0.45	-0.72 -0.95	3.63 -0.22	8 0.5	-0.45 -0.90	3.90 -0.18	8.2 0.54
$mC_4$	-5.36 -0.72	-1 0	3.36 0.72	-5.09 -0.68	-0.72 0.045	3.63 0.77	-4.81 -0.63	-0.45 0.09	3.90 0.81
$mC_7$	-9.72 -0.4545	-5.36 0.27	-1 1	-9.45 -0.40	-5.09 0.31	-0.72 1.04	-9.18 -0.36	-4.81 0.36	-0.45 1.09
$mC_2$	-0.27 -1.04	4.09 -0.31	8.45 0.40	0 -1	4.36 -0.27	8.72 0.45	0.27 -0.95	4.63 -0.22	9 0.5
$mC_5$	-4.63 -0.77	-0.27 -0.045	4.09 0.68	-4.36 -0.72	0 0	4.36 0.72	-4.09 -0.68	0.27 0.045	4.63 0.77
$mC_8$	-9 -0.5	-4.63 0.22	-0.27 0.95	-8.72 -0.45	-4.36 0.27	0 1	-8.45 -0.40	-4.09 0.31	0.27 1.04
$mC_3$	0.45 -1.09	4.81 -0.36	9.18 0.36	0.72 -1.045	5.09 -0.31	9.45 0.40	1 -1	5.36 -0.27	9.72 0.45
$mC_6$	-3.90 -0.81	0.45 -0.09	4.81 0.63	-3.63 -0.77	0.72 -0.045	5.09 0.68	-3.36 -0.72	1 0	5.36 0.72
$mC_9$	-8.27 -0.54	-3.90 0.18	0.45 0.90	-8 -0.5	-3.63 0.22	0.72 0.95	-7.72 -0.45	-3.36 0.27	1 1

Observando estos valores pueden determinarse las manipulaciones que son plausibles mediante inspección. Se eligen aquellas que no ofrezcan un efecto desbalanceado en las variables de salida, es decir, que no den una salida muy grande para una muy pequeña en una de las otras y no provoque acoplamientos importantes. Utilizando este criterio se agregan a los pares de manipulaciones los que se muestran en la tabla 3.2.

### Caso de dos signos diferentes

Se aplicó la misma metodología anterior para el caso de dos signos diferentes para las ganancias mostradas en 3.39. Se observó que el patrón trazado en la tabla 3.2 se repite en este caso también, pero cambia la posición de las salidas de los ejes verticales

TABLA 3.2: Pares de manipulaciones permitidos.

	$mF_7$	$mF_4$	$mF_1$	$mF_8$	$mF_5$	$mF_2$	$mf_9$	$mF_6$	$mF_3$
$mC_1$	✓			✓			✓		
$mC_4$		✓			✓			✓	
$mC_7$			✓			✓			✓
$mC_2$	✓			✓			✓		
$mC_5$		✓			✓			✓	
$mC_8$			✓			✓			✓
$mC_3$	✓			✓			✓		
$mC_6$		✓			✓			✓	
$mC_9$			✓			✓			✓

y horizontales. El resumen de la información recopilada se presente en las tablas 3.3 y 3.4.

$$\mathbf{K} = \begin{bmatrix} 6.4 & -2.1 \\ 0.4 & -0.35 \end{bmatrix} \quad (3.39)$$

TABLA 3.3: Pares de salidas mF y mC para 2 signos.

	$mF_7$	$mF_4$	$mF_1$	$mF_8$	$mF_5$	$mF_2$	$mf_9$	$mF_6$	$mF_3$
$mC_7$	-1.0	8.6	18.2	-1.6	8.0	17.6	-2.2	7.4	17
	-1.0	0.6	2.2	-1.1	0.5	2.1	-1.2	0.4	2
$mC_4$	-10.6	-1.0	8.6	-11.2	-1.6	8	-11.8	-2.2	7.4
	-1.6	0	1.6	-1.7	-0.1	1.5	-1.8	-0.2	1.4
$mC_1$	-20.2	-10.6	-1	-20.8	-11.2	-1.6	-21.4	-11.8	-2.2
	-2.2	-0.6	1	-2.3	-0.7	0.9	-2.4	-0.8	0.8
$mC_8$	0.6	10.2	19.8	0	9.6	19.2	-0.6	9	18.6
	-0.9	0.7	2.3	-1	0.6	2.2	-1.1	0.5	2.1
$mC_5$	-9.0	0.6	10.2	-9.6	0	9.6	-10.2	-0.6	9
	-1.5	0.1	1.7	-1.6	0	1.6	-1.7	-0.1	1.5
$mC_2$	-18.6	-9	0.6	-19.2	-9.6	0	-19.8	-10.2	-0.6
	-2.1	0.5	1.1	-2.2	-0.6	1	-2.3	-0.7	0.9
$mC_9$	2.2	11.8	21.4	1.6	11.2	20.8	1	10.6	20.2
	-0.8	0.8	2.4	-0.9	0.7	2.31	-1	0.6	2.2
$mC_6$	-7.4	2.2	11.8	-8	1.6	11.2	-8.6	1	10.6
	-1.4	0.2	1.8	-1.5	0.1	1.7	-1.6	0	1.6
$mC_3$	-17	-7.4	2.2	-17.5	-8	1.6	-18.2	-8.6	1
	-2.0	-0.4	1.2	-2.1	-0.5	1.	-2.2	-0.6	1

TABLA 3.4: Pares de manipulaciones permitidos

	$mF_7$	$mF_4$	$mF_1$	$mF_8$	$mF_5$	$mF_2$	$mf_9$	$mF_6$	$mF_3$
$mC_7$	✓			✓			✓		
$mC_4$		✓			✓			✓	
$mC_1$			✓			✓			✓
$mC_8$	✓			✓			✓		
$mC_5$		✓			✓			✓	
$mC_2$			✓			✓			✓
$mC_9$	✓			✓			✓		
$mC_6$		✓			✓			✓	
$mC_3$			✓			✓			✓

### Caso sin signos

Para este caso se supusieron positivos todos los términos de ganancia de la función de transferencia como se muestra en la ecuación 3.40. Se repitió el procedimiento mostrado anteriormente para llegar finalmente a las manipulaciones permitidas para el proceso. Estos resultados aparecen en las tablas 3.5 y 3.6.

$$\mathbf{K} = \begin{bmatrix} 6.4 & 2.1 \\ 0.4 & 0.35 \end{bmatrix} \quad (3.40)$$

### 3.3.7. Máquina de inferencia

Si conocemos aproximadamente (es decir, en términos difusos) los valores de  $A_1, A_2, \dots, A_m$ , podremos saber aproximadamente el valor de B. Este es un mecanismo de inferencia extendido de la lógica clásica y se le conoce como Modus Ponens Generalizado. La regla mostrada en la ecuación 3.46 aplica esta estructura lógica.

Cuando trabaja un sistema difuso, es natural que más de una regla se dispare ante los valores nítidos de sus entradas. Esto es parte del concepto de difuso, ya que nada es enteramente de un solo tipo, sino que puede ser, en diferente grado, parte de varias categorías. Así, es necesario saber de que manera se van a combinar las reglas que se disparen en un momento determinado.

Existen dos métodos principales: la combinación de Göedel y la combinación de Mamdani. La combinación de Göedel consiste en juntar todas las reglas mediante alguna norma t para obtener una sola regla que se evaluará ante las entradas. La combinación de Mamdani hace lo contrario, evalúa cada regla por separado y combina los resultados mediante una norma s. Se ha demostrado que la combinación de Göedel no es práctica para aplicaciones en tiempo real dado que requiere gran capacidad de cálculo. Normalmente se utiliza, como es el caso de estos FKBC, la combinación de Mamdani. En 3.41 y 3.42 se definen la combinación de Mamdani y Göedel, respectivamente.

TABLA 3.5: Pares de salida mF y mC para caso sin signos

	$mF_7$	$mF_4$	$mF_1$	$mF_8$	$mF_5$	$mF_2$	$mf_9$	$mF_6$	$mF_3$
$mC_3$	-1 -1	8.6 0.6	18.2 2.2	-1.6 -1.1	8 0.5	17.6 2.1	-2.2 -1.2	7.4 0.4	17 2
$mC_6$	-10.6 -1.6	-1 0	8.6 1.6	-11.2 -1.7	-1.6 -0.1	8 1.5	-11.8 -1.8	-2.2 -0.2	7.4 1.4
$mC_9$	-20.2 -2.2	-10.6 -0.6	-1 1	-20.8 -2.3	-11.2 -0.7	-1.6 0.9	-21.4 -2.4	-11.8 -0.8	-2.2 0.8
$mC_2$	0.6 -0.9	10.2 0.7	19.8 2.3	0 -1	9.6 0.6	19.2 2.2	-0.6 -1.1	9 0.5	18.6 2.1
$mC_5$	-9 -1.5	0.6 0.1	10.2 1.7	-9.6 -1.6	0 0	9.6 1.6	-10.2 -1.7	-0.6 -0.1	9 1.5
$mC_8$	-18.6 -2.1	-9 -0.5	0.6 1.1	-19.2 -2.2	-9.6 -0.6	0 1	-19.8 -2.3	-10.2 -0.7	-0.6 0.9
$mC_1$	2.2 -0.8	11.8 0.8	21.4 2.4	1.6 -0.9	11.2 0.7	20.8 2.3	1 -1	10.6 0.6	20.2 2.2
$mC_4$	-7.4 -1.4	2.2 0.2	11.8 1.8	-8 -1.5	1.6 0.1	11.2 1.7	-8.6 -1.6	1 0	10.6 1.6
$mC_7$	-17 -2	-7.4 -0.4	2.2 1.2	-17.6 -2.1	-8 -0.5	1.6 1.1	-18.2 -2.2	-8.6 -0.6	1 1

TABLA 3.6: Pares de manipulaciones permitidos, caso de todos los signos positivos

	$mF_7$	$mF_4$	$mF_1$	$mF_8$	$mF_5$	$mF_2$	$mf_9$	$mF_6$	$mF_3$
$mC_7$	√			√			√		
$mC_4$		√			√			√	
$mC_1$			√			√			√
$mC_8$	√			√			√		
$mC_5$		√			√			√	
$mC_2$			√			√			√
$mC_9$	√			√			√		
$mC_6$		√			√			√	
$mC_3$			√			√			√

$$Q_M = \bigcup_{i=1}^M Ru^{(i)} \quad (3.41)$$

$$Q_G = \bigcap_{i=1}^M Ru^{(i)} \quad (3.42)$$

donde Q representa la combinación de las reglas Ru, desde  $Ru^{(1)}$  hasta  $Ru^{(M)}$ .

Solo resta definir la manera en que se calculará la implicación denotada en la regla. La implicación es la operación que surge de procesar el enunciado IF...THEN. La implicación es denotada por la expresión

$$p \rightarrow q = (p \vee q) \wedge \bar{p} \quad (3.43)$$

dado que las aplicaciones de control son las que requieren la mayor velocidad posible, es sumamente conveniente utilizar la implicación de Mamdani con producto como norma t para simplificar lo más posible los cálculos. Siguiendo estos lineamientos la implicación queda simplificada a la expresión

$$p \rightarrow q = pq \quad (3.44)$$

tomando en cuenta todas estas consideraciones obtenemos que el controlador difuso puede ser expresado como el máximo de la multiplicación de todas las reglas por el valor de entrada. El valor de la función de pertenencia de salida B' puede conocerse tras realizar la operación

$$\mu_{B'}(y) = \max_{i=1}^M \left( \sup_x \left( \mu_{A'}(x) \prod_{i=1}^n \mu_{A'}(x_i) \mu_{B'}(y) \right) \right) \quad (3.45)$$

### 3.3.8. Base de reglas

La base de reglas consiste en el corazón de FKBC, dado que el conocimiento del experto traducido a estas reglas. Las reglas consisten en enunciados de la forma “Si . . . , luego entonces . . .” (IF... THEN). Los cuales se escriben de una manera general como prosigue

$$IF \ x_1 \text{ is } A'_1 \text{ AND } x_2 \text{ is } A'_2 \ \dots \ \text{AND } x_M \text{ is } A'_M \ \text{THEN } y \text{ is } B' \quad (3.46)$$

tomando en cuenta la existencia para tres conjuntos difusos de entrada, esto nos daría un total de 9 reglas, es decir, de nueve diferentes combinaciones de los conjuntos difusos a lo largo de la respuesta transitoria del proceso. Dado que son 2 entradas habría que evaluar el estado de una de ellas mientras la otra se mantiene constante.



Debido a que la segunda entrada también tiene nueve casos diferentes existirían un total de 81 combinaciones que se traducirían en reglas de la forma 3.46. Este resultado es concordante con el que brinda la ecuación 3.26. Una vez definido esto, es necesario determinar el conjunto de salida que será activado por cada uno de las reglas.

Debe de tomarse en cuenta que la base de reglas es el corazón del controlador difuso, y en ella se encierra la inteligencia que permitirá tanto lograr el control del sistema como eliminar el acoplamiento. Así, la elección de éstas es una parte a la que debe de prestarse atención.

Basado en las referencias [22] y la extensión realizada en las secciones anteriores, se pueden tener hasta 27 pares de conjuntos difusos con la que podemos dar una respuesta a los predicados de las reglas del FKC.

Para el caso del controlador multivariable, inicialmente se pensó que un enfoque basado en la intuición podría ser suficientemente bueno como para llenar la base de reglas, pero se llegó a la conclusión que esto no es necesariamente cierto, dado que como la dinámica de cada una de las funcione de transferencia puede ser muy distinta entre sí, por lo que pueden obtenerse resultados que parecen desafiar al “sentido común”. Por ello, lo que se hizo fue la asignación de las manipulaciones de salida en base al análisis anterior basado en las ganancias estáticas del proceso.

El criterio que se utilizó para aparear a las reglas con su predicado consistió en establecer una norma de semejanza entre las reglas y los valores a los que converge el proceso por aplicar cierta manipulación de las previamente seleccionadas. Esto no representa ningún inconveniente dado que es posible obtener nuevamente las manipulaciones mediante 3.33.

Para realizar esto es necesario establecer un punto de comparación entre ambas variables. El problema consiste en que mientras a los valores de manipulación o de salida se pueden manejar como nítidos, no se tiene esta propiedad para las reglas, dado que tienen un valor meramente lingüístico.

Con ello en mente se decidió asignar un valor numérico que reflejara el valor de cada uno de los conjuntos difusos implicados. Debe remarcarse que este valor es arbitrario y su objetivo es reflejar que tan “positivo” o “negativo” es el efecto de uno de los miembros de la regla difusa.

Para ello se tomó en cuenta tanto la magnitud de error como la derivada del mismo, y a cada par se les asignó el número mostrado en la tabla 3.7. En él se asignan a las combinaciones de los valores lingüísticos de Negativo (N), Cero (Z) y Positivo (P) una cantidad. De esta manera se pudieron determinar los 9 casos posibles con los que se puede realizar la comparación.

TABLA 3.7: Valores atribuidos a las reglas difusas

Regla	NN	NZ	NP	ZN	ZZ	ZP	PN	PZ	PP
Valor	-4	-3	-2	-1	0	1	2	3	4

Al elegir las cantidades de esta manera es más fácil realizar la comparación con los valores de salida del proceso. Por ejemplo, si decimos que el error de una variable es negativo y el error de la otra también es negativo, significa que ambas variables se encuentran por encima de su referencia y que los valores de manipulación deben disminuir la magnitud tanto de la temperatura como del nivel. Esto, en parte, también tiene un significado semántico, dado que tanto la regla expresada numéricamente como los valores de salida van a tener el mismo signo, y por ende, el mismo sentido. Por esta razón, la comparación entre ambas variables es posible. Por ejemplo, si tenemos la regla expresada como en la ecuación 3.47, obtendríamos en su lugar una regla como se muestra en la ecuación 3.48.

$$\begin{aligned} &IF \text{ EN is } N \text{ AND } dEN \text{ is } Z \text{ AND } ET \text{ is } EN \text{ AND } dET \text{ is } Z \\ &THEN \text{ mF is } mF_1 \text{ Z AND } mC \text{ is } mC_1 \end{aligned} \quad (3.47)$$

$$IF -3 \text{ AND } -3 \text{ THEN } -1 \text{ AND } -1 \quad (3.48)$$

Realizar la comparación es complicado en estos términos, dado que son dos cantidades diferentes las que se deben comparar. Se consideró conveniente expresar los pares de números de una manera que pudieran ser manejados fácilmente. Se pensó entonces en utilizar números complejos. De esta manera, sería posible obtener el vector unitario en el plano de los números reales e imaginarios. Este vector sería el que verdaderamente sería comparado. Puede desprejarse la magnitud en la mayoría de los casos porque se han elegido solo las manipulaciones plausibles, es decir, que el valor de cada una de ellas es igual o menor a la unidad. Retomando el ejemplo mostrado en 3.47 y en 3.48, las cantidad es que serían comparadas se escribirían de la manera siguiente

$$IF -3 \text{ AND } -3 \text{ THEN } -1 \text{ AND } -1$$

$$IF -3 - j3 \text{ THEN } -1 - j$$

$$IF 3\sqrt{2}\left(-\frac{\sqrt{2}}{2} - j\frac{\sqrt{2}}{2}\right) \text{ THEN } \sqrt{2}\left(-\frac{\sqrt{2}}{2} - j\frac{\sqrt{2}}{2}\right) \quad (3.49)$$

$$IF 3\sqrt{2} e^{j\frac{5\pi}{4}} \text{ THEN } \sqrt{2} e^{j\frac{5\pi}{4}}$$

$$IF \mathbf{R}_i \text{ THEN } \mathbf{Y}_j$$

Basándose en estos argumentos, se puede decir que la comparación de las reglas y las manipulaciones es posible. La asignación de las manipulaciones basadas en la comparación se puede realizar mediante la siguiente manera

$$IF \|\mathbf{R}_i - \mathbf{Y}_j\| < \epsilon \text{ THEN } \mathbf{U}_i = \mathbf{K}^{-1} \mathbf{Y}_j \quad (3.50)$$

donde la norma de comparación queda definida como

$$\|\mathbf{R}_i - \mathbf{Y}_j\| = \angle \mathbf{R}_i - \angle \mathbf{Y}_j \quad (3.51)$$

El único caso donde la magnitud debe ser tomada es cuando alguna de las cantidades introducidas es un número exclusivamente real o imaginario, dado que la fase puede dar valores repetidos. Para esos casos deben de seleccionarse los valores aplicando el primer criterio, y los candidatos restantes se obtendrán aplicando como norma

$$\|\mathbf{R}_i - \mathbf{Y}_j\| = \text{abs}(\mathbf{R}_i) - \text{abs}(\mathbf{Y}_j) \quad (3.52)$$

En base a estas ideas, se pudo realizar un programa que arrojó como salida las reglas del sistema. En los diagramas 3.7 y 3.8 se presentan las reglas para el caso de un signo distinto. Los casos de dos signos distintos aparecen en las figuras 3.9, 3.10 y las de todos los signos positivos, en las figuras 3.11 y 3.12. Este procedimiento puede ser utilizado para cualquier combinación de signos y cualquier matriz de ganancias estáticas de un proceso multivariable.

Para una representación simplificada de las reglas del controlador se utilizará el diagrama de la figura 3.2. Debe de tomarse en cuenta que el estado de una salida determinada es positivo, cero o negativo respecto a lo que denota su propio eje. Así aunque en la posición vertical el lado superior indica positivo y el inferior, lo contrario, esto no implica una contradicción; simplemente en esa región una salida es de un valor y las demás de algún otro posible. En esta figura se muestra como quedan ordenadas las reglas al acomodarlas bajo estas convenciones.

		e1										
		N	N	N	Z	Z	Z	P	P	P		
de1	N	1	4	7	28	31	34	55	58	61	N	
	N	2	5	8	29	32	35	56	59	62	Z	
	N	3	6	9	30	33	36	57	60	63	P	
	Z	10	13	16	37	40	43	64	67	70	N	
	Z	11	14	17	38	41	44	65	68	71	Z	
	Z	12	15	18	39	42	45	66	69	72	P	
	P	19	22	25	46	49	52	73	76	79	N	
	P	20	23	26	47	50	53	74	77	80	Z	
	P	21	24	27	48	51	54	75	78	81	P	
		N	Z	P	N	Z	P	N	Z	P		
		e2										

FIGURA 3.2: Diagrama para representar las reglas del controlador

### 3.3.9. Defuzificador

Como se ha dicho, el defuzificador se encarga de convertir los conjuntos difusos de salida en un valor nítido. Todo defuzificador debe de cumplir con los criterios de plausibilidad, simpleza de cálculo y continuidad. Por plausibilidad se entiende que el resultado concuerde con el sentido común y por continuidad, que el método de defuzificación no de grandes cambios del valor nítido  $y^*$  para pequeños cambios en el grado de pertenencia de los conjuntos difusos.

Existen varios tipos de defuzificadores, entre los cuales destacan principalmente el de centro de área, el de centro promedio y el de defuzificador máximo. El defuzificador de centro de área es la idea más básica de la cual se desarrollaron las demás. El defuzificador de centro de área toma el área total que es ocupada por los conjuntos difusos de salida resultantes del proceso de inferencia y calcula el centroide de la misma. Esto se muestra en la figura 3.3. La operación para obtener el centroide está definida mediante la ecuación 3.53.

$$y^* = \frac{\int_U y \mu(y) dy}{\int_U \mu(y) dy} \quad (3.53)$$

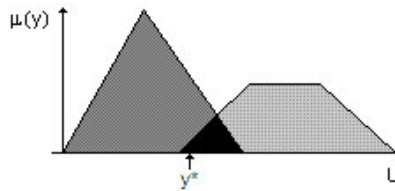


FIGURA 3.3: Fuzificador Centro de Gravedad

Este algoritmo es plausible y continuo, pero las integrales que lo conforman son lentas para evaluar en una computadora. Por ello la literatura propone también el método de centro promedio que será el utilizado en el diseño del controlador difuso. El método de máximo defuzificador no se contempló no cumple con el requerimiento de continuidad. El método de suma promedio hace la suma ponderada de los centros de cada función de pertenencia involucrada en la salida, como lo muestra la figura 3.4.

Se puede definir el método de suma promedio mediante la expresión 3.54.

$$y^* = \frac{\sum_{i=1}^M \bar{y}_i w_i}{\sum_{i=1}^M w_i} \quad (3.54)$$

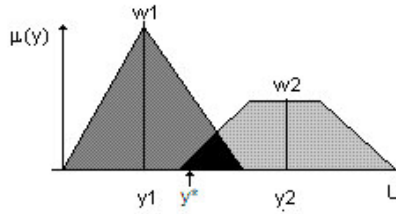


FIGURA 3.4: Fuzificador de centro promedio

### 3.4. Otras estructuras de controlador difuso

En la sección anterior se describió con detalle cada una de las partes del controlador difuso y se le dio un enfoque concreto a la estructura de un controlador difuso multivariable para dar al mismo tiempo una explicación de cada una de las partes de los elementos constitutivos. Sin embargo, esta no es la única estructura posible a utilizar, aunque probablemente sea la más obvia que viene al momento de pensar en un controlador difuso para esta aplicación.

Las principales desventajas que sufre esta estructura son que debe lidiar tanto con el control como con el desacoplamiento de las salidas, lo cual se encuentra implícito en su diseño como ya se ha descrito. La otra desventaja importante es que por su gran base de reglas, requiere mayor tiempo computacional del que podría requerir un controlador convencional.

Como respuesta a estas cuestiones, se decidió plantear otras estructuras más sencillas que permitan explorar un poco más a fondo las posibilidades de la lógica difusa en la ingeniería de control para procesos multivariables. Las técnicas presentadas se basan en la idea de que un desacoplador convencional tiene un muy buen desempeño, y por lo tanto, puede aprovecharse este bloque para constituir junto con él otras estrategias de control que pueden ser efectivas.

Los casos a analizar corresponden a controladores difusos univariados de 9 reglas para cada una de los lazos resultantes del sistema y la segunda es la idea de 3 sencillos controladores difusos que emulen al PID pero con mayor cantidad de grados de libertad.

#### 3.4.1. Controlador difuso univariable de 9 reglas

El controlador antes descrito está diseñado para encargarse tanto de las acciones de control como del acoplamiento del proceso multivariable. Para ello los conjuntos difusos fueron seleccionados bajo un criterio de estado estable y se arreglaron dentro de la base de reglas conforme a un criterio de magnitud.

Para poder llevar a cabo estas acciones, considerando los tres conjuntos difusos con los valores de NEGATIVO, CERO y POSITIVO para cada una de las entradas, se tuvo necesidad de llenar una base de 81 reglas en total.

De no contarse con efectos de acoplamiento, el problema podría ser simplificado al caso univariable, utilizando dos controladores de este tipo para poder colocar a las diferentes salidas en los valores de referencia deseados. Se pensó entonces que podía ser provechoso implementar no solo un controlador, sino una estrategia que utilizara controladores difusos univariables y desacopladores convencionales.

### Estructura del controlador difuso univariable de 9 reglas

El controlador difuso univariable, como es de esperarse, es mucho más sencillo que su contraparte de dos entradas y dos salidas. De utilizar tres conjuntos difusos, la base de reglas resultante es de 9 debido a que la cantidad de entradas ha sido reducida a la mitad. Como se ha elegido diseñar un controlador que tenga efecto PI, se consideró el integrador a la salida y la configuración de los conjuntos difusos para que el comportamiento de la manipulación fuera en modo velocidad. Dado que se espera que el controlador sea sintonizado mediante la técnica de optimización, no es necesario imponer un estricta sintonía inicial a los parámetros. Se consideró, como en el caso anterior, colocar los conjuntos difusos de entrada en un estado unitario, y a los conjuntos difusos de salida como fracciones uniformes de la unidad. En los cuadros 3.8 y 3.9 se resume las posiciones iniciales de los conjuntos. La columna que dice ganancia no debe de causar confusión. Solo existen 4 ganancias, dos para las respectivas entradas y dos para las respectivas salidas. Los valores aparecerán repetidos en los conjuntos correspondientes.

TABLA 3.8: Posiciones iniciales de conjuntos difusos de entrada de controlador difuso univariable.

Etiqueta	Valor	alfa $\alpha$	beta $\beta$	gamma $\gamma$	Ganancia
Error N	N	-1	1	0	1
Error Z	Z	-1	1	0	1
Error P	P	-1	0	1	1
Derror N	N	-1	1	0	1
Derror Z	Z	-1	1	0	1
Derror P	P	-1	0	1	1

La forma de colocar las reglas puede deducirse con ayuda de la figura 3.5. En ella se muestra el comportamiento típico de una variable mientras se aproxima a la referencia deseada. La gráfica determina las zonas en el que error es positivo o negativo, y en las que la derivada también cambia de valor. Las acciones a realizar ante estos valores se concentran en la tabla 3.10. En base a este razonamiento se enuncia la base de reglas que aparece en la tabla 3.11.

TABLA 3.9: Posiciones iniciales de conjuntos difusos de salida del controlador difuso univariable.

Etiqueta	Salida	Ganacia
m0	-1	1
m1	-0.75	1
m2	-0.5	1
m3	-0.25	1
m4	0	1
m5	0.25	1
m6	0.5	1
m7	0.75	1
m8	1	1

TABLA 3.10: Ley de control para procesos univariabes

Error	Derivada	Acción de control
Negativo	Negativo	Disminuir U
Negativo	Cero	Disminuir U
Negativo	Positivo	Disminuir U
Cero	Negativo	Disminuir U
Cero	Cero	Mantener U
Cero	Positivo	Aumentar U
Postivo	Negativo	Aumentar U
Postivo	Cero	Aumentar U
Postivo	Positivo	Aumentar U

TABLA 3.11: Base de reglas de controlador univariable

Error	Derror	Salida
N	N	m0
N	Z	m1
N	P	m2
Z	N	m3
Z	Z	m4
Z	P	m5
P	N	m6
P	Z	m7
P	P	m8

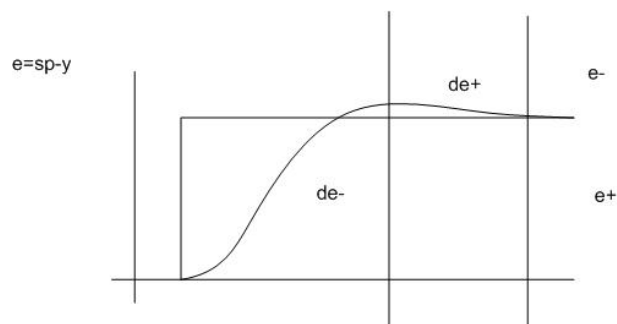


FIGURA 3.5: Comportamiento del error y su derivada durante la acción de control.

### 3.4.2. Controladores difusos univariables para emulación de PID

La idea de este controlador es sumamente sencilla. Basados en la información reportada por Viljamaa [21], la principal desventaja de los controladores con estructuras que buscan realizar un control tipo PI como las expuestas en los casos anteriores, es que son difíciles de ajustar porque los modos integral y proporcional se encuentran acoplados. Haciendo cita del mismo autor en [22] él asegura en su artículo que “ Aumentar la magnitud de las bases de los conjuntos difusos de la derivada entrada del error aumenta el modo integral. A su vez, aumentar la magnitud de las bases de los conjuntos difusos del error disminuye el modo proporcional pero aumenta el integral al mismo tiempo”.

La solución que propone Viljamaa en [21] es una nueva estructura donde se trabaje por separado el error proporcional y el error integral, es decir, al error y la derivada del mismo. De esta manera se tendrían dos controladores difusos correspondientes a cada acción en lugar de uno solo, más complejo, que realizara todas ellas al mismo tiempo.

En base a eso se decidió extender la misma idea para un controlador del tipo PID, solo que para poder conseguir el mismo efecto se requerirían de tres controladores que se encargaran de hacer cada una de las acciones de control. Al final cada una de las acciones sería sumada para obtener la manipulación final que iría al proceso.

Se presenta un esquema descriptivo del controlador en la figura 3.6.

#### Entradas y salidas de controlador difuso tipo PID

En este punto debe de pensarse en las entradas que puede tener tal controlador. En este caso se tienen dos posibilidades. La primera es tomar como entradas, el error, su primera y segunda derivadas respecto al tiempo; la segunda consiste en tomar, como en el caso del PID, el error, su derivada y su integral a lo largo del tiempo.

Esta decisión no era necesaria de tomar en los casos anteriores, dado que la necesidad de un integrador a la salida eliminaba la posibilidad de que se pudiera contemplar otro a la entrada del sistema. Esto puede verse con claridad si se consideran las ecuacio-



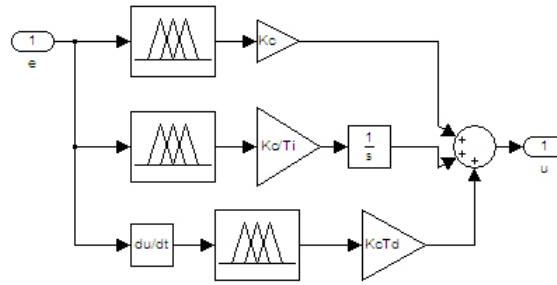


FIGURA 3.6: Estructura de Controlador Tipo PID difuso.

nes 3.22 y 3.23. Como puede verse, de tomarse la integral como entrada de los primeros casos esto resultaría en una alteración del modo de control que se pretende conseguir.

Sin embargo, en este caso es posible seguir el esquema fielmente, dado que se requiere de un integrador y este puede ser colocado en cualquier parte de la estructura del controlador. Podría objetarse que es preferible evitar colocar un integrador, de ser posible, pero comparando que las otras estructuras también lo requieren y ambos están sujetos al mismo riesgo de saturación, puede considerarse que este esquema es, por lo menos, no más propenso a saturación que el anterior.

El problema que puede resultar más grande de controlar la integral del error está relacionado con la variabilidad de este parámetro. Una posibilidad que podría elegirse es tomar como máximo la integral del error equivalente a no realizar ninguna acción durante el tiempo de evaluación. El problema es que, a diferencia de la magnitud del error, la intuición para determinar el máximo que puede tomar la integral no es una noción tan común. La idea básica para corregir este problema es que, suponiendo el caso de máxima linealidad, es más práctico fuzificar primero el error ponderado por la ganancia correspondiente e integrar después que hacerlo a la inversa. Es obvio que ante el caso de linealidad completa este orden es trivial, y es la no linealidad que surgirá al cambiar el orden y las posiciones de los conjuntos difusos de salida el potencial que hace interesante a esta estructura de controlador.

Por su parte, las ganancias del tiempo integral y derivativo es conveniente colocarlos a la salida de los controladores, con el fin de no perturbar el espacio normalizado en el que se desenvolverán los controladores difusos.

### Base de reglas, entradas y salidas del controlador tipo PID

Hasta el momento solo se ha descrito lo básico de las estructura, es decir, las entradas del sistema y como interactuarán entre sí los bloques. Ahora se discutirán cuestiones concretas como la cantidad y posición de conjuntos difusos así como su organización en las reglas del controlador.

Siguiendo las recomendaciones mencionadas en [6], para que un controlador difuso sea lo más lineal posible, sus conjuntos difusos, independientemente de su número, deben de repartirse uniformemente en el universo de discurso de tal manera que el corte entre ellos sea del 50%. Agregando más recomendaciones, se observó que para garantizar la linealidad debe de ser igual la cantidad de conjuntos difusos de entrada y salida, y colocar cada conjunto en la misma posición que su pareja del otro lado del controlador. De hacerse así no existirá diferencia entre la entrada y salida tal como si el sistema difuso no estuviera ahí, sin embargo, esto puede romperse mediante una variación de estas condiciones. Estas recomendaciones traen algunas consecuencias dignas de mención:

- Si el número de conjuntos de entrada es igual al número de conjuntos de salida y están en la misma posición esto significa que la base de reglas es directa, uniendo cada entrada con su salida respectiva.
- Otra ventaja de esto es que se elimina la necesidad de una implicación, dado que la operación queda hecha por la fuzificación y la defuzificación directamente.

Para el caso de este controlador, se utilizó una base de 9 conjuntos de entrada y 9 conjuntos difusos de salida con la mera intención de tener varios puntos de ajuste posibles. Esto aplicó para los tres bloques difusos del controlador.

### **Sintonización del controlador tipo PID**

Finalmente, descrito todo el controlador resta decidir como ajustar los parámetros. Debe decirse que de seguir todos los criterios antes mencionados el resultado es un PID que es posible ajustar el controlador con cualquier técnica que aplique para un PID convencional, porque, como ya se dijo anteriormente. Al ser completamente lineales los bloques difusos, su acción es transparente, restando solamente la estructura original.

Sin embargo, de ser así, esta estructura no ofrecería ventaja alguna. El potencial de lo propuesto es que la linealización puede ser rota para obtener respuestas que un PID normal no podría ofrecer. De alterar la posición de los conjuntos difusos de entrada o de salida se obtendría una respuesta distinta que puede ser mejor en un momento dado.

Tomando estas consideraciones, no se cree que se deba invertir más esfuerzo en sintonizar las ganancias del controlador que el que se utiliza para sintonizar un PID. De tener un juego de parámetros de satisfactorio desempeño al alcance de la mano, lo más conveniente es aplicar esa misma sintonización. El verdadero ajuste convendría en variar la posición de los conjuntos difusos de entrada o los de salida. Para simplificar el problema lo más recomendable sería variar solo un tipo de ellos a la vez, al menos hasta estar conscientes de las consecuencias específicas de tales acciones.

En este trabajo se analizarán las posibilidades que ofrece la flexibilidad de la estructura propuesta buscando mejoras a sintonías de PID proporcionadas como punto de partida. El estado inicial de los sistema difusos del PID difuso se presentan en la tabla 3.12.

TABLA 3.12: Configuración inicial de PID difuso. Posiciones de centros de entradas y salidas.

Etiqueta	C.Entrada	Salida
m0	-1	-1
m1	-0.75	-0.75
m2	-0.5	-0.5
m3	-0.25	-0.25
m4	0	0
m5	0.25	0.25
m6	0.5	0.5
m7	0.75	0.75
m8	1	1

### 3.5. Conclusión

A lo largo de este capítulo se hizo un compendio de lo que es lógica difusa y se explicaron las estructuras de los controladores que se diseñaron en base a esta técnica.

Las estructuras propuestas consisten en un controlador difuso multivariable que se encarga de combatir el acoplamiento y ejercer la acción de control, un par de controladores difusos univariables que forman una estrategia con ayuda de desacopladores y el controlador PID difuso apoyado de igual manera en el bloque desacoplador. Estas serán sometidas aún a modificaciones provocadas por el algoritmo de sintonización.

Se utilizarán las operaciones producto y máximo para la resolución de las normas  $t$  y normas  $s$  que requieran resolverse. Se utilizará la implicación de Mamdani y la combinación de Mamdani para obtener el resultado de salida. Este valor será defuzificado mediante el uso del método de centro promedio debido a sus características de plausibilidad equilibrada con el tiempo computacional que demanda.

Para las bases de reglas, se definió una metodología para acomodar las reglas y los conjuntos difusos de salida, para el caso del controlador multivariable. Para el caso de los controladores univariables y el PID difuso, la definición de las reglas se redujo a una caso trivial debido a la manera en que se seleccionaron la cantidad de conjuntos y sus relaciones entre ellos.

El resto de los ajustes necesarios dependerá de los resultados de la estrategia de optimización.

e1											
N N N			Z Z Z			P P P					
de1	N	6	6	0	7	6	0	2	2	8	N
	N	6	3	0	7	3	1	2	5	8	Z
	N	6	3	0	6	0	1	2	8	8	P
	Z	6	6	0	7	7	1	2	5	8	N
	Z	6	3	0	7	4	1	2	5	8	Z
	Z	6	3	0	7	1	1	2	5	8	P
	P	6	6	0	7	8	2	2	5	8	N
	P	6	3	0	7	5	1	2	5	8	Z
	P	6	0	0	8	2	1	2	5	8	P
N Z P			N Z P			N Z P					
e2											

FIGURA 3.7: Reglas para el caso de 1 signo. Salida 1.

e1											
N N N			Z Z Z			P P P					
de1	N	0	0	6	1	0	6	2	2	8	N
	N	0	3	6	1	3	7	2	5	8	Z
	N	0	3	6	0	6	7	2	8	8	P
	Z	0	0	6	1	1	7	2	5	8	N
	Z	0	3	6	1	4	7	2	5	8	Z
	Z	0	3	6	1	7	7	2	5	8	P
	P	0	0	6	1	2	8	2	5	8	N
	P	0	3	6	1	5	7	2	5	8	Z
	P	0	6	6	2	8	7	2	5	8	P
N Z P			N Z P			N Z P					
e2											

FIGURA 3.8: Reglas para el caso de 1 signo. Salida 2.

e1												
N N N			Z Z Z			P P P						
de1	N	6	6	0	7	6	0	8	8	2	N	de2
	N	6	3	0	7	3	1	8	5	2	Z	
	N	6	3	0	6	0	1	8	2	2	P	
	Z	6	6	0	7	7	1	8	5	2	N	
	Z	6	3	0	7	4	1	8	5	2	Z	
	Z	6	3	0	7	1	1	8	5	2	P	
	P	6	6	0	7	8	2	8	5	2	N	
	P	6	3	0	7	5	1	8	5	2	Z	
	P	6	0	0	8	2	1	8	5	2	P	
N Z P			N Z P			N Z P						
e2												

FIGURA 3.9: Reglas para el caso de 2 signo. Salida 1.

e1												
N N N			Z Z Z			P P P						
de1	N	6	6	0	7	6	0	8	8	2	N	de2
	N	6	3	0	7	3	1	8	5	2	Z	
	N	6	3	0	6	0	1	8	2	2	P	
	Z	6	6	0	7	7	1	8	5	2	N	
	Z	6	3	0	7	4	1	8	5	2	Z	
	Z	6	3	0	7	1	1	8	5	2	P	
	P	6	6	0	7	8	2	8	5	2	N	
	P	6	3	0	7	5	1	8	5	2	Z	
	P	6	0	0	8	2	1	8	5	2	P	
N Z P			N Z P			N Z P						
e2												

FIGURA 3.10: Reglas para el caso de 2 signo. Salida 2.

e1												
N N N			Z Z Z			P P P						
de1	N	6	6	0	7	6	0	8	8	2	N	de2
	N	6	3	0	7	3	1	8	5	2	Z	
	N	6	3	0	6	0	1	8	2	2	P	
	Z	6	6	0	7	7	1	8	5	2	N	
	Z	6	3	0	7	4	1	8	5	2	Z	
	Z	6	3	0	7	1	1	8	5	2	P	
	P	6	6	0	7	8	2	8	5	2	N	
	P	6	3	0	7	5	1	8	5	2	Z	
	P	6	0	0	8	2	1	8	5	2	P	
N Z P			N Z P			N Z P						
e2												

FIGURA 3.11: Reglas para el caso sin signo. Salida 1.

e1												
N N N			Z Z Z			P P P						
de1	N	2	2	8	1	2	8	0	0	6	N	de2
	N	2	5	8	1	5	7	0	3	6	Z	
	N	2	5	8	2	8	7	0	6	6	P	
	Z	2	2	8	1	1	7	0	3	6	N	
	Z	2	5	8	1	4	7	0	3	6	Z	
	Z	2	5	8	1	7	7	0	3	6	P	
	P	2	2	8	1	0	6	0	3	6	N	
	P	2	5	8	1	3	7	0	3	6	Z	
	P	2	8	8	0	6	7	0	3	6	P	
N Z P			N Z P			N Z P						
e2												

FIGURA 3.12: Reglas para el caso sin signo. Salida 2.

## Capítulo 4

# Estrategia de Sintonización

Una vez que la estructura del controlador ha sido definida por completo, el siguiente paso a seguir es el ajuste de los parámetros. Como es el caso de los controladores convencionales, una selección cuidadosa de las variables del sistema de control repercutirán directamente en su desempeño.

En general, para poder controlar un proceso es indispensable tener información de él. En el caso de los controladores difusos, donde se busca capturar el conocimiento humano en un algoritmo de manipulación, se supone la existencia de un experto que provee de la información necesaria para determinar los valores del sistema así como sirvió para determinar la estructura del mismo.

Como la ausencia de dicho experto es un problema bastante frecuente, es necesario hacer uso de otros recursos para obtener la información necesaria para sintonizar el controlador. En caso de los controladores convencionales, existe toda una teoría de por medio que ofrece cierta variedad de técnicas, métodos y heurísticas que permitirán el diseño para un caso particular.

En el caso del controlador difuso, aunque es cierto que los ajustes que se hagan sobre él deben tomar en cuenta tanto la naturaleza física del proceso como sus limitaciones, puede no existir una base determinística sólida para un caso específico o puede no existir una única configuración que nos permita obtener una buena respuesta. Esto ha llevado a que los diseños basados en lógica difusa normalmente se ajusten (basados necesariamente en el conocimiento de la dinámica del proceso) bajo el método de prueba y error.

Debido a que uno de los objetivos del trabajo presente es el de lograr obtener un algoritmo de sintonía del controlador difuso, es necesario un procedimiento que pueda ser automatizado para minimizar en él la intervención del usuario.

### 4.1. El problema de optimización

Una manera de atacar este problema es verlo como si fuera un problema de optimización. Las ventajas que ofrece este enfoque consisten en la existencia toda una

teoría de control óptimo que puede ser consultada. Se tiene la oportunidad de abstraer el problema para resolverlo sin tener que tomar en cuenta la naturaleza particular del proceso y de utilizar la gran disponibilidad de algoritmos para optimizar una función de costo existentes.

En los objetivos se tiene el evaluar el desempeño del controlador mediante parámetros tales como la suma de errores absolutos de la variable de proceso controlada, la suma de errores absolutos de la variable acoplada, el sobretiro y el tiempo de establecimiento. Éstos pueden convertirse en las restricciones de una función objetivo que forzarán al controlador difuso a modificar sus parámetros de tal manera que la respuesta del proceso cumpla con tales condicionamientos.

En los inicios de la investigación se había planteado el uso de algoritmos basados en el gradiente de la función para encontrar el estado de mínimo costo, es decir, aquella combinación de parámetros del controlador difuso que ofrecerían la mejor respuesta.

Tras hacer una mayor investigación [7] y algunas pruebas se concluyó que dicho algoritmo no podía ofrecer los resultados que se pretendía obtener, y esto se debe a la naturaleza que tiene esta clase de métodos a encontrar un mínimo local y permanecer en él. La única alternativa posible con tal algoritmo consistiría en repetirlo varias veces en puntos aleatorios y suficientemente espaciados del dominio de la función y guardar la configuración menor, o bien, realizar una búsqueda exhaustiva por sectores a diferentes niveles de resolución hasta encontrar un óptimo global [3].

Dado que el tiempo computacional es también un factor a considerar, en especial porque la cantidad de parámetros y operaciones de ajuste aumentan directamente con la cantidad de salidas del sistema, se buscó una técnica que pudiera manejar el problema de los mínimos locales y que lo consiguiera con un número menor de iteraciones. Debido a estas razones se eligió aplicar la técnica del recocido simulado para la sintonización del controlador.

## 4.2. Recocido Simulado

La técnica del recocido simulado fue introducida en los años ochenta como una opción heurística para la solución a problemas combinatorios de gran escala. Así, se reportó en la literatura la manera en que el recocido simulado puede ser utilizado para resolver problemas como la colocación de los circuitos integrados en una tarjeta impresa de tal manera que tanto el cableado como la interferencia entre los circuitos fuera la mínima, o también la forma en que podía resolverse el problema del viajero para cientos o incluso miles de ciudades [7].

El algoritmo del recocido simulado está basado en la analogía que existe entre la manera en que los átomos de un material fundido se van acomodando hasta formar un cristal mediante un proceso de enfriamiento y la minimización de una función de opti-



mización. El algoritmo simula tal proceso a nivel microscópico variando un parámetro al que se le conoce como temperatura para obtener estados del sistema que conllevarán al mínimo costo ante las restricciones presentadas.

La base del algoritmo consiste en la presencia de un ciclo interno y de un ciclo externo. En el ciclo interno se van haciendo variaciones aleatorias al sistema y se evalúa el costo de la nueva configuración. Debe notarse que esto es análogo al cambio en la energía del material fundido debido a un nuevo acomodo de sus átomos. En caso de que el costo del nuevo estado sea menor al del estado anterior ésta se aceptará sin discusión. En caso de que el nuevo estado genere un mayor costo, el aceptar o rechazar la nueva configuración se hará probabilísticamente mediante la expresión

- Si  $\Delta J \leq 0$ , aceptar el nuevo estado
- Si  $\Delta J > 0$ , aceptar el nuevo estado si

$$R \in [0, 1] < \exp\left(\frac{-\Delta J}{T}\right) \quad (4.1)$$

donde 4.1 es la ecuación del algoritmo de Metrópolis,  $\Delta J$  Es el cambio en la función de costo,  $T$  es el parámetro de temperatura y  $R$  es un número aleatorio entre cero y uno. Cuando un estado es aceptado se tomará como un nuevo punto de referencia para buscar el siguiente.

Para cuestiones del algoritmo, el parámetro  $J$  representa el costo de mantener al sistema en un estado determinado. Este costo será una función íntimamente relacionada a las variables del controlador difuso como pueden ser, por ejemplo, la distribución y anchura de las funciones de pertenencia. Debe observarse que  $J$  aumentará en la medida la respuesta del sistema se desvíe de las directrices marcadas por los condicionamientos representados por los parámetros de desempeño.

El ciclo externo del algoritmo consiste en la manera en que la temperatura del proceso irá variando a lo largo del tiempo. Fuera del contexto de la fundición de materiales, el parámetro de temperatura carece de un significado físico, por lo tanto, cuando se habla refiriéndose al recocido simulado, solo puede decirse que es un parámetro de control que tiene las mismas unidades que posee la función de costo.

Es notable como la distribución de probabilidad, con una temperatura suficientemente elevada, aceptará en un principio muchos saltos hacia estados de mayor costo. Sin embargo, conforme el sistema se va enfriando y perdiendo energía, la probabilidad de aceptar estados de mayor costo será cada vez menor, llevando a la larga al algoritmo a encontrar un estado final que se encuentra “cuesta abajo” en el hiperespacio de trabajo. Otro aspecto importante es que independientemente de la temperatura, si el algoritmo cae en un punto donde las variaciones en el costo sean muy pequeñas, inmediatamente permitirá estados de mayor costo. Esta característica es la que previene

al algoritmo de quedar atrapado en mínimos locales, dado que al encontrarse con un mínimo buscará salir de él. En teoría, solo el mínimo global será un estado de tan baja energía que a pesar de los saltos que conduzcan fuera de él regresarán a la larga hacia ese mismo punto.

#### 4.2.1. Aspectos prácticos del algoritmo

Para implementar el algoritmo, es necesario considerar 4 aspectos: [20]

1. Configuración del problema.
2. Definición de vecindario.
3. Una función de costo.
4. Calendarización del enfriamiento.

#### Configuración del problema

Respecto a la configuración del problema, se tiene el lazo de control multivariable mostrado anteriormente. En este lazo de control se tendrá el modelo del proceso a controlar conectado con la estructura del controlador difuso. Para evaluar la función de costo se realizará una prueba escalón para cada entrada, el cual tendrá un tiempo máximo de cuatro veces la mayor constante de tiempo del modelo. Este criterio de paro es importante dado que la mayor parte del tiempo computacional consiste en obtener el costo del estado actual. Se realizará la prueba escalón para cada entrada manteniendo las demás constantes y se observará los efectos en cada una de las salidas. En base a la respuesta del sistema se podrá calcular la función de costo, dando vida así al algoritmo.

Los cambios en el estado se lograrán mediante variaciones aleatorias independientes en los parámetros de configuración del controlador, que consistirán en los tres parámetros de cada una de las funciones de membresía de entrada, así como una ganancia que afectará a cada entrada del sistema. En este caso no se cuenta como entrada las derivadas de las variables de entrada, dado que dicha operación se considerará dentro del algoritmo de control. Para el caso de la salida habrá una sola ganancia para todas las señales.

El efecto de las ganancias de entrada y salida consisten en afectar a todos los conjuntos de manera uniforme, lo cual puede entenderse como un ajuste a groso modo. Por otra parte, la configuración individual de cada conjunto difuso consistirá en un ajuste fino al desempeño del controlador. En el caso de las salidas existe una sola ganancia debido a que de utilizar diferentes cantidades se estarían afectando las propiedades desacoplates que el diseño pretende ofrecer.

Para el procedimiento de optimización se correrán varias pruebas hasta obtener el exponente que ostente el menor costo para cada una de las estrategias a sintonizar. Esto es útil y factible dado que por ser el Recocido Simulado un algoritmo no determinístico, es posible obtener diferentes soluciones con cada ejecución que se realice. Para garantizar el carácter aleatorio del algoritmo, en el programa de aplicación se generan semillas diferentes en cada ejecución. Para conocer más acerca de la aplicación desarrollada, debe referirse al apéndice C del presente documento.

### Definición de Vecindario

Por vecindario se entiende una delimitación de los estados en los que el sistema se puede encontrar. Como la técnica del recocido simulado estaba pensada para resolver problemas combinatorios, diseñar el vecindario se reducía al diseño de un diagrama de estados, al cual normalmente se le conoce como grafo. En el caso del problema del viajero, el diseño de este gráfico es relativamente sencillo puesto que cada estado corresponde a cada una de las ciudades involucradas en el circuito y el costo en la distancia que existe de viajar de una ciudad a la otra que trazara el itinerario.

En el caso del ajuste del controlador difuso, no es posible delimitar de esta manera el vecindario de operación. Sin embargo, es posible establecer límites a la variación de los parámetros así como la magnitud en la que se podrá variar los cambios aleatorios del sistema para alcanzar un estado nuevo.

Respecto a lo que se entiende como las limitaciones del vecindario, podemos decir que la base de las funciones de pertenencia pueden ser de cualquier tamaño excepto cero. Asimismo se limitan que las ganancias de normalización no valieran cero, dado que esto daría como resultado que el sistema no controlara en absoluto. Como es un tiempo limitado en el que se realizan las simulaciones, la condición de no control produce un error finito que incluso puede llegar a convertirse en un mínimo local. Sin embargo, pueden tomarse criterios para que los valores no oscilen en un rango infinito. Un criterio que puede utilizarse sería hacer variar los valores entre cero y el valor máximo de ganancia última de las procesos de la planta multivariable que quedan en retroalimentación dentro de los lazos de control. Otra regla de dedo que puede recomendarse es determinar un rango de 0 a 100. En base al valor de ganancia del proceso, puede obtenerse una noción si este rango es demasiado amplio o es necesario aumentarlo.

Respecto a lo que representa el salto de cada estado, se determinó que un número aleatorio entre  $\pm 1$  era suficiente como para hacer funcionar el algoritmo. Además, se siguió la recomendación consultada en la bibliografía [20] para que el rango de perturbación se fuera reduciendo como una función de la temperatura del material fundido. La expresión que describe a la función de delimitación es

$$\epsilon = \epsilon_{max} \left( \frac{\log(T - T_f)}{\log(T_0 - T_f)} \right) \quad (4.2)$$

En este caso, el cambio máximo  $\epsilon_{mas}$  será de uno. El valor de  $\epsilon$  servirá para determinar el rango máximo en el que variarán los números aleatorios que se utilizarán para generar las perturbaciones y para comparar respecto a la distribución de Boltzmann en el ciclo interno del algoritmo.

## Función de costo

La función de costo es de las partes medulares del algoritmo porque éste tratará de modificar los parámetros con el objetivo de hacer de ésta un valor mínimo.

Como se mencionó anteriormente, el costo será calculado mediante una prueba escalón donde se medirán los parámetros con los que se cuantificará la calidad de la respuesta obtenida. La función de optimización debe de ser planteada con cuidado para que obtenga soluciones que sean las deseadas y no casos que conlleven a estados adversos como el de no control o el de oscilación constante. Cabe mencionar que estos estados pueden convertirse en mínimos locales debido a que el tiempo de simulación de cada prueba es finito, de no ser así, éstos serían fácilmente rechazados por el algoritmo de selección.

Para las pruebas realizadas se utilizó una función de optimización que consideraba los factores siguientes:

- La suma de errores absolutos en la variable de salida a controlar, uno de los principales parámetros de desempeño, denotado como  $\sum |e_i(k)|$ .
- El sobretiro de la variable de salida a controlar. Es definido de la forma

$$Mp = -\text{mín}(y_{ref} - y(k)) \quad (4.3)$$

- El tiempo de establecimiento, tomando un valor máximo posible al de cuatro veces la mayor constante de tiempo del modelo. Se indicará como  $t_i$ .
- La suavidad de la manipulación, la cuál es importante para evitar que el algoritmo termine por dar como solución un estado de oscilación constante. Está definido como

$$S_U = \sum |u(k) - u(k - 1)| \quad (4.4)$$

- El valor de error final que se alcanzó al término del experimento, condición impuesta para garantizar que el sistema ejerza una manipulación y no opte por dar respuesta alguna.
- La suma de errores absolutos de la variable acoplada, denotados como  $\sum |e_m(k)|$ . Para una salida  $i$  de  $N$  totales que se están evaluando,  $\sum |e_m(k)|$  representa la suma de errores de las otras  $m$  salidas que no son  $i$ .

Debido a que las cantidades generadas por cada uno de estos aspectos no tienen porque ser del mismo orden de magnitud, el uso de constantes de ponderación es necesario para otorgar a cada uno la importancia que tiene en la calidad de la respuesta del sistema. Basados en la idea para el cálculo del error global de un instrumento de medición a partir de las imprecisiones individuales de cada uno de sus componentes, se pensó que una buena opción para ponderar los pesos era que al inicio todos aportaran la misma importancia al error total.[8] Este ajuste lo ejecuta el algoritmo automáticamente, realizando una estimación del costo en el estado inicial y salvando las aportaciones individuales. En base a esta información es sencillo calcular el valor de las ganancias de ponderación. Siendo  $K$  la ganancia de ponderación, esto se puede expresar como procede

$$K_i = \frac{J_N}{J_i} \quad (4.5)$$

Tras hacer algunas pruebas, se concluyó aumentar el valor de las ganancias calculadas por el algoritmo para los parámetros de tiempo de establecimiento y suavidad de la manipulación, esto con el fin de eliminar la posibilidad de que el resultado final fuera una situación mínima pero impráctica en términos de control del proceso. Estas ganancias se aumentaron en un factor de 10.

En resumen, la función de optimización  $J$  para  $N$  salidas se puede expresar como en la ecuación 4.6, donde el subíndice  $i$  denota a cada una de las salidas del sistema y el subíndice  $m$ , las otras salidas que son diferentes a  $i$ .

$$J = \sum_{i=1,2,\dots,N} \left( K_{1+N_i} \sum |e_i(k)| + K_{2+N_i} M p_i + K_{3+N_i} t_i + K_{4+N_i} S U_i + \right. \\ \left. + K_{5+N_i} \sum |e_m(k)| + K_{6+N_i} |e_i(N)| \right) \quad (4.6)$$

Otra posibilidad es tomar solamente los máximos de los indicadores de todas las entradas a la vez, esto haría que la función de costo quedara definida de la siguiente manera

$$J = \sum_{i=1,2,\dots,N} \left( K_1 \text{máx}(|e_i(k)|) + K_2 \text{máx}(M p_i) + K_3 \text{máx}(t_i) + \right. \\ \left. + K_4 \text{máx}(S U_i) + K_5 \text{máx}(|e_i(N)|) \right) \quad (4.7)$$

## Calendarización del Enfriamiento

El ciclo externo del algoritmo del recocido simulado consiste precisamente en la calendarización del enfriamiento, es decir, de que manera irá variando este parámetro

a lo largo del procedimiento. Es fácil de suponer que entre más lentamente se realice el cambio en la temperatura mayores posibilidades tendrá el algoritmo de revisar todo el espacio de soluciones y encontrar la ruta hacia la configuración óptima.

Siguiendo las recomendaciones de la bibliografía [20], se hizo que la temperatura  $T_j$  varíe linealmente entre un valor inicial  $T_0$  y un valor final  $T_f$

$$T_{j+1} = \alpha T_j \quad (4.8)$$

La selección de los valores  $T_f$ ,  $T_0$  debe realizarse con cierto cuidado. Idealmente,  $T_f$  debe ser lo suficientemente alta como para dar al algoritmo la energía para visitar todo el espacio de soluciones, por su parte,  $T_0$  debe de ser lo suficientemente baja para que los saltos de estado sean lo suficientemente pequeños como se denota en 4.2. La selección puede realizarse a prueba y error. En el caso de este trabajo, se siguió la recomendación en [20] en la cual se establece que la temperatura superior debe de permitir un nivel de aceptación del 80%, es decir, que al inicio del algoritmo por lo menos este porcentaje de estados generados fuera aceptado. Para estimar su valor se realiza una perturbación aleatoria al proceso y se calcula la función de costo como en 4.6 e igual se aplica al estado inicial. Solo resta despejar de la distribución de Boltzmann para obtener el valor de temperatura como

$$T_0 = \begin{cases} \frac{\Delta J}{\ln(0,8)}, & \text{si } T_0 > 10000 \\ 10000, & \text{si } T_0 < 10000 \end{cases} \quad (4.9)$$

Para  $T_f$  se dio un valor de 25, haciendo a la analogía que éste corresponde a la temperatura ambiente. El valor de 10000 para  $T_0$  se obtuvo empíricamente tras las pruebas realizadas en el trabajo.

Para el caso del ciclo interno, lo importante es determinar la cantidad de estados que se evaluarán a un nivel determinado de temperatura. Para garantizar que a bajos rangos aumente la cantidad de estados puede usarse la siguiente expresión [20]. Esto se hace con el fin de lograr el cuasi equilibrio del sistema [9].

$$M = N_f \left[ 2 + 8 \left( 1 - \frac{\log(T - T_f)}{\log(T_0 - T_f)} \right) \right] \quad (4.10)$$

donde  $N_f$  Representan los grados de libertad del sistema, los cuales se pueden obtener a partir de la cantidad de iteraciones que se desean hacer del ciclo externo. Otra posibilidad consiste en hacer los grados de libertad iguales al producto del número de puntos de interés por la cantidad de números aleatorios que se generan para calcular la perturbación en cada punto.[16]

### 4.2.2. Algoritmo de sintonización

Con los aspectos anteriores ya mencionados, así como la estructura del controlador difuso y la manera en que se van a diseñar la base de reglas, es posible describir un algoritmo de sintonía basado en toda la información recabada. El algoritmo puede ser programado en una computadora para librar al usuario de la tarea del diseño del controlador.

En general, el algoritmo de sintonía se puede resumir en los siguientes pasos:

1. Identificación del proceso multivariable.
2. Diseño del controlador difuso mediante los lineamientos mencionados y el criterio de norma (Para el caso multivariable) o las reglas convenidas para las otras estructuras de control.
3. Aplicación del algoritmo del recocido simulado para realizar el óptimo ajuste de los parámetros.
4. Elegir el mejor candidato que será aquél que obtenga menor valor al evaluar la función de costo.

La justificación de los tres pasos consiste en lo siguiente.

- Para controlar un proceso es indispensable tener información de él. En una situación ideal en la que se aplicara el control difuso, se contaría con la información de un experto quien podría capturar su conocimiento en la base de reglas. Eso no siempre es posible y en caso de un algoritmo de sintonía eso es inoperable, dado este conocimiento solo aplicaría a una dinámica en particular. Por lo tanto, la manera más viable de obtener ese conocimiento que no se tiene acerca del proceso a controlar es mediante una prueba de identificación.
- Aunque el algoritmo del recocido podría configurarse para el diseño completo del controlador (determinación de conjuntos difusos de entrada, salida, etc), se ha considerado que un cierto grado de estructura es deseable para disminuir la complejidad del problema, lo cual se traducirá en menor esfuerzo computacional para su determinación. Para efectos de lo realizado, se utilizó entradas del error, derivada del error, cada una con tres conjuntos difusos para la variable. Las características de cada estrategia de control están definidas en el capítulo correspondiente.
- Siendo extremistas respecto al punto anterior, podría pensarse que es posible determinar un buen controlador en base a una ordenada e inteligente selección de parámetros basado en uno o varios criterios o heurísticas. Esto es efectivamente

correcto, pero a menos de que se haga un profundo análisis del comportamiento del controlador difuso, lo más que se pueden establecer son reglas de dedo o aproximaciones. La ventaja de la aplicación del algoritmo de recocido simulado es que libera de la necesidad de hacer ese análisis exhaustivo y garantiza de que se está trabajando con una sintonización óptima en función a las restricciones del problema, algo que con otros métodos no es necesariamente posible de determinar.

- Otra de las ventajas que nos ofrece el uso del recocido simulado es la existencia de una medida de la calidad de la respuesta mediante la cual podemos hacer comparaciones con otras técnicas. Así, la función de optimización y el análisis de su comportamiento nos puede brindar información sobre el desempeño del controlador y la estructura del mismo.
- Una ventaja más del enfoque propuesto es que los tres pasos son posibles de programar para formar en conjunto un algoritmo automático de sintonización de controladores difusos. En la literatura [18] se ha publicado un algoritmo que puede identificar procesos mediante el uso de mínimos cuadrados de manera automática.

Lo que se hará a continuación consiste en la descripción de algoritmos para programar cada uno de las etapas ya ennumeradas, mostrando sus diagramas de flujo y dando una breve explicación de cada una de ellas. Se presentarán y explicarán los algoritmos para el diseño del controlador y la sintonización de los parámetros por vía del recocido simulado.

En el diagrama de la figura 4.1 se muestra la manera en que trabaja el algoritmo. Primeramente se inicializan las variables tales como las temperaturas de operación así como determinar el costo del estado inicial para correr el método a partir de ese punto. En el diagrama de flujo, las variables  $J$  representan el valor del costo,  $Z$ , el estado o configuración del controlador difuso.  $M$  es la cantidad de iteraciones del ciclo interno;  $T$ , la temperatura y  $\alpha$  será el parámetro de calendarización del ciclo externo. El subíndice 1 representa a las variables relacionadas al estado considerado como óptimo y el subíndice 2, al del nuevo estado que se propone como el mínimo.

La función `CalcularIteraciones()` se encarga de determinar la cantidad de veces que se ejecutará el ciclo interno en función de los valores de temperatura inicial, final y actual. La función `CalcularPert()` tiene el objetivo obtener la máxima magnitud de las perturbaciones aleatorias que se utilizarán durante una iteración del ciclo interno. `CalcularEstado()` será una función que en base a la perturbación máxima generará números aleatorios para modificar los parámetros del controlador difuso. En este caso, se generará un número aleatorio por cada parámetro y éste se sumará algebraicamente al valor actual para obtener la nueva configuración. La función `CalcularJ()` realiza la prueba escalón para cada entrada manteniendo las demás constantes y contabiliza el valor total de la función de costo. La función `ProB()` se encargará de evaluar la función de



Boltzmann para los valores de temperatura específicos.

Inmediatamente inicia el ciclo interno, el cual está controlado por la variable M, la cual va variando la cantidad de estados a evaluar en concordancia con la ecuación 4.10 . La perturbación se calcula del modo que lo indica la ecuación 4.2, generando un número aleatorio entre el rango indicado. En el ciclo interno se decide si se acepta el estado en función al costo, y en caso de ser mayor, se considerará la distribución de probabilidad de Boltzmann para llegar a una conclusión.

El ciclo se repite N veces hasta que terminan las iteraciones indicadas. Posteriormente se cambia la temperatura en concordancia con la ecuación 4.8 y se detiene cuando se ha alcanzado la temperatura final. Una variante del algoritmo presenta que el ciclo interno experimentará una temperatura que será proporcional a la función de costo, como se muestra en la ecuación 4.11. En este caso se toma el cambio cada vez que se acepta un nuevo estado. El aplicar una u otra forma no es crítico, pero puede modificar la administración de los recursos de cómputo.

$$T_{i+1} = T_i \frac{J2}{J1} \quad (4.11)$$

### 4.2.3. Ejemplo de aplicación del algoritmo

Suponiendo un proceso multivariable con la función de transferencia [22]

$$H(s) = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \quad (4.12)$$

Se desea crear un controlador difuso utilizando la metodología descrita.

A partir del análisis estático descrito en el capítulo de la estructura del controlador difuso, se pueden aplicar las ecuaciones correspondientes para determinar los conjuntos difusos de salida del controlador. El arreglo de estos, como ya se ha dicho, se elige para que el valor de estado estable suba o baje a una tasa unitaria mientras la otra lo hace del mismo modo o permanece sin cambio alguno. En la figura se muestran los conjuntos difusos generados por el algoritmo.

Haciendo uso del criterio de la norma de fase mencionado anteriormente, es posible determinar las reglas del controlador difuso. Como se explicó, en este método se les da a cada implicación un par de valores con los que se realiza la comparación respecto a los predicados. Para determinar la semejanza entre los pares de la implicación y los pares correspondientes a cada salida de estado estable de las manipulaciones puede establecerse una norma de magnitud o de fase para saber como acomodar cada una de las reglas de la base.

Aplicando este principio fue posible determinar la base de reglas, la cual se muestra en las figuras 4.3 y 4.4. Se utilizó para su representación el diagrama que se ha explicado

en capítulos anteriores.

Tras definir las salidas y las reglas mediante el análisis de ganancias estáticas se prosigue utilizando el recocido simulado para poder ajustar las ganancias de entradas y salidas así como las posiciones de los conjuntos difusos en las entradas. Una vez aplicado el recocido simulado sobre este proceso, se obtiene un juego de parámetros tal que genera el valor mínimo de  $J=733$ ; este valor fue calculado mediante la ecuación 4.6. La posición de los conjuntos difusos de entrada y las ganancias se muestra a continuación. Para este caso en particular, el algoritmo del recocido simulado encontró las ganancias para las entradas y la salida deberían de ser de los valores que se muestran en la tabla 4.1.

TABLA 4.1: Ganancias de entrada y salida del controlador difuso

Entrada 1	Entrada 2	Salidas
1.53	0.34	0.62

A su vez, se incluyen gráficas para mostrar el desempeño del controlador ajustado. Se ajustaron las manipulaciones al 4.79 % y al 0.6 %, respectivamente para mantener las salidas en un estado inicial del 50 % y del 20 %. Se aplicaron cambios del 10 % en cada una de las referencias para poder probar el desempeño del controlador. Para dar una mejor idea respecto al desempeño del controlador, en la tabla se muestra la efectividad del algoritmo mediante la cuantización de los parámetros de desempeño.

TABLA 4.2: Parámetros de desempeño del controlador difuso

Salida	IAE	IAE Acop	% Mp	ts
Lazo 1	155.2	36.92	0	125 s
Lazo 2	357.8	64.66	5.89	211 s

#### 4.2.4. Ejemplo Comparativo

Para probar la eficacia del algoritmo se tomó como prueba el mejorar la sintonización del controlador propuesto en la bibliografía que dio fuente a la manera de decidir las posiciones de los conjuntos difusos de salida [22]. Para ello se evaluaron los indicadores de desempeño que se contemplan en la función de optimización descrita en la ecuación 4.6. Posteriormente se introdujeron los datos del controlador al algoritmo del recocido simulado y se dejó correr bajo las condiciones descritas a lo largo del capítulo. Una vez terminada la sintonización, midieron los indicadores de desempeño para hacer finalmente el análisis comparativo. El algoritmo se corrió en una vecindad de 0 a 10 como rango de oscilación de los parámetros para este experimento. Como el controlador utiliza solo dos conjuntos difusos de entrada y nueve de salida, no se varió el ancho de las bases.

La configuración de los controladores se resume en los cuadros 4.4 y 4.5. De observar los indicadores de desempeño se observa una mejoría notable, sobretodo en términos de velocidad y tiempo de establecimiento se refiere. El sobretiro es ligeramente superior al obtenido por el algoritmo original, pero es muy aproximado al mismo. El resultado del experimento se concentra en la tabla 4.3 En las gráficas comparativas de 4.8 a 4.11 se puede apreciar la mejoría en la respuesta.

TABLA 4.3: Función de costo para las pruebas realizadas en ambas sintonías de controlador de 16 reglas.

Prueba	IAE	IAED	TS	%MP	SU	$J$
Lazo 1 (original)	45.1793	16.589	182	0	0.157	
Lazo 2 (original)	34.4271	9.9139	128	6.79	0.11	416.4443
Lazo 1 (RS)	34.2147	10.8213	80	0	0.159	
Lazo 2 (RS)	25.1351	7.5377	79	10.23	0.1396	237.1096

	Alfa	Beta	Amp. Original	Amp. RS
error 1	-1	1	0.27	0.12
derivada 1	-1	1	0.2	1.02
error 2	-1	1	1	7.13
derivada 2	-1	1	0.2	1.34
Salidas			0.59	8.13

TABLA 4.4: Configuración de controladores difusos de 16 reglas. Sintonía original y la obtenida mediante algoritmo de sintonización.

	Salida 1	Salida 2
m0	-0.31	-0.157
m1	-0.157	-0.104
m2	-0.153	-0.053
m3	-0.004	-0.05
m4	0	0
m5	0.004	0.05
m6	0.153	0.053
m7	0.157	0.104
m8	0.31	0.157

TABLA 4.5: Configuración de controladores difusos de 16 reglas. Conjuntos difusos de salida.

### 4.3. Conclusión

En este capítulo se habló acerca de la técnica de optimización que será utilizada para sintonizar los controladores difusos propuestos. El algoritmo del recocido simulado

es un método estocástico basado en la analogía que existe entre los problemas de optimización y la formación de cristales durante el enfriamiento de metales y en la teoría de cadenas de Markov.

Las ventajas de utilizar el recocido simulado es que consiste en un método que puede encontrar mínimos globales y puede demostrarse que bajo ciertas condiciones éste tenderá siempre hacia ese punto. Debido a que los tiempos de simulación y el número de iteraciones son finitos, esta situación solo puede ser aproximada. Una buena selección de la cantidad de iteraciones en cada temperatura así como de la agenda de enfriamiento harán que el algoritmo encuentre soluciones aceptablemente cercanas al óptimo global.

Se hizo una descripción de la técnica y de algunos de sus aspectos operativos y la manera en que se aplicaron para el caso en particular. Se presentó un ejemplo de sintonización del controlador difuso multivariable el cual fue ajustado a partir del punto inicial descrito en el capítulo de sistemas difusos. Se demostró que el método es capaz de ajustar un controlador difuso a partir de una base inicial no necesariamente cerca del óptimo global. Finalmente, se hizo un ejemplo comparativo para probar la efectividad del algoritmo, mejorando el desempeño de un controlador difuso multivariable reportado en la literatura.

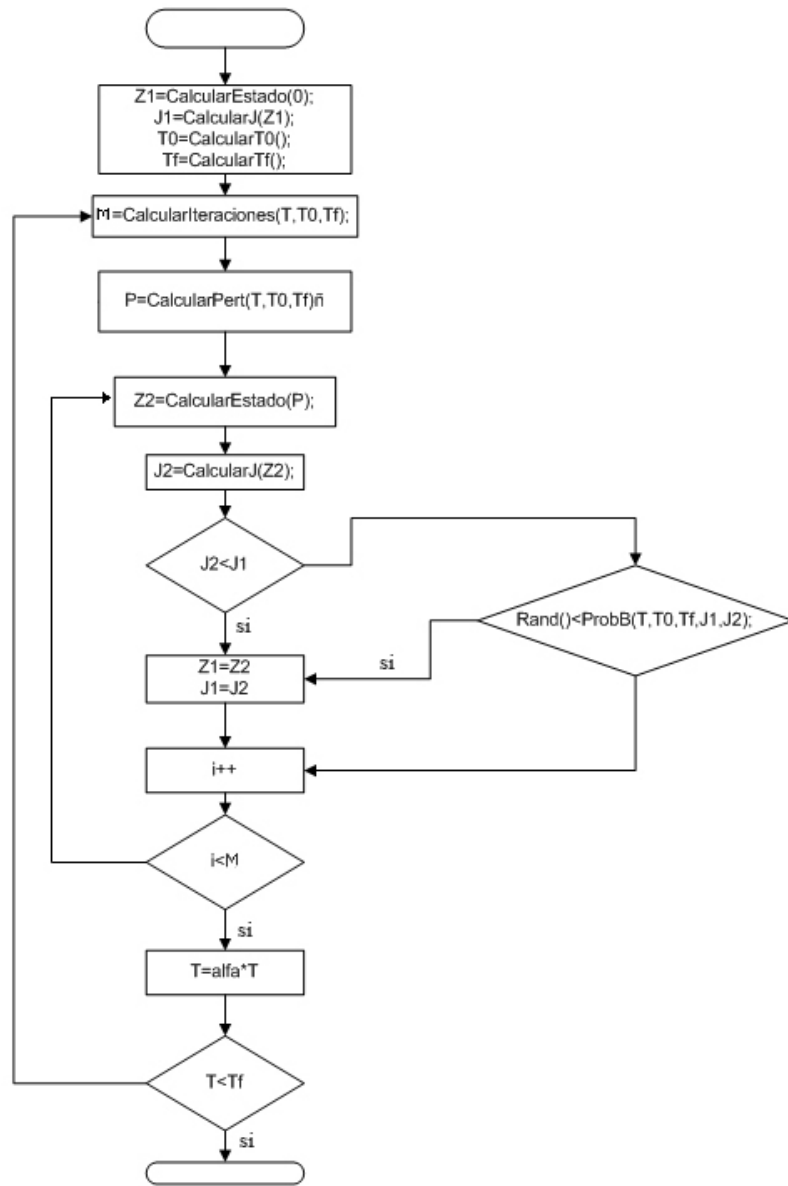


FIGURA 4.1: Diagrama de Flujo del Recocido Simulado

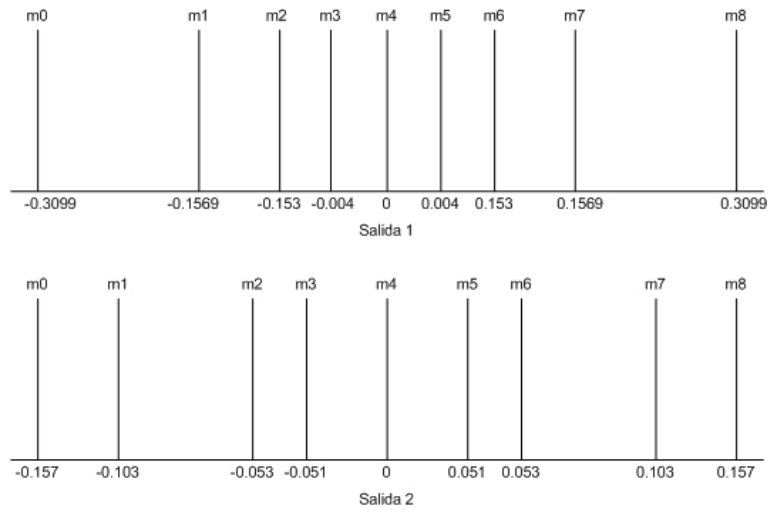


FIGURA 4.2: Conjuntos difusos de salida del controlador.

		e1										
		N	N	N	Z	Z	Z	P	P	P		
de1	N	3	3	0	6	3	0	8	8	5	N	
	N	3	1	0	6	1	2	8	7	5	Z	
	N	3	1	0	3	0	2	8	5	5	P	
	Z	3	3	0	6	6	2	8	7	5	N	
	Z	3	1	0	6	4	2	8	7	5	Z	
	Z	3	1	0	6	2	2	8	7	5	P	
	P	3	3	0	6	8	5	8	7	5	N	
	P	3	1	0	6	7	2	8	7	5	Z	
P	3	0	0	8	5	2	8	7	5	P		
		N	Z	P	N	Z	P	N	Z	P		
		e2										

FIGURA 4.3: Base de reglas para la salida 1

		e1										
		N	N	N	Z	Z	Z	P	P	P		
de1	N	5	5	0	7	5	0	8	8	3	N	
	N	5	2	0	7	2	1	8	6	3	Z	
	N	5	2	0	5	0	1	8	3	3	P	
	Z	5	5	0	7	7	1	8	6	3	N	
	Z	5	2	0	7	4	1	8	6	3	Z	
	Z	5	2	0	7	1	1	8	6	3	P	
	P	5	5	0	7	8	3	8	6	3	N	
	P	5	2	0	7	6	1	8	6	3	Z	
P	5	0	0	8	3	1	8	6	3	P		
		N	Z	P	N	Z	P	N	Z	P		
		e2										

FIGURA 4.4: Base de reglas para la salida2

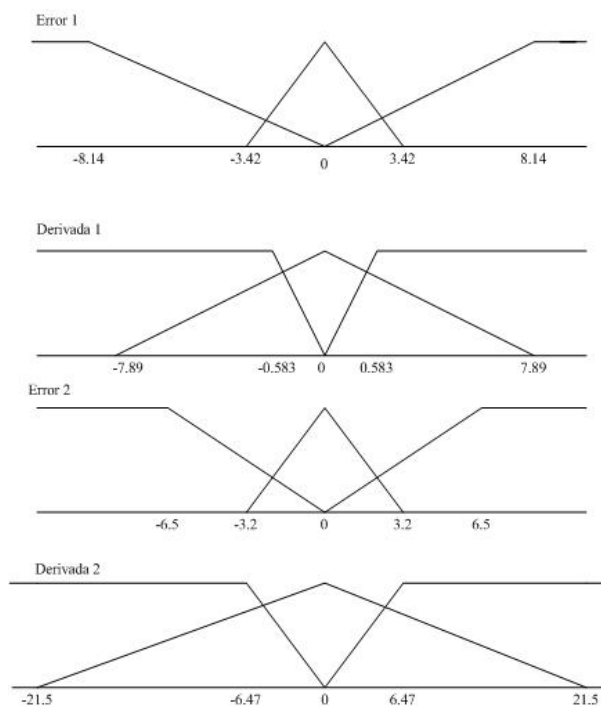


FIGURA 4.5: Ajuste de los conjuntos difusos de entrada

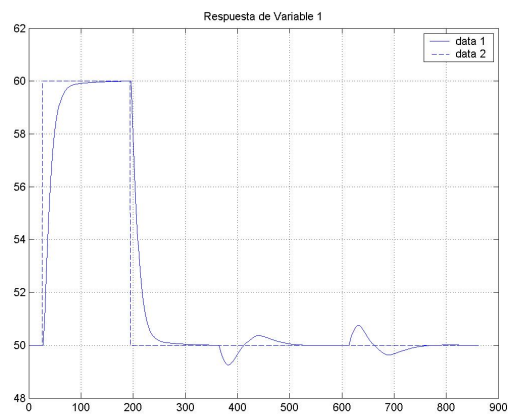


FIGURA 4.6: Respuesta del sistema sintonizado mediante RS.

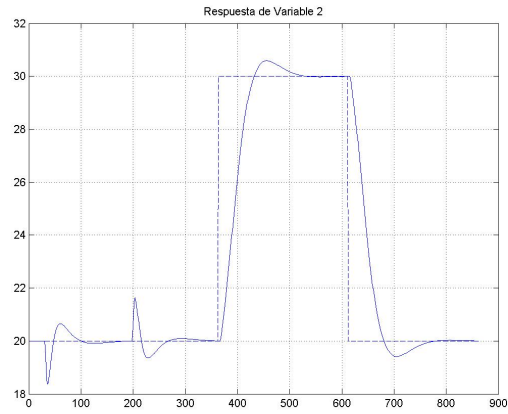


FIGURA 4.7: respuesta del sistema sintonizado mediante RS.

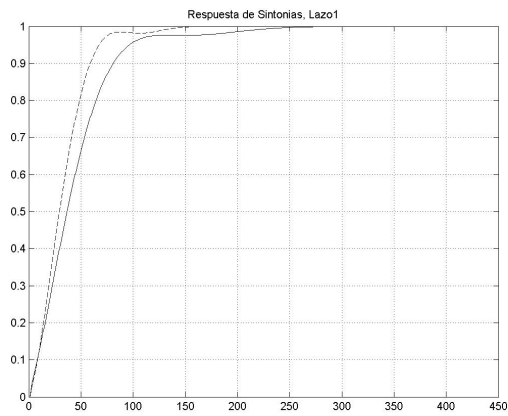


FIGURA 4.8: Controlador de 16 reglas. Comparación de sintonía en servocontrol, lazo 1. Sólido: original. Punteado: Obtenido mediante RS.

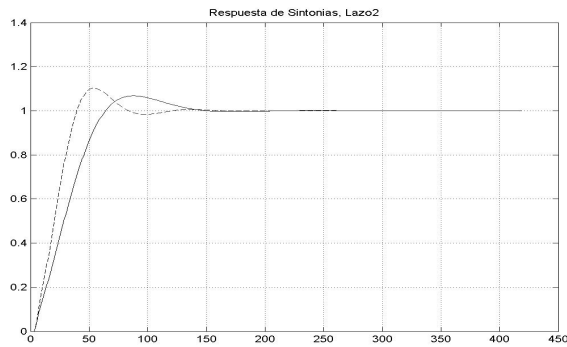


FIGURA 4.9: Controlador de 16 reglas. Comparación de sintonía en servocontrol, lazo 2. Sólido: original. Punteado: Obtenido mediante RS.



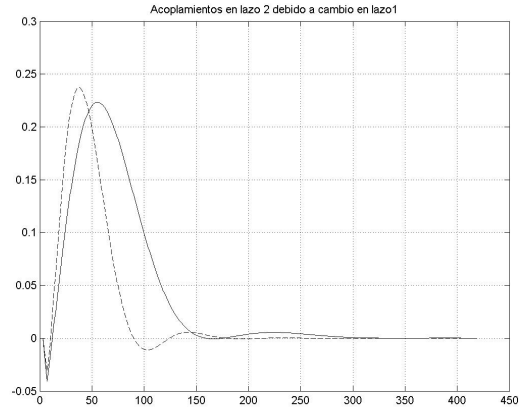


FIGURA 4.10: Controlador de 16 reglas. Acoplamiento de lazo 2 debido a cambio en lazo 1. Sólido: original. Punteado: Obtenido mediante RS.

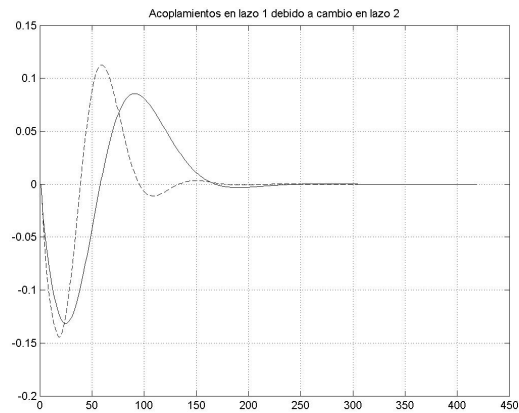


FIGURA 4.11: Controlador de 16 reglas. Acoplamiento de lazo 1 debido a cambio en lazo 2. Sólido: original. Punteado: Obtenido mediante RS.

## Capítulo 5

# Estrategia de comparación

### 5.1. Introducción

El problema del control multivariable ha sido ampliamente estudiado y atacado mediante diversas técnicas. Controladores mediante cancelación de polos o basados en el concepto de espacio de estados son solo algunos de los ejemplos disponibles.

La teoría de control clásica ha aportado a su vez su contribución para controlar sistemas de múltiples entradas y múltiples salidas (MIMO) de manera efectiva. En este caso, estrategias basadas en controladores convencionales y bloques de atraso/adelanto son las opciones que se utilizan en aplicaciones prácticas con gran aceptación.

Debido a estas razones, es necesario hacer algún tipo de análisis con el fin de determinar la calidad del desempeño del controlador difuso en comparación con las técnicas existentes. Para ello se eligió una estrategia convencional que se considerara conveniente, si sintonizó para que ofreciera el mejor desempeño posible. En el presente capítulo se describirán estos hechos.

### 5.2. Descripción de la estructura de PID con desacoplador

De todas las técnicas disponibles para atacar el problema del control multivariable se optó por elegir la estrategia del PID con desacoplador por las siguientes razones:

1. El algoritmo de PID con desacoplador es de las estrategias más comunes en aplicaciones industriales.
2. El desempeño, dentro de sus limitaciones, es muy aceptable.
3. Es un algoritmo sencillo y ampliamente conocido y estudiado.
4. Existen reportadas numerosas maneras de sintonizar sus parámetros.

La estrategia de PID con desacoplador se basa en un análisis de ganancias relativas para determinar con cuales entradas se debe de controlar cuales variables de proceso y en el diseño de bloques de cancelación que eliminen el efecto de una variable de manipulación en las salidas que no le corresponde controlar. Una vez diseñados estos bloques, puede decirse que el proceso multivariable se comporta como si hubiera sido partido en lazos independientes, por lo tanto, un controlador univariable como es el PID puede controlar el proceso. En este caso solo hay que observar que los lazos resultantes estén dentro de los alcances del PID, por ejemplo, que la relación entre el tiempo muerto y la constante de tiempo sea tal que

$$0.1 < \left| \frac{\theta}{\tau} \right| < 1 \quad (5.1)$$

En la figura 5.1 se muestra la estrategia de control esquematizada para el proceso multivariable.

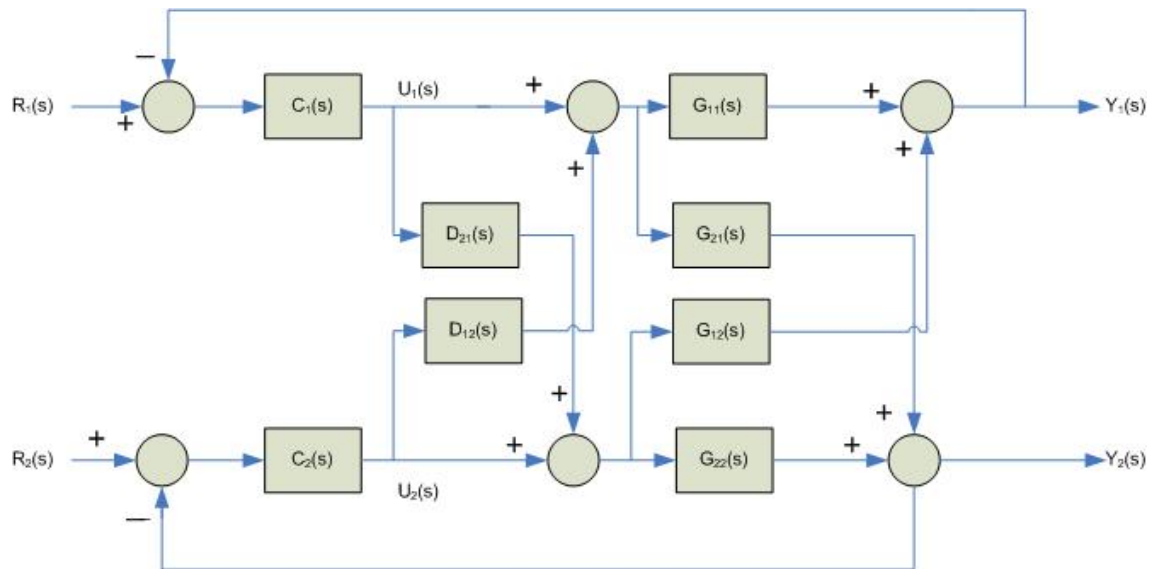


FIGURA 5.1: Diagrama de bloques del lazo de control multivariable

### 5.2.1. Ganancias Relativas

La teoría de ganancias relativas es una herramienta que nos ayuda a determinar cual de las entradas disponibles debe de utilizarse para controlar cada variable de proceso. En la bibliografía [4] se reporta una amplia explicación de este concepto y de los cálculos pertinentes.

Suponiendo que en estado estable se puede representar al proceso multivariable de la siguiente manera:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (5.2)$$

Es posible obtener la ganancia de lazo abierto  $K_{ij}$  y la de lazo cerrado  $K'_{ij}$ . Cada una de estas ganancias está definida como se muestra

$$K_{11} = \left. \frac{\partial y_1}{\partial u_1} \right|_{u_2=0} \quad (5.3)$$

$$K'_{11} = \left. \frac{\partial y_1}{\partial u_1} \right|_{y_2=0} \quad (5.4)$$

lo mismo aplica para el resto de las ganancias.

La ganancia de Bristol  $\lambda$  está definida por estas cantidades como

$$\lambda_{ij} = \frac{K_{ij}}{K'_{ij}} \quad (5.5)$$

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} \quad (5.6)$$

La razón expresada en la ecuación 5.5 implica que para que una entrada se considere la manipulación adecuada de una salida determinada, su efecto debe ser tan significativo en lazo cerrado como en lazo abierto. Esto se interpreta con valores cercanos a 1. En algún otro caso, la razón nos expresará que el efecto de tal salida es muy disperejo, y eso se deberá a que alguna otra entrada es la que ejerce una mayor interacción sobre la salida que se analiza.

Para el caso especial de  $2 \times 2$ , el término  $\lambda_{11}$  se puede obtener por la expresión 5.7. La suma de las ganancias relativas debe dar 1 tanto en los renglones como en las columnas de 5.6. Para el caso de  $2 \times 2$ , con una sola es suficiente como para definir el orden de los lazos de control.

$$\lambda_{11} = \frac{K_{12}K_{21}}{K_{12}K_{21} - K_{11}K_{22}} \quad (5.7)$$

### 5.2.2. Diseño del desacoplador

Es posible ver los efectos del acoplamiento como una perturbación en una de las salidas por efecto de las otras, por ello, técnicas que se utilizan en el control antealimentado son viables para eliminar los efectos de las manipulaciones ajenas sobre el lazo de control.

Partiendo de que el valor de salida es el resultado de los cambios en ambas entradas y que existen los bloques  $D_{12}(s)$  para eliminar el efecto de  $Gp_{12}(s)$  en  $Y_1(s)$  y  $D_{21}(s)$

para eliminar el efecto de  $Gp_{21}(s)$ , se analiza el diagrama de bloques de la figura 5.1 para obtener

$$\begin{aligned} Y_1(s) &= [D_{12}(s)Gp_{11}(s) + Gp_{12}(s)]U_2(s) \\ Y_2(s) &= [D_{21}(s)Gp_{22}(s) + Gp_{21}(s)]U_1(s) \end{aligned} \quad (5.8)$$

Al decir que se desea eliminar el efecto se refiere a hacer  $Y_1(s)$  y  $Y_2(s)$  iguales a 0 en su caso respectivo. Despejando estas expresiones se obtienen las funciones del desacoplador

$$\begin{aligned} D_{12} &= \frac{-Gp_{12}(s)}{Gp_{11}(s)} \\ D_{21} &= \frac{-Gp_{21}(s)}{Gp_{22}(s)} \end{aligned} \quad (5.9)$$

Esto se desarrolló para el caso de 2 x 2 solo por sencillez, pero el procedimiento es el mismo al descrito en órdenes mayores, por lo que no se pierde generalidad.

Suponiendo que las expresiones del modelo son de primer orden con tiempo muerto, la forma del desacoplador es la de un bloque de atraso adelante, es decir

$$D_{ij}(s) = \frac{-K_{ij} \tau_{ij}s + 1}{K_{ii} \tau_{ii}s + 1} e^{-(\theta_{ij}s - \theta_{ii}s)} \quad (5.10)$$

En ocasiones la misma dinámica del proceso impedirá la implementación del desacoplador completo; esto ocurrirá cuando  $\theta_{ij} < \theta_{ii}$ , dado que esto representa un irrealizable adelante en el tiempo. En estos casos, la mejor solución será eliminarlo. Esto deteriorará un poco el desempeño pero se conservará en buena medida los beneficios del desacoplador.

Para efectos del PID utilizado, se eligió en este caso utilizar la estructura de PID ideal.

### 5.3. Sintonización de la estrategia de control

Una vez determinados los bloques que conformarán la estrategia, resta realizar el ajuste necesario para que ésta se desempeñe adecuadamente.

Para el caso de los bloques desacopladores, la sintonía depende completamente de los parámetros del proceso multivariables. Por lo tanto, en cuanto se hayan identificado las funciones de transferencia que lo conforman se puede proceder directamente a su diseño haciendo uso de las ecuaciones 5.9 y 5.10.

Sin embargo, el ajuste de los controladores PID es una tarea un poco más complicada, dado que existen infinidad de combinaciones de los parámetros de ganancia proporcional, tiempo integral y tiempo derivativo. Además, de que existen múltiples criterios para escoger.

En aras de diseñar un PID con desacoplador con el mejor desempeño posible, se retomó del trabajo mencionado en [15] para obtener las reglas de sintonía del controlador.

### 5.3.1. Sintonía obtenida mediante programación genética

En trabajos anteriores [15], se ha buscado mejorar la sintonía de los controladores PID, dadas las razones que en este documento ya se han enumerado. Dado que este método ha demostrado dar mejores resultados que otras reglas de sintonía, se decidió elegirla como una base para establecer la sintonización final de los controladores implementados.

La técnica se basa en un método propuesto originalmente por Nishikawa et al. donde la información que se utiliza para sintonizar el controlador consiste en ciertas áreas del error y de la respuesta ante un cambio en escalón. En [15], lo que se buscó fue generar nuevas reglas basándose en programación genética a partir de las cantidades que se obtienen de las áreas mencionadas.

Las áreas mencionadas se denotan en la figura 5.2. En ella pueden apreciarse las áreas  $ST$  y  $S(A)$  que están definidas por las expresiones

$$A = \frac{ST}{X_{SS}} \quad (5.11)$$

$$B = \frac{eS(A)}{X_{SS}} \quad (5.12)$$

donde  $e$  es la base del logaritmo natural y  $X_{SS}$  es el valor de estado estable al que llega el proceso ante el cambio en escalón. De hacer un análisis de la respuesta pura de primer orden, podrá apreciarse que dichas cantidades están relacionadas a los parámetros del modelo FOPDT.

Tomando directamente las reglas de sintonía de la bibliografía citada, tenemos que las relaciones entre  $A$ ,  $B$ , la ganancia del proceso  $Kp$  y los parámetros del PID son

$$Kc = \left| 0.569703048 \frac{A}{Kp(A - B)} \right| \quad (5.13)$$

$$Ti = 0.2973385445A + 0.7926614555B \quad (5.14)$$

$$Kc = |0.3630116172(A - B)| \quad (5.15)$$

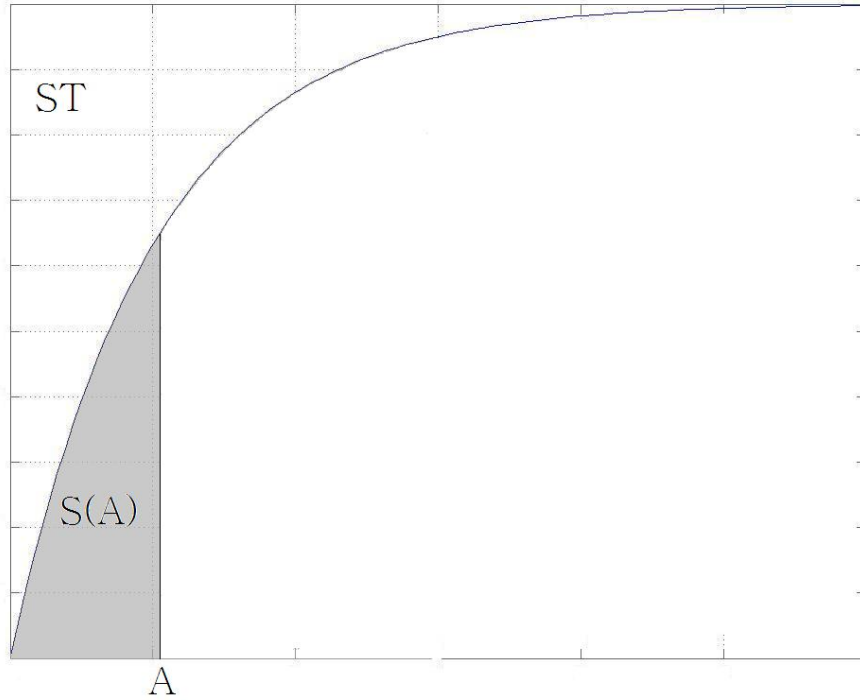


FIGURA 5.2: Definición de áreas A y B de la respuesta de primer orden

### 5.3.2. Ejemplo de diseño de PID con desacoplador

Para el proceso multivariable propuesto en la bibliografía [22]

$$H(s) = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \quad (5.16)$$

se le aplicó la metodología mencionada para diseñar el PID con desacoplador. Para este caso en particular, los desacopladores se diseñan usando 5.9 y 5.10 para obtener

$$D_{ij}(s) = \frac{18.9}{12.8} \frac{16.7s + 1}{21s + 1} e^{-2s} \quad (5.17)$$

$$D_{ij}(s) = \frac{6.6}{19.4} \frac{14.4s + 1}{10.9s + 1} e^{-4s} \quad (5.18)$$

Haciendo una prueba escalón unitario a cada uno de los lazos resultantes se obtienen los valores de  $K_p$ , A y B necesarios para sintonizar el controlador. El resultado se aprecia en la tabla 5.1

TABLA 5.1: Sintonía de estrategia PID con desacoplador

	$A$	$B$	$K_p$	$K_c$	$T_i$	$T_d$
Lazo 1	11.662	4.836	6.387717	0.152	6.866	2.477
Lazo 2	11.066	3.957	-9.655	-0.091	6.071	2.580

## 5.4. Mejoramiento del desempeño del controlador mediante el uso del recocido simulado

Dado que el objetivo de esta sección es buscar un controlador difuso lo mejor sintonizado posible y se cuenta con la técnica del recocido simulado desarrollada para realizar esos fines en los controladores difusos, se consideró interesante la idea de utilizar la misma técnica para buscar una mejora en el desempeño del controlador. Para ello se siguió la metodología descrita en el capítulo referente al tema. Esta se aplicó prácticamente sin cambios, salvo que la cantidad de parámetros a modificar era mucho menor ya que solo se fueron perturbando aleatoriamente los valores de  $K_c$ ,  $T_i$  y  $T_d$  de cada controlador. El punto de comparación que se utilizó para este caso fue el mismo que se aplicó para el controlador difuso, el cual está descrito por la ecuación 4.6. La única modificación que pudiera ser pertinente [20] es seguir la recomendación de considerar solamente el máximo de cada parámetro de la función de optimización en lugar de la suma de todos ellos, lo cual puede aplicarse para los casos que el primer enfoque de una sintonización muy buena y la otra con un gran costo. El hecho de utilizar una u otra dependerá del caso en particular, por lo que no se hará una recomendación que se incline por alguna de estas opciones.

Otra de las diferencias importantes que deben de mencionarse consiste en que para el PID con desacoplador se partió directamente de la sintonización dada en la tabla 5.1 como inicio del algoritmo. En el difuso se invierte relativamente poco esfuerzo en hacer ajustes iniciales dado que la idea principal del método consiste en librar al diseñador de tales inconvenientes, permitiéndole ajustar el controlador con poca información.

Partiendo de estos puntos, se aplicó el algoritmo de recocido simulado al PID con desacoplador y logró ajustarlo de tal manera que la función de optimización fuera menor. Debido al éxito obtenido, se decidió utilizar este nuevo juego de valores como el punto de comparación entre las técnicas difusas y las convencionales.

TABLA 5.2: Comparación de sintonía obtenida con programación genética y la mejorada por el recocido simulado

	IAE1	IAED	Ts	%Mp	Ess	SU	J
Lazo 1 (G)	9.2051	0	56	4.39	0	1.7421	
Lazo 2 (G)	9.844	0	49	011.1	0	1.2848	127.2365
Lazo 1 (RS)	6.9116	0	26	20.21	0	1.2284	
Lazo 2 (RS)	12.3239	0	41	0.09	0	0.3023	87.97



En la tabla 5.2 se muestra la comparación entre ambas sintonías. La ventaja principal de la propuesta por el recocido simulado es su velocidad de respuesta. El resto de los parámetros son de magnitud comparable. En las figuras 5.3 y 5.4 se aprecia la respuesta de cada sintonización ante un cambio escalón unitario en la entrada del lazo de control.

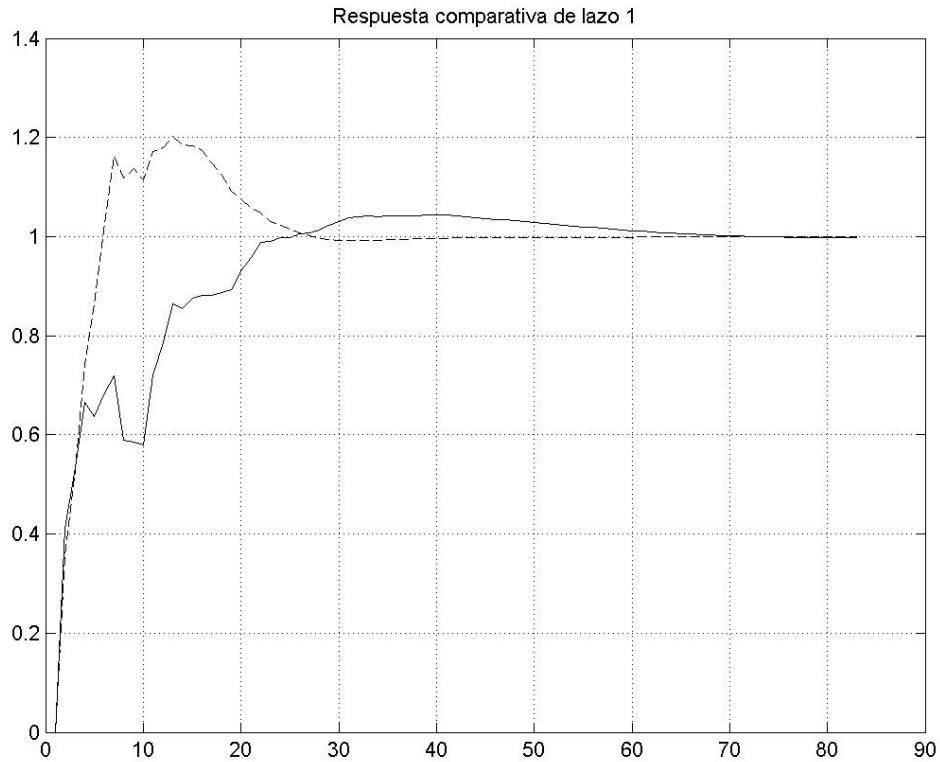


FIGURA 5.3: Respuesta de PID en lazo 1. Continua: Genético. Punteada: Propuesto por RS.

## 5.5. Conclusión

En este capítulo se realizó un breve resumen de la técnica de PID con desacoplador, se justificó el porqué de su elección y se hizo una descripción de su funcionamiento. Se propuso una sintonía basada en la investigación bibliográfica realizada y se sometió a un proceso de mejoramiento mediante el uso del recocido simulado. Se hizo una comparación de los resultados obtenidos. En los capítulos siguientes se enfocará a hacer comparaciones entre esta estrategia ajustada mediante el recocido simulado y las estrategias difusas que han sido descritas previamente.

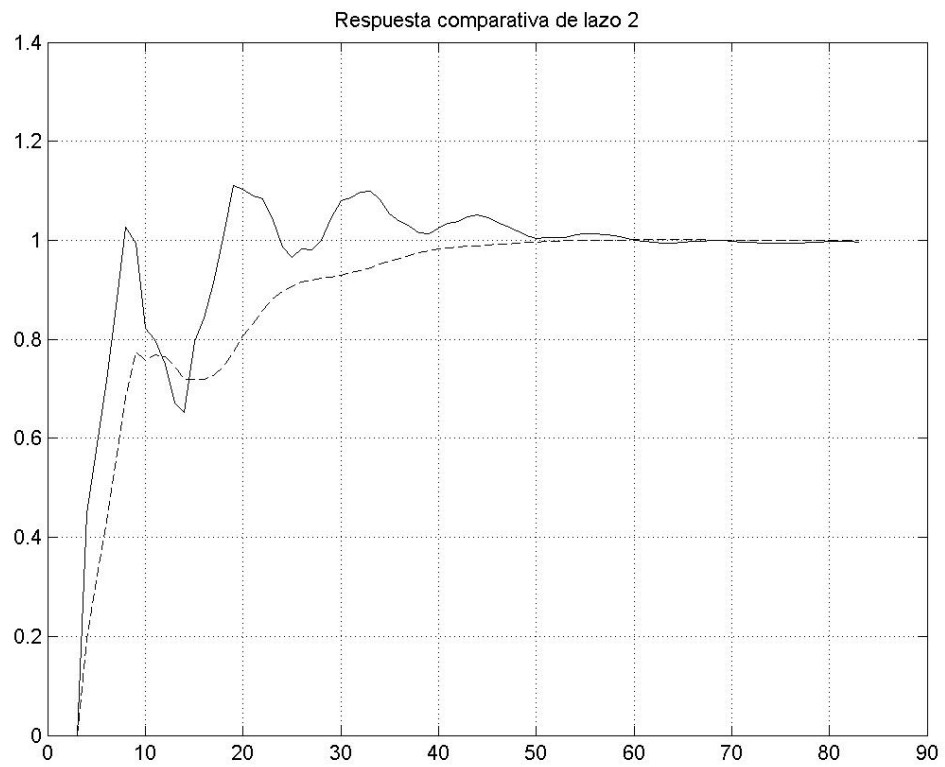


FIGURA 5.4: Respuesta de PID en lazo 2. Continua: Genético. Punteada: Propuesto por RS.

## Capítulo 6

# Presentación y análisis de resultados

### 6.1. Introducción

Hasta el momento han sido descritas las diferentes estrategias basadas en lógica difusa que se consideraron para el control de procesos multivariables; se mostró su estructura y los principios que rigen su comportamiento. Además, se presentó la estrategia convencional que serviría como punto de comparación junto con las reglas y métodos de sintonía para cada uno de los casos que se tomó en consideración.

Aunque se han presentado ejemplos que muestran el funcionamiento de los controladores, es necesario medir su desempeño para poder hacer una comparación objetiva de los mismos en base a los indicadores de interés. En este caso, la estrategia del PID con desacoplador fue elegida como base de comparación por las razones que han sido mencionadas en los capítulos anteriores.

Con estas ideas en mente, en este capítulo se describirán las pruebas que fueron realizadas para medir este desempeño así como un análisis comparativo de los resultados obtenidos. Para ello se planteará, en primer lugar, la naturaleza que deben de tener estas pruebas y la forma en que se medirá el desempeño. Una vez hecho esto, se presentarán los resultados obtenidos para finalmente deducir conclusiones a partir de ellos.

### 6.2. Diseño de las pruebas

La mejor forma de medir el desempeño del controlador es hacer pruebas de lazo cerrado sobre el sistema completo. Para ello se utilizaron las rutinas programadas y se hicieron simulaciones de las que se evaluó la respuesta.

Dado que el escalón unitario es la entrada ideal más utilizada para evaluar un sistema de control a nivel teórico, se decidió aplicarla en las simulaciones para probar el desempeño. Otra razón que justifica esta elección es que esta señal fue utilizada en el algoritmo de optimización, por lo que será la entrada a la que los controladores ofrezcan el mejor de sus comportamientos.

Otro aspecto importante es que las simulaciones que se presentan fueron realizadas

considerando los modelos de variable desviación únicamente, es decir, que se realizaron considerando cero las condiciones iniciales. Esto se puede tomar como válido porque el proceso de identificación normalmente es hecho alrededor de un punto de operación, el cuál es descrito por el modelo utilizado. Así, para las regiones cercanas a este punto, la respuesta obtenida será la misma que la presentada en los resultados sumándole al valor de estado estable en el que se encontrara el proceso. Por estas razones, valores negativos de las variables de proceso y de las manipulaciones deben ser consideradas como desviaciones del valor inicial y no como señales fuera del rango posible.

Se decidió diseñar dos tipos de pruebas, las cuales reflejaran el funcionamiento ante condiciones que pueden ocurrir en la realidad. La primera de ellas, y la más sencilla, se basa en el supuesto de que el modelo identificado es válido y describe al proceso de la mejor manera posible. Este se considera como el caso ideal en el que se espera que los controladores den sus mejores resultados.

El otro escenario que puede plantearse se presenta bajo la suposición de que el modelo puede no ser la mejor manera de describir al proceso. Esto puede deberse a imprecisiones durante el proceso de identificación, limitaciones de la estructura del modelo que eliminan efectos dinámicos de alto orden, no linealidades del sistema, cambios de las condiciones de operación, o simplemente, modificaciones naturales del proceso debidas al paso del tiempo. Bajo estas condiciones las expectativas de desempeño disminuyen dado que el proceso para el que los controladores habían sido sintonizados se volvió distinto al original. Este análisis ofrecerá una idea de la insensibilidad al cambio de parámetros que tiene el controlador bajo prueba, y a su vez, esto nos dará una idea de la robustez de los algoritmos.

En un sistema multivariable existen numerosas posibilidades de combinaciones de parámetros. Si se toma el ejemplo concreto de un proceso multivariable de  $2 \times 2$  y se supone que toda su dinámica es modelada mediante funciones de transferencia de primer orden con tiempo muerto, se tendrá un total de 531441 modelos diferentes. Si a ello se le agrega la posibilidad de que es posible variar la magnitud de los cambios, se tiene como resultado una infinidad de experimentos por realizar por cada proceso a analizar.

Tomando las experiencias reportadas en la bibliografía [3] se concluyó que no es necesario hacer un análisis tan extensivo, sino que es suficiente con hacer las pruebas que resulten significativas.

Basados en lo reportado en el trabajo citado, se determinó que no era necesario hacer las pruebas donde varía un solo parámetro a la vez y que lo más significativo es cambiar todos los parámetros del mismo tipo al mismo tiempo. Así, lo que se realizará serán pruebas donde se aumenten o disminuyan en la misma proporción las ganancias, las constantes de tiempo y los tiempos muertos del sistema, respectivamente.

Además de estos experimentos, se recomienda realizar pruebas en la que todos los parámetros del modelo sean cambiados simultáneamente. Esto puede consistir la

situación más cercana a la realidad, dado que nadie puede predecir realmente de que manera puede ser modificado el proceso. Debido a que las posibilidades de cambio en este caso son también muy amplias se decidió hacer la prueba recomendada como el peor caso. Haciendo referencia al trabajo antes citado y partiendo que la dinámica se describe con procesos de primer orden con tiempo muerto se tiene que el peor caso ocurre cuando

- Aumenta la ganancia del proceso.
- Disminuye la constante de tiempo.
- Aumenta el tiempo muerto.

Una justificación que puede darse al respecto es que al aumentar la ganancia, el proceso se vuelve mucho más sensible, requiriendo que la manipulación sea más suave y precisa para mantenerlo bajo control. Respecto a los cambios de la constante de tiempo y el retraso, esto tiene que ver con la razón  $\theta/\tau$ , la cual debe de mantenerse menor a 1 para que pueda ser controlado sin la necesidad de técnicas de predicción. Esto se expresa en la desigualdad denotada en la ecuación 5.1.

Lo que fue referente a la magnitud del cambio en los parámetros se buscó un cambio de un impacto significativo en los resultados obtenidos. Se consideró que cambios de un 50 % deberían de ser suficientes como para observar el efecto de éstos en el desempeño del controlador.

Respecto a las estrategias que se sometieron a prueba fueron la del PID con desacoplador mejorado mediante el recocido simulado, el controlador difuso multivariable, el par de controladores difusos con desacoplador y el PID de 3 difusos con desacoplador. A su vez, y solo con fines de tener un punto de comparación con las referencias anteriores, también se realizaron los experimentos al PID ajustando originalmente con programación genética.

Para conocer el desempeño de cada controlador ante cada una de las pruebas realizadas, éste fue evaluado mediante la función de costo presentada en la ecuación 4.6. En el caso de los valores de cada una de las constantes de esta ecuación, no es posible utilizar la función original con la que se sintonizó para estos experimentos porque éstas son calculadas automáticamente bajo los criterios mencionados en el capítulo correspondiente. Por tal motivo, y con el fin de tener una misma base para evaluar el desempeño, se tomaron todas las constantes de 4.6 como unitarias y se presentó el valor de cada uno de los indicadores de desempeño para poder analizar a mayor profundidad la calidad del comportamiento.

Finalmente, haciendo un resumen de lo antes mencionado, las pruebas que se realizaron son las siguientes:

1. Pruebas en condiciones ideales, sin cambio de parámetros.

2. Pruebas aumentando los valores de ganancia en un 50 %.
3. Pruebas disminuyendo los valores de ganancia en un 50 %.
4. Pruebas aumentando los valores de constante de tiempo en un 50 %.
5. Pruebas disminuyendo los valores de constante de tiempo en un 50 %.
6. Pruebas aumentando los valores de tiempo muerto en un 50 %.
7. Pruebas disminuyendo los valores de tiempo muerto en un 50 %.
8. Pruebas aumentando todos los valores en un 50 %.
9. Pruebas disminuyendo todos los valores en un 50 %.
10. Pruebas aumentando en el peor caso.

### 6.3. Resumen de sintonizaciones utilizadas

Se siguieron los lineamientos mencionados en los capítulos anteriores. Las sintonías que se utilizaron como base a las pruebas realizadas son resumidas de las tablas 6.1 a 6.16.

TABLA 6.1: Sintonización de controlador difuso multivariable. Caso 1 signo. Conjuntos difusos de entrada.

Conjunto	Alfa $\alpha$	Beta $\beta$	Gamma $\gamma$
Z err1	-0.59	0	0
T err1	-1.42	0	1.42
S err1	0	0.59	0
Z derr1	-0.03	0	0
T derr1	-0.62	0	0.62
S derr1	0	0.03	0
Z err2	-0.48	0	0
T err2	-0.59	0	0.59
S err2	0	0.48	0
Z derr2	-1.05	0	0
T derr2	-0.57	0	0.57
S derr2	0	1.05	0

En las tablas 6.1 y 6.2 se muestra la sintonización de los conjuntos difusos de entrada para los casos evaluados. A diferencia del caso del PID difuso, también se modificaron los valores de los conjuntos difusos de salida (tablas 6.4 y 6.3) y las ganancias que del controlador (tablas 6.5 y 6.6). Debido a que en esta estrategia de control no se

TABLA 6.2: Sintonización de Controlador difuso multivariable. Caso 2 signos. Conjuntos difusos de entrada.

Conjunto	Alfa	Beta	Gamma
Z err1	-0.39	0	0
T err1	-1.08	0	1.08
S err1	0	0.39	0
Z derr1	-0.06	0	0
T derr1	-0.72	0	0.72
S derr1	0	0.06	0
Z err2	-0.61	0	0
T err2	-0.66	0	0.66
S err2	0	0.61	0
Z derr2	-0.06	0	0
T derr2	-0.18	0	0.18
S derr2	0	0.06	0

TABLA 6.3: Sintonización de controlador difuso multivariable. Caso 1 signo. Conjuntos difusos de salida.

Manipulación	Salida 1	Salida 2
m0	-0.310	-0.157
m1	-0.157	-0.104
m2	-0.153	-0.053
m3	-0.004	-0.050
m4	0.000	0.000
m5	0.004	0.050
m6	0.153	0.053
m7	0.157	0.104
m8	0.310	0.157

TABLA 6.4: Sintonización de Controlador difuso multivariable. Caso 2 signos. Conjuntos difusos de salida.

Manipulación	Salida 1	Salida 2
m0	-0.310	-0.157
m1	-0.157	-0.104
m2	-0.153	-0.053
m3	-0.004	-0.050
m4	0.000	0.000
m5	0.004	0.050
m6	0.153	0.053
m7	0.157	0.104
m8	0.310	0.157

TABLA 6.5: Sintonización de controlador difuso multivariable. Caso 1 signo. Ganancias

Señal	Ganancia
Salidas	0.03
err1	1.39
derr1	1.39
err2	0.27
derr2	0.27

TABLA 6.6: Sintonización de Controlador difuso multivariable. Caso 2 signos. Ganancias.

Señal	Ganancia
Salidas	0.28
err1	0.65
derr1	0.65
err2	0.23
derr2	0.23

partía de un caso completamente lineal no era necesario observar tal simetría, por lo que ambos tipos de conjuntos pueden ser variados simultáneamente. Lo más recomendable en estos casos es hacer la modificación dentro de un espacio normalizado, dado que la modificación de las ganancias se encarga de la acción lineal del controlador. De todos modos se presentan estos resultados porque fueron los candidatos que obtuvieron el mejor desempeño.

TABLA 6.7: Sintonización de Controladores difusos univariables . Caso 1 signo. Conjuntos difusos de entrada

Conjunto	Alfa	Beta	Gamma
Zerr1	-0.954	0	0
Terr1	-0.496	0	0.496
Serr1	0	0.954	0
Zderr1	-0.737	0	0
Tderr1	-0.204	0	0.204
Sderr1	0	0.737	0
Zerr2	-0.801	0	0
Terr2	-0.001	0	0.001
Serr2	0	0.801	0
Zderr2	-0.459	0	0
Tderr2	-0.805	0	0.805
Sderr2	0	0.459	0

En las tablas 6.7, 6.8 y 6.9 se presenta la sintonización de los controladores univariables para el caso de un signo y lo mismo ocurre para el caso de dos signos negativos en las tablas 6.10, 6.11 6.12. Se presentan los conjuntos difusos de entrada, de salida y



TABLA 6.8: Sintonización de controlador difuso univariable. Caso 1 signo. Conjuntos difusos de salida.

Manipulación	Salida 1	Salida 2
m0	-0.310	-0.157
m1	-0.157	-0.104
m2	-0.153	-0.053
m3	-0.004	-0.050
m4	0.000	0.000
m5	0.004	0.050
m6	0.153	0.053
m7	0.157	0.104
m8	0.310	0.157

TABLA 6.9: Sintonización de controlador difuso univariable. Caso 1 signo. Ganancias.

Señal	Ganancia
err1	13.51
salida 1	0.01
err2	6.16
salida 2	0.0029

TABLA 6.10: Sintonización de Controladores difusos univariables . Caso 2 signos. Conjuntos difusos de entrada.

Conjunto	Alfa	Beta	Gamma
Z err1	-1.686	0	0
T err1	-155.358	0	155.358
S err1	0	1.686	0
Z derr1	-3.277	0	0
T derr1	-12.748	0	12.748
S derr1	0	3.277	0
Z err2	-1.091	0	0
T err2	-1.146	0	1.146
S err2	0	1.091	0
Z derr2	-1.640	0	0
T derr2	-7.565	0	7.565
S derr2	0	1.640	0

TABLA 6.11: Sintonización de Controladores difusos univariables . Caso 2 signos. Conjuntos difusos de salida.

Manipulación	Salida 1	Salida 2
m0	-0.147	-0.864
m1	-0.012	-0.020
m2	-0.743	-0.136
m3	-0.470	-0.114
m4	0.000	0.000
m5	0.470	0.114
m6	0.743	0.136
m7	0.012	0.020
m8	0.147	0.864

TABLA 6.12: Sintonización de Controladores difusos univariables . Caso 2 signos. Ganancias.

Señal	Ganancia
err1	1.94
salida 1	1.06
err2	0.28
salida 2	3.34

las ganancias, respectivamente. En este caso se siguió los lineamientos del caso multivariable y se modificaron todos los valores a la vez. Las mismas recomendaciones aplican para este caso, pero de igual manera se exponen los candidatos que ofrecieron los mejores resultados. Esta estrategia difusa tiene ventaja sobre la estrategia multivariable porque el desacoplador se encarga de eliminar la interacción de los lazos de control, lo cual disminuye los condicionamientos de la acción de control y permite un desempeño con mayor velocidad.

TABLA 6.13: Sintonización de PID (RS) . Caso 1 signo.

PID	Lazo 1	Lazo 2
Kc	0.16	0.04
Ti	25.05	16.0579
Td	0.02	0.40

En las tablas 6.13 y 6.14 se presentan las sintonizaciones optimizadas de los PID propuestos. Estos controladores se ajustaron con ayuda de las ecuaciones 5.13, 5.14 y 5.15 y posteriormente se sometieron al algoritmo de optimización, obteniendo los resultados mostrados. En las comparaciones de las pruebas realizadas se incluyeron pruebas con la sintonización original, a sabiendas de que existe un mejor ajuste. Esto se hizo con el fin de tener un punto de comparación con los trabajos anteriores.

En la tablas 6.15 y 6.16 se presentan la sintonizaciones de los PID difusos propues-

TABLA 6.14: Sintonización de Controlador PID. Caso 2 signos. Ganancias.

PID	Lazo 1	Lazo 2
Kc	0.13	0.08
Ti	1.93	10.04
Td	2.22	0.67

TABLA 6.15: Sintonización de PID de 3 difusos. Caso 1 signo.

Acción	Kc1	Ti1	Td1	Kc2	Ti2	Td2
Ganancia	0.16	25.0543	0.02	0.04	16.0579	0.40
m0	-1	-1.00	-1.00	-1.00	-1.00	-1.00
m1	-0.70559	-0.67	-0.84	-0.91	-0.99	-0.91
m2	-0.37779	-0.64	-0.54	-0.86	-0.61	-0.85
m3	-0.24635	-0.22	-0.14	-0.11	-0.20	-0.78
m4	0	0.00	0.00	0.00	0.00	0.00
m5	0.24635	0.22	0.14	0.11	0.20	0.78
m6	0.37779	0.64	0.54	0.86	0.61	0.85
m7	0.70559	0.67	0.84	0.91	0.99	0.91
m8	1	1.00	1.00	1.00	1.00	1.00

TABLA 6.16: Sintonización de PID de 3 difusos . Caso 2 signos.

Acción	Kc1	Ti1	Td1	Kc2	Ti2	Td2
Ganancia	0.13	1.93	2.22	0.08	10.04	0.67
m0	-1.00	-1.00	-1.00	-1.00	-1.00	-1.00
m1	-0.67	-0.87	-0.46	-0.78	-0.98	-0.81
m2	-0.41	-0.86	-0.38	-0.62	-0.97	-0.37
m3	-0.17	-0.20	-0.13	-0.32	-0.73	-0.09
m4	0.00	0.00	0.00	0.00	0.00	0.00
m5	0.17	0.20	0.13	0.32	0.73	0.09
m6	0.41	0.86	0.38	0.62	0.97	0.37
m7	0.67	0.87	0.46	0.78	0.98	0.81
m8	1.00	1.00	1.00	1.00	1.00	1.00

tos para cada uno de los casos. Es importante recalcar en este punto que el algoritmo solamente ajustó los conjuntos difusos de la salida como puede observarse. Este ajuste es el responsable de generar el efecto no lineal sobre la estructura del PID que permite obtener respuestas que normalmente no podrían aparecer. Lo referente a la sintonización de sus constantes proporcionales, derivativas e integrales, se tomaron las mismas que para el caso del PID convencional sintonizado mediante el Recocido Simulado y no sufrieron ninguna modificación.

## 6.4. Análisis de los resultados

Los resultados de las pruebas realizadas se resumen en las tablas 6.17 al 6.20. En ellos se presenta el valor que obtuvo la función de costo como resultado ante los cambios de escalón realizados. Para revisar el valor específico de los indicadores de desempeño involucrados, deberá de revisarse el apéndice A. Se presentan para los casos en que la matriz de funciones de transferencia tienen uno o dos signos negativos.

Es importante especificar que una comparación directa de los costos obtenidos originalmente es imposible. Esto se debe a que el algoritmo de optimización calcula el valor de las constantes de la función de costo para cada caso en particular, por lo tanto, éstas dependen del tipo de controlador sometido a prueba y del valor aleatorio que se utilizó para determinar el estado con el que se hace la medición. Por ende, se consideró que la mejor manera de establecer un punto de comparación era tomar las cantidades de los indicadores de interés ponderados por una constante unitaria, de esta manera sería posible observar todos los resultados bajo la misma perspectiva.

TABLA 6.17: Resultados de Pruebas de robustez. Función de costo para caso de 2 signos.

	Ideal	K+	K-	T+	T-	Q+	Q-
PID (G)	129.2	158.4	181.1	163.4	Inestable	124.7	139.8
PID (RS)	87.9	84.6	162.1	169.7	147.9	97.5	93.7
Difuso MIMO	111.9	154.9	165.6	209.6	C.Estable	148.9	145.9
2 Difusos SISO	88.7	172.5	152.9	174.8	Inestable	214.7	144.7
3 Difusos (PID difuso)	85.7	102.1	168.3	167.0	182.1	117.7	87.3
3 Difusos* (PID difuso)	76.67	84.94	135.43	86.61	91.27	113.76	78.05

Los datos presentados son el resultado de aplicar los experimentos descritos anteriormente. Como primera instancia, se aprecia que el caso de un solo signo negativo resultó más fácil de controlar que el caso con dos signos, cuyo peor caso resultó en controladores inestables. Esto se puede observar de la inspección de las tablas 6.17, 6.18, 6.19 y 6.20. En estas tablas lo que se muestra es el costo total ponderado de forma unitaria de cada una de las pruebas realizadas. Para conocer los valores de cada indicador es necesario consultar el apéndice A. Para apreciar las respuestas directamente

TABLA 6.18: Resultados de Pruebas de robustez. Función de costo para caso de 2 signos (Continuación)

	Todos +	Todos -	Peor Caso
PID (G)	141	133.9	Inestable
PID (RS)	113.4	92.8	Inestable
Difuso MIMO	216.3	117.2	Inestable
2 Difusos SISO	283.1	77.2	Inestable
3 Difusos (PID Difuso)	140.6	188.8	Inestable
3 Difusos* (PID Difuso)	114.16	91.16	Inestable

TABLA 6.19: Resultados de Pruebas de robustez. Función de costo para caso de 1 signo

	Ideal	K+	K-	T+	T-	Q+	Q-
PID (G)	187.4	164.7	180.9	268.4	118.2	219.6	172.8
PID (RS)	94.1	123.6	228.6	134.6	161.7	141.7	128.1
Difuso MIMO	213.4	264.1	279.3	344.7	186.3	298.2	238.9
2 Difusos SISO	238.4	242.4	238.6	363.9	124.4	269.4	240.9
3 Difusos (PID Difuso)	85.2	120.2	191.3	136.7	109.6	141.6	97.7

TABLA 6.20: Resultados de Pruebas de robustez. Función de costo para caso de 1 signo (Continuación)

	Todos +	Todos -	Peor Caso
PID (G)	303.8239	241.2515	164.4248
PID (RS)	169.401	281.0197	226.6564
Difuso MIMO	528.9046	218.581	407.8928
2 Difusos SISO	Inestable	Inestable	Inestable
3 Difusos (PID Difuso)	172.0158	240.339	254.2546

se puede consultar el apéndice B.

De los resultados obtenidos se pueden obtener conclusiones interesantes. El primero de ellos es que los controladores basados en el desacoplador son los que ofrecieron el mejor desempeño en lo que se refiere al manejo del acoplamiento bajo las condiciones simuladas. En este caso, el único exponente que no estaba basado en la idea del desacoplador consiste es el controlador difuso multivariable. Aunque su IAE para el acoplamiento es relativamente bajo, el resultado obtenido en las simulaciones de las estrategias basadas en desacoplador fueron mucho menores. Esto refuerza la teoría de que puede tener mejor desempeño una estrategia basada en varios controladores que un solo controlador que se encargue de todas las acciones. En lo que respecta al resto de los indicadores de desempeño, puede verse que los que fueron más decisivos en la sintonización fueron los IAE y los tiempos de establecimiento. A pesar de que en el momento de correr los algoritmos se les daba mayor peso al porcentaje de sobretiro (Con el fin de compensar su magnitud en relación a los otros parámetros). Parece que para disminuir la función de costo fue a final de cuentas más conveniente hacer un controlador más rápido que un controlador con menor sobretiro. De observar los resultados del apéndice A, se observará que la magnitud del parámetro de sobretiro ( $M_p$ ) es variable en los diferentes casos expuestos. A pesar de ello, el controlador difuso MIMO tiene la tendencia a ofrecer manipulaciones más suaves que el resto de las estructuras, solo que la función de costo penaliza esto fuertemente, interpretándolo como falta de velocidad.

Respecto a la comparación de los PID, puede verse que la estrategia sometida al proceso del recocido simulado tiene un desempeño superior a su configuración inicial. Puede verse que la estrategia de 1 difuso logra a superar a las convencionales en el caso de 2 signos, pero no en el segundo caso. Lo que es importante de notar es que la estrategia basada en los 3 difusos logra el menor costo en condiciones ideales. Debe de recordarse que en los casos mostrados solo se modificaron las posiciones de los singleton de salida sin variar la sintonización. Esto demuestra que es posible lograr una mejoría en el desempeño mediante el uso de esta técnica. Analizando el resto de las simulaciones podemos ver que el desempeño ante la variación de parámetros se empobrece, siendo superior el PID sintonizado con el recocido simulado en el primero (contando la cantidad de casos en que uno superó al otro), pero en el caso de un solo signo negativo, la situación se invierte.

La conclusión a la que se llega respecto a la estrategia basada en PID de 3 difusos por lazo es que ésta ofrece el mejor desempeño bajo condiciones ideales, para mostrar ello se presenta en el caso de dos signos también una sintonización denotada como 3 Difusos. Como se puede ver, en este caso se logró el mínimo de los costos de todos los anteriores. Igualmente, en el caso de un solo signo negativo se determinó que en condiciones ideales esta estrategia es la mejor. Lo que resalta este resultado fue que a pesar de ser el mínimo en condiciones controladas, en el momento de presentarse

la variación de parámetros presentó sensibilidad. Esto lleva a pensar que la sintonía propuesta va haciendo al controlador mucho más a la medida del proceso en particular. Aunque se demostró que su robustez es más baja que el resto de las estrategias, se ha demostrado que con una cuidadosa sintonización es posible que esta estrategia de el mejor desempeño posible. Hablando en generalidad, puede decirse que la estructura de PID y PID difuso son las que presentan el mejor desempeño. Esto queda asentado en las gráficas de resultado 6.1 y 6.2. En estas gráficas lo que se hizo fue cuantificar el lugar que cada controlador obtuvo en las pruebas realizadas y se sumó el valor de las posiciones. Por lo tanto, entre menor es el número calculado, mejores posiciones alcanzó el controlador. Es claro de ver en estas gráficas que el PID ajustado mediante el recocido simulado y su contraparte difusa fueron los que obtuvieron en la mayoría de los casos las mejores posiciones.

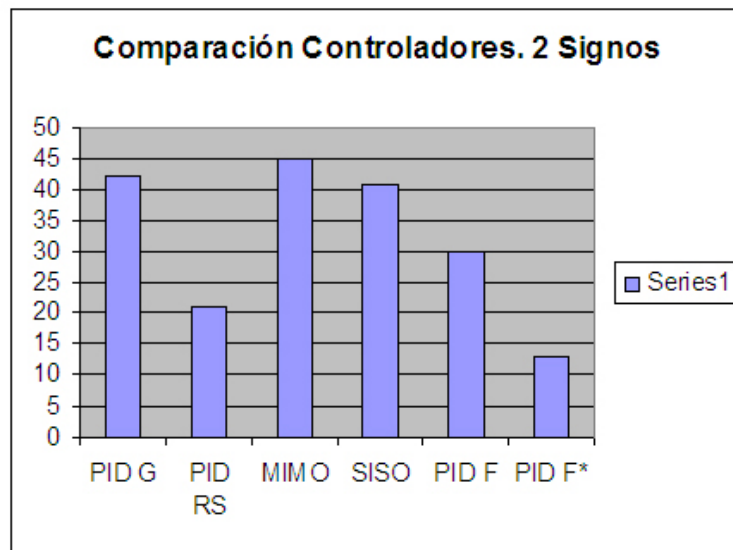


FIGURA 6.1: Calidad de éxito de controlador de 2 signos.

Hay algunos puntos que deben de ser considerados para la interpretación de estos resultados. El primero de ellos es que aunque unos controladores tengan un mayor o menor costo en su condición inicial. Esto se debe a que el algoritmo optimizó el controlador para el problema proporcionado y las constantes de ponderación proporcionadas para cada caso. Por lo tanto, probar con otra ponderación es factible y también es probable que se encuentren resultados diferentes y quizás mejores. Esto es un punto que se puede tomar en cuenta en las recomendaciones para trabajos futuros.

El segundo punto importante es que el controlador difuso multivariable se encontró en cierta desventaja respecto a las otras estrategias de control propuestas, dado que fue el único de ellos que tuvo que realizar tanto la acción de control como la compensación del acoplamiento, esto se refleja en los resultados presentados en el apéndice A

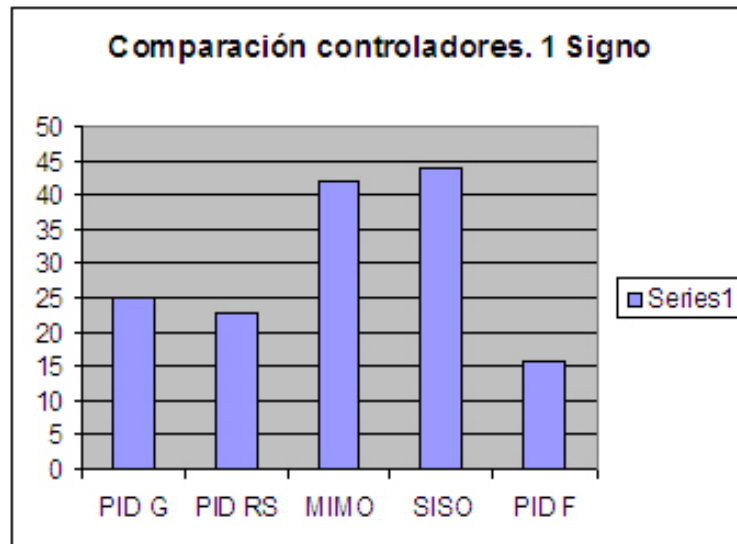


FIGURA 6.2: Calidad de éxito de controlador de 1 signo.

donde se muestra que esta estrategia fue la que mayor acoplamiento mostró. Sin embargo, de analizar estos datos, puede verse que esta estrategia ofrece respuestas más suaves con menor costo de manipulación, lo cual puede ser de interés en aplicaciones donde esto puede significar un ahorro o donde la velocidad del control no es un parámetro crítico de desempeño. Comparaciones con otros esquemas multivariables pueden ser de interés.

El tercer punto consiste que la validez de estos resultados se limita al caso lineal, dado que los controladores fueron optimizados y probados ante este tipo de modelos exclusivamente. Esto ofrece ventaja al PID dado que está diseñado para esta clase de situaciones. Es muy probable que el desempeño de los controladores difusos mejore al momento de evaluarlos ante sistemas no lineales donde las dinámicas no modeladas y otros efectos hagan cambiar el comportamiento del proceso. Esto se hizo de esta manera para seguir la línea de investigación planteada por los trabajos anteriores.

Por otra parte, el experimento denotado como 3Difusos\* arroja otro resultado interesante. Si se observa con cuidado, puede verse que es el que mejores resultados obtiene, solo siendo inferior al PID con desacoplador sintonizado con el recocido simulado en el caso en que se aumenta solamente el tiempo muerto de los procesos, indicando que para ese caso, un PID lineal es la mejor opción. En esta prueba lo que se hizo fue correr el algoritmo de sintonización para buscar el mejor resultado posible, pero sin cambiar en absoluto las ganancias proporcional, integral o derivativa ni tampoco recalculando los desacopladores para cada variación de parámetros. Solo se modificaron las posiciones de los conjuntos difusos de salida, conservando el resto de los parámetros del controlador constantes. Al hacer esto se demostró que a pesar de haber cambiado el modelo, el mero ajuste de estos conjuntos permitió lograr un buen desempeño en



la mayoría de los casos, y muchas veces el mejor de todos ellos. La conclusión que arroja este experimento es que dicha estructura puede ser utilizada para hacer un tipo de control adaptable, donde no es necesario modificar las ganancias que el usuario está acostumbrado a ver en un controlador convencional, pero que serán modificadas de un modo no lineal por el algoritmo adaptable para dar un mejor desempeño. Otra forma de ver esta posibilidad es la de utilizar un esquema supervisorio como es el de “Gain Scheduling”, solo que en lugar de modificar la sintonización del controlador, se modificarían la posición de los conjuntos. Un detalle que cabe señalar es que estas variaciones no son lineales, es decir, no pudo haberse logrado el mismo efecto por el hecho de multiplicar la entrada o la salida del controlador por una ganancia fija. A su vez, estos cambios no aumentan o disminuyen la ganancia del controlador de forma lineal, dado que los cambios fueron hechos dentro del espacio normalizado y las posiciones no cumplen con ninguna proporción en particular. En los experimentos realizados no se puede determinar exactamente el efecto que tiene el haber modificado las posiciones de estos conjuntos difusos de la manera en que se colocaron, dado que estos ajustes fueron realizados de manera aleatoria por el algoritmo de optimización. El estudiar los efectos de mover no linealmente las posiciones de estos conjuntos así como diseñar un mecanismo de adaptación en base a las conclusiones obtenidas puede ser un tema digno de investigación.

Respecto al trabajo demostrado en referencias anteriores, se demostró que existe todavía mejores formas de sintonizar un controlador PID para que éste de un buen desempeño. No es posible hacer una comparación directa entre los resultados anteriores y los obtenidos en otros trabajos. Lo que puede decirse es que para una estrategia de dos controladores univariados aunados a un desacoplador la técnica de PIDDE propuesta en [3] funciona mejor que la propuesta en este trabajo, dado que contempla las tres componentes del PID mientras que lo expuesto aquí solo presenta 2. La comparación directa entre el PIDDE y la estrategia basada en 3 difusos queda como una interrogante, dado que habría de programarse tanto la estructura de control como el método de sintonización para hacer una comparación directa. Otra cuestión que elimina la posibilidad de una comparación indirecta es que tanto una como la otra superan al PID sintonizado con las reglas obtenidas mediante programación genética, lo cual solo indica que para conocer la verdad de estos dos controladores es necesario indagar con mayor profundidad. Ésta es una de las recomendaciones que se harán como guías para investigaciones venideras.

La eficacia del método de sintonización se demostró al presentar los resultados de la comparación entre los controladores propuestos en [22] y el propuesto en el trabajo de tesis sintonizado mediante el recocido simulado. Se observó como la técnica, partiendo de un punto que puede llamarse neutral, generó una sintonización que superó a la de el algoritmo publicado. A su vez también se realizaron pruebas de mejorar el controlador difuso propuesto en el artículo citado mediante el recocido simulado y

logró un controlador más fino. Los resultados obtenidos para el PID y obtenidos por la estrategia basada en 3 difusos muestran el poder de la técnica y su viabilidad para obtener soluciones sin la necesidad de hacer un estudio exhaustivo del proceso o de las estructuras de control.

## Capítulo 7

# Conclusiones y recomendaciones

### 7.1. Conclusiones

A lo largo del trabajo presentado se trató el problema del control de procesos multivariantes acoplados mediante diversas técnicas. Inicialmente se explicó en que consisten esta clase de procesos y la problemática que encierran. Se declararon los objetivos de utilizar estrategias basadas en sistemas difusos y se explicó la metodología a seguir.

Se profundizó en la lógica difusa y algunos de sus principios. Se dio explicación detallada de las tres estrategias difusas a seguir: Controlador difuso multivariable, controlador difuso univariable con desacoplador y controlador PID difuso con desacoplador.

Se expuso la naturaleza del recocido simulado y los detalles concretos del algoritmo. Finalmente se aplicaron estos conocimientos para el diseño y sintonización de los controladores propuestos para finalmente compararlos contra la estrategia convencional de PID con desacoplador sintonizada por dos métodos diferentes, es decir, programación genética y el mejoramiento mediante el recocido simulado.

De la investigación realizada pudieron obtenerse varias conclusiones interesantes.

Una de las conclusiones es que el recocido simulado es una herramienta poderosa para la sintonización de controladores. Por un lado es un algoritmo relativamente sencillo que tiene un buen rendimiento de definir sus características apropiadamente. En este caso se decidió ponderar cada parámetro por igual para aumentar luego el peso del error final y el tiempo de establecimiento. Sin embargo, otras ponderaciones son posibles y pueden cambiarse de acuerdo del interés de la aplicación en particular.

A lo largo de esta tesis se utilizó para obtener la sintonía óptima de diferentes estrategias de control y se llegaron a obtener buenos resultados. Para que pueda alcanzar el mínimo global de la función de optimización es necesario permitirle hacer las suficientes iteraciones con los suficientes decrementos de temperatura. Básicamente, entre más intentos realice, en mayor medida se aproxima al óptimo global. Otra de las ventajas que pueden decirse al respecto, es que exenta al usuario de tener un conocimiento profundo del proceso y de la técnica de control. Obviamente, ciertas nociones

son necesarias en el momento del diseño. Pero como puede verse, la aplicación de esta clase de algoritmos ahorran el esfuerzo de ajustar un sistema complejo como puede ser uno difuso mediante el método de prueba y error, permitiendo a su vez obtener controladores óptimos a la luz de la función de costo definida.

Otra de las conclusiones alcanzadas es la noción de que puede ser mejor establecer una estrategia de varios controladores que realicen un mismo fin, en lugar de construir un solo controlador que deba batallar con numerosos condicionamientos. Esto se aprecia fácilmente al comparar el desempeño de estrategias como el PID difuso contra el controlador difuso multivariable. En el caso del PID difuso, el acoplamiento y los efectos de control fueron realizados por bloques separados. Sin embargo, no se puede menospreciar la calidad del controlador difuso multivariable, el cual logra una buena respuesta a pesar de tener que lidiar con todos los condicionamientos de manera simultánea. La ventaja que ofreció este tipo de controladores fue una respuesta más suave que el resto de los controladores, aunque en la mayoría de los casos implicó una respuesta más lenta. Un controlador difuso multivariable puede ser entonces útil para ciertas aplicaciones, sobre todo en aquellas donde se desea hacer el menor esfuerzo de control para conseguir algún ahorro. Por ende, puede realizarse más investigación de esta estrategia y compararla contra alguna otra de naturaleza similar para indagar en mayores resultados.

Algo más que puede decirse respecto a los controladores difusos es que estos sistemas son herramientas de control no lineal y en sus no linealidades se basa su verdadero potencial. Como se observó durante los experimentos realizados, el PID difuso logra un mejor ajuste que el resto de los controladores sometidos a comparación cuando se hace uso de estas propiedades. Cabe mencionar que a su vez esto hizo al algoritmo el más sensible de todos una vez que fueron ajustadas las posiciones de sus conjuntos difusos de salida, pero a su vez se demostró que de tener un mecanismo de adaptación de estos parámetros, una misma sintonización de PID puede dar un buen resultado aún cuando se presenten cambios en el proceso a controlar.

A la luz de los experimentos que se realizaron con el algoritmo del recocido simulado se mostró que la sintonización de controladores convencionales es todavía un campo donde se pueden lograr avances. Los experimentos mostraron que este controlador puede obtener todavía mejores resultados si se ajusta correctamente. El PID es el algoritmo más común en la industria y esto se basa en su sencillez y su eficiencia. Invertir recursos en lograr mejor formas de ajustarlo es, entonces, una veta por explotar.

Entre los avances que presenta esta tesis es que se evaluaron diferentes estructuras de controladores difusos y fueron evaluadas bajo un mismo criterio de costo. Se aplicaron métodos de optimización como fue sugerido en [24] y se dejó de recurrir entonces a métodos como el de prueba y error. Respecto al método reportado en [3], la ventaja principal es que el método de optimización no es exhaustivo, por lo que el tiempo de ejecución es menor. Esto no puede cuantificarse directamente, dado que las plataformas

en las que fueron programados los algoritmos respectivos son muy distintas. Además, se puede decir que otro avance alcanzado es que se encontró que el recocido simulado es también útil para mejorar la sintonización de controladores convencionales, ofreciendo una opción para mejorar las sintonizaciones disponibles en caso de que el operador no las encuentre satisfactorias.

Finalmente, cabe agregar que bajo las condiciones aplicadas en este trabajo los mejores controladores fueron el PID optimizado y su contraparte difusa. En parte esto era de esperarse porque se trabajó con procesos lineales donde el PID goza de las mayores ventajas. Pruebas contra modelos no lineales son el siguiente paso lógico en la línea de investigación para poder evaluar de una manera completa el desempeño de los controladores propuestos.

## 7.2. Recomendaciones para investigaciones futuras

A la luz del trabajo realizado puede decirse que aún queda mucho por hacer en esta línea de investigación. En los experimentos realizados, dado que el recocido simulado modificó los parámetros de forma aleatoria hasta encontrar el mínimo posible, no fue posible conocer con profundidad el efecto de mover estos parámetros en las estructuras propuestas. En parte, este nivel de desconocimiento fue inducido e intencionado, porque el objetivo de aplicar un algoritmo como el recocido simulado fue precisamente exentar al usuario de requerir tal grado de calificación para ajustar los controladores. Sin embargo es un estudio interesante conocer más a fondo las estructuras propuestas, sus limitaciones y sus ventajas latentes.

Referente a continuaciones del trabajo realizado, se puede explorar con otras funciones de optimización o incluso modificarla para cada caso. Lo que se propone es que se experimente con las constantes de ajuste, buscando las más adecuadas para cada situación. Esto se hace con el fin de encontrar la mejor función de costo para lograr el mejor resultado, dado que el Recocido Simulado es un algoritmo ciego que solo puede optimizar en base a los lineamientos que encuentra establecidos.

Otra forma de continuar con el trabajo realizado es aplicar pruebas sobre modelos no lineales, dado que la validez de los resultados alcanzados se limita al caso lineal donde el PID tiene enormes ventajas. De esta manera es posible observar otros comportamientos beneficiosos de los controladores difusos que no se pueden evaluar bajo otras condiciones.

A parte de las estructuras propuestas, existen un sinnúmero de combinaciones posibles. En estos trabajos solo se estudiaron una pequeña fracción de toda la gama disponible. Repetir los experimentos reportados en este documento con otras estrategias de controladores difusos seguramente ayudará a determinar algoritmos más eficientes y precisos. Por ejemplo, en este trabajo ya no se tuvo la oportunidad de evaluar el desempeño

del PIDDE reportado en [3] frente a las estrategias analizadas. A su vez, tampoco fue posible aplicarle el recocido simulado para conocer si realmente se tiene ajustado lo mejor posible. Estas interrogantes pueden ser despejadas en trabajos venideros.

La estructura de PID difuso propuesta como controlador puede ser explorada más a fondo. Como se mostró en las pruebas realizadas, tiene propiedades de adaptación que le permiten combatir sus debilidades consiguiendo resultados mucho mejores. Por este camino lo que faltaría hacer es determinar cual sería el mecanismo de adaptación apropiado para lograr lo que se sugiere. El recocido simulado podría ser un candidato si se ajusta apropiadamente, pero seguramente el uso de otras técnicas será una interesante área de oportunidad. Otra posibilidad consiste en ver este ajuste no como un mecanismo de adaptación, sino como un ajuste referente al punto de operación del proceso, caso que es típico del control supervisorio y de la técnica de “Gain Scheduling”. En este caso, se requerirá de un sistema supervisorio que se encargue de hacer el ajuste correspondiente en la posición de los conjuntos difusos del controlador.

Para terminar, también se ve como una posibilidad de investigación el buscar mejores maneras de sintonizar un controlador PID, dado que en este trabajo se demostró que éste puede dar aún más de lo que se esperaba. Aplicación de otras técnicas a parte de las mencionadas en esta tesis pueden ser tema de interés para trabajos posteriores.

## Bibliografía

- [1] Bretado A. Control de presión y flujo de inyección de oxígeno en la estación de recalentamiento químico en ahmsa usando lógica difusa. Master's thesis, ITESM, Departamento de Mecatrónica y Automatización, Mayo 2002.
- [2] Buelna C and Soto R. Pc based fuzzy-neuro controller for a synchronous generator. In *Proceedings of the American Control Conference.*, 1999.
- [3] Palma C. Controlador difuso que mejora a un pid mediante el ajuste de sus bases. Master's thesis, ITESM, Departamento de Mecatrónica y Automatización, Agosto 2005.
- [4] Smith C and Corripio A. *Principles and practice of automatic process control.* Wiley, USA, 1997.
- [5] Villa C. Controlador difuso en una planta multivariable dos entradas dos salidas. Master's thesis, ITESM, Departamento de Mecatrónica y Automatización, Mayo 2000.
- [6] Driankov D. *An introduction to fuzzy control.* Springer-Verlag, USA, 1996.
- [7] Kirk D. *Optimal Control Theory. An Introduction.* Electric Engineering Series. Prentice Hall, USA, 1970.
- [8] Doebelin. *Instrumentation and measurement systems.* Pearson Education, USA, 1996.
- [9] Aarts E and Korst J. *Simulated Annealing and Boltzmann Machines.* John Wiley and Sons, USA, 1989.
- [10] T Fukuda, H Ishigami, T Shibata, and F Arai. Auto fuzzy tuning having minimum structure by using genetic algorithm and delta rules. In *Emerging Technologies and Factory Automation, 1993. Design and Operations of Intelligent Factories. Workshop Proceedings IEEE 2nd International Workshop on*, 1993.

- [11] S Guan, H Li, and Tsao S. Multivariable fuzzy supervisory control for the laminar cooling process of hot roller slab. *IEEE Transactions on Control Systems Technology*, 9(2):348–356, 2001.
- [12] N Hazem and K Passino. Stable auto-tuning of adaptive fuzzy/neural controllers for nonlinear discrete-time systems. *IEEE Transactions on Fuzzy Systems*, 12(01):70–83, 2004.
- [13] H Hong, S Park, S Han, and Cho K. A design of auto-tuning pid controller using fuzzy logic. In *Industrial Electronics, Control, Instrumentation, and Automation, 1992. 'Power Electronics and Motion Control, 1992*.
- [14] Juárez I. Diseño de un controlador desacoplante y dos difusos para una planta multivariable de dos entradas y dos salidas. Master's thesis, ITESM, Departamento de Mecatrónica y Automatización, Julio 2003.
- [15] Lopez I. Optimización de ecuaciones para autosintonía de controladores pid mediante programación genética. Master's thesis, ITESM, Departamento de Mecatrónica y Automatización, Junio 1998.
- [16] Galván J.A. Síntesis de mecanismos de cuatro barras para n puntos de precisión con restricciones de espacio empleando el algoritmo de recocido simulado. Master's thesis, ITESM, Maestría con especialidad en sistemas inteligentes, Julio 2000.
- [17] Ogata K. *Sistemas de Control en Tiempo Discreto*. Prentice Hall, USA, 1999.
- [18] Rosas L. Metodología para la automatización del diseño de controladores rst robustos. Master's thesis, ITESM, Maestría en Ciencias con especialidad en Automatización, Julio 2002.
- [19] Wang L. *A course in fuzzy systems and control*. Prentice Hall, USA, 1997.
- [20] Ruiz M. Diseño de sistemas de control óptimo en tiempo discreto utilizando recocido simulado. Master's thesis, ITESM, Maestría con especialidad en sistemas inteligentes, Julio 2003.
- [21] Viljamaa P. *Fuzzy Gain Scheduling and tuning of multivariable fuzzy control. Methods of fuzzy computing in control systems*. PhD thesis, Tampere University of Technology, Junio 2000.
- [22] Viljamaa P and Koivo H. Tuning of multivariable fuzzy logic controller. *Fuzzy Systems, 1993., Second IEEE International Conference on*, 2:697–701, 1993.
- [23] Diaz-Flores R. Control difuso con capacidad de emulación pid. Master's thesis, ITESM, Departamento de Mecatrónica y Automatización, Mayo 2001.



- [24] Ruíz R. Diseño de un controlador difusa multivariable para una planta de dos entradas dos salidas. Master's thesis, ITESM, Departamento de Mecatrónica y Automatización, Mayo 1997.
- [25] R Ruiz-Leon and Rodriguez-Ortiz J. Fuzzy control of a multivariable coupled process. *ISA Thechnical Conference*.
- [26] Y Shi, M Mizumoto, N Yubazaki, and M Otani. A learning algorithm for tuning fuzzy rules based on gradient descent method. *Computational Intelligence for Measurement Systems and Applications*, XX(XX):55–61, 2003.

## Apéndice A

### Tablas de resultados en pruebas de desempeño

#### A.1. Presentación de resultados. Caso de dos signos distintos

Se presentan los resultados obtenidos de los experimentos. En las tablas siguiente se muestran los parámetros de desempeño obtenidos en cada una de las pruebas realizadas. En el capítulo correspondiente a la presentación de resultados se realiza un análisis de los valores obtenidos.

##### Prueba en condiciones ideales.

En esta parte se presentan los indicadores de desempeño de las pruebas de robustez realizadas.

TABLA A.1: Prueba de Desempeño. Indicadores de desempeño en condiciones ideales (Lazo1)

	IAE	IAED	Ts	Mp	SU
PID (G)	9.4508	0	57	0.0494	1.6518
PID (RS)	6.9116	0	26	0.2021	1.2284
1 Difuso	12.3036	4.2148	37	0.063	0.2814
2 Difusos	9.7068	0	24	0.4031	0.5636
3 Difusos	7.1529	0	22	0.3026	1.1865
3 Difusos*	7.5756	0	23	0.3077	1.1307

TABLA A.2: Prueba de Desempeño. Indicadores de desempeño en condiciones ideales (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	10.8257	0	49	0.1257	1.1434
PID (RS)	12.3239	0	41	0.0009	0.3023
1 Difuso	18.454	2.4892	37	0.019	0.1135
2 Difusos	13.7933	0	39	0.7499	0.4913
3 Difusos	11.7453	0	43	0	0.3394
3 Difusos*	10.2866	0	34	0.0725	1.1307

### Pruebas de sensibilidad ante cambio de parámetros

En esta parte se presentan los indicadores de desempeño que se obtuvieron en las pruebas de variación de parámetros.

TABLA A.3: Prueba de Robustez. Aumento de 50% en las ganancias del proceso. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)	7.3528	0	50	0.1251	2.3442
PID (RS)	6.1026	0	23	0.4199	1.5889
1 Difuso	10.3792	4.6003	39	0.0688	0.281
2 Difusos	12.7465	0	51	0.7013	0.7808
3 Difusos	7.1448	0	25	0.591	1.2868
3 Difusos*	6.8921	0	22	0.4253	1.4575

TABLA A.4: Prueba de Robustez. Aumento de 50% en las ganancias del proceso. (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	14.1243	0	82	0.474	2.0635
PID (RS)	9.9732	0	43	0.1335	0.4257
1 Difuso	14.9184	2.5119	83	0.0291	0.1576
2 Difusos	26.7598	0	78	1.4746	1.04
3 Difusos	11.3635	0	56	0.1951	0.5197
3 Difusos*	9.6645	0	44	0.1092	0.3928

TABLA A.5: Prueba de Robustez. Disminución de 50% en las ganancias del proceso. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)	15.7972	0	76	0.0307	1.2672
PID (RS)	10.4662	0	42	0.1653	1.1768
1 Difuso	17.255	3.5064	53	0.0583	0.4489
2 Difusos	12.6969	0	65	0.1836	0.6928
3 Difusos	10.2539	0	38	0.1942	1.3214
3 Difusos*	8.9088	0	34	0.2008	1.2522

TABLA A.6: Prueba de Robustez. Disminución de 50% en las ganancias del proceso (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	16.1	0	71	0.0577	0.8389
PID (RS)	24.0431	0	84	0	0.3111
1 Difuso	27.2415	2.9079	61	0.0134	0.2097
2 Difusos	11.89	0	62	0.1185	0.3615
3 Difusos	21.1856	0	97	0	0.354
3 Difusos*	20.5715	0	70	0.0822	0.4186

TABLA A.7: Prueba de Robustez. Aumento de 50% en las constantes de tiempo. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)	13.8013	0.9787	68	0.1189	1.3993
PID (RS)	10.2218	1.2791	53	0.2803	1.2567
1 Difuso	15.8942	6.3538	68	0.1454	0.4234
2 Difusos	13.7716	1.4041	60	0.3612	0.7603
3 Difusos	11.6887	1.4734	56	0.3365	1.4
3 Difusos*	8.2332	0.9129	25	0.2705	1.2027

TABLA A.8: Prueba de Robustez. Aumento de 50% en las constantes de tiempo (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	14.9691	1.0604	62	0.1642	0.9235
PID (RS)	16.0935	0.323	87	0.0472	0.284
1 Difuso	24.2598	3.244	91	0.1232	0.1564
2 Difusos	16.946	0.7555	80	0.407	0.0024
3 Difusos	14.3712	0.3781	81	0.045	0.3312
3 Difusos*	13.4543	0.2259	37	0.0105	0.305

TABLA A.9: Prueba de Robustez. Disminución de 50 % en las constantes de tiempo (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)					Inestable
PID (RS)	7.43	0.8796	45	0.3508	2.2051
1 Difuso					C.estable
2 Difusos					Inestable
3 Difusos	7.1418	0.9255	44	0.4399	2.035
3 Difusos*	5.8175	0.856	12	0.5106	1.2589

TABLA A.10: Prueba de Robustez. Disminución de 50 % en las constantes de tiempo (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)					Inestable
PID (RS)	14.899	0.2924	76	0.2102	0.6457
1 Difuso					C estable
2 Difusos					Inestable
3 Difusos	22.5156	0.462	103	0.5924	1.0811
3 Difusos*	13.7548	1.1501	55	0.2448	0.6779

TABLA A.11: Prueba de Robustez. Aumento de 50 % en el tiempo muerto. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)	9.1886	2.376	54	0.0486	1.3362
PID (RS)	7.7149	4.4161	21	0.363	1.0098
1 Difuso	13.025	6.7012	57	0.0856	0.2914
2 Difusos	14.6944	10.02	74	0.6042	0.8608
3 Difusos	10.2863	5.2614	36	0.5024	1.1506
3 Difusos*	8.163	3.5559	36	0.2466	1.2334

TABLA A.12: Prueba de Robustez. Aumento de 50 % en el tiempo muerto. (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	10.6151	1.1791	45	0.1632	0.8854
PID (RS)	12.3357	1.3402	49	0.0073	0.346
1 Difuso	18.3222	3.3214	50	0.0654	0.1499
2 Difusos	26.0171	7.49	79	1.2492	0.015
3 Difusos	12.5552	1.4804	50	0.1173	0.3832
3 Difusos*	12.9537	1.2326	50	0.0199	0.359

TABLA A.13: Prueba de Robustez. Disminución de 50 % en el tiempo muerto. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)	10.0298	3.0434	58	0.0732	1.5114
PID (RS)	6.9578	3.1953	30	0.1925	1.0153
1 Difuso	11.9253	2.6364	35	0.0475	0.3034
2 Difusos	9.48	4.548	50	0.231	0.5314
3 Difusos*	6.63	3.45	27	0.2158	1.059

TABLA A.14: Prueba de Robustez. Disminución de 50 % en el tiempo muerto. (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	11.4209	1.6314	53	0.1344	1.0071
PID (RS)	12.5035	0.6264	39	0.0043	0.2574
1 Difuso	18.993	1.8924	75	0.0375	0.1193
2 Difusos	11.8483	2.441	65	0.29	0.4133
3 Difusos	10.84	0.8395	37	0.0027	0.2773
3 Difusos*	10.9617	0.7209	27	0.019	0.2827

TABLA A.15: Prueba de Robustez. Aumento de 50 % en los parámetros. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)	10.2377	3.5237	55	0.1182	1.3516
PID (RS)	8.7498	5.944	21	0.4553	1.0607
1 Difuso	14.0049	10.1007	78	0.1597	0.3787
2 Difusos	18.31	15.55	98	0.7471	1.066
3 Difusos	12.6683	7.3995	40	0.5945	1.2273
3 Difusos*	8.6422	3.8634	33	0.2319	1.2928

TABLA A.16: Prueba de Robustez. Aumento de 50 % en los parámetros. (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	12.3658	1.2575	56	0.2546	0.8962
PID (RS)	12.2546	1.5364	62	0.1028	0.3547
1 Difuso	20.3194	4.075	89	0.1759	0.164
2 Difusos	32.86	10.18	104	1.49	0.97
3 Difusos	13.2681	1.8302	63	0.2411	0.4116
3 Difusos*	13.6208	0.95	52	0.1711	0.3939

TABLA A.17: Prueba de Robustez. Disminución de 50% en los parámetros. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)	13.9618	2.3771	55	0	1.4911
PID (RS)	5.737	2.2695	25	0.033	0.9182
1 Difuso	14.1518	0.9078	38	0	0.31
2 Difusos	6.61	2.69	27	0.02	0.364
3 Difusos	5.1826	2.8729	24	0.0452	0.9868
3 Difusos*	7.5756	0	23	0.3077	1.1307

TABLA A.18: Prueba de Robustez. Disminución de 50% en los parámetros. (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)	12.5331	1.5479	46	0	1.0022
PID (RS)	17.9389	0.4115	40	0	0.2959
1 Difuso	22.4808	1.0229	40	0	0.1757
2 Difusos	6.01	1.21	33	0.03	0.32
3 Difusos	22.3645	0.9859	132	0	0.3802
3 Difusos*	10.2866	0	34	0.0725	0.3032

TABLA A.19: Prueba de Robustez. El peor caso. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID (G)					Inestable
PID (RS)					Inestable
1 Difuso					Inestable
2 Difusos					Inestable
3 Difusos					Inestable
3 Difusos*					Inestable

TABLA A.20: Prueba de Robustez. El peor caso. (Lazo2).

	IAE	IAED	Ts	Mp	SU
PID (G)					Inestable
PID (RS)					Inestable
1 Difuso					Inestable
2 Difusos					Inestable
3 Difusos					Inestable
3 Difusos*					Inestable

## A.2. Presentación de resultados. Caso de un signo distinto

Se presentan los resultados obtenidos de los experimentos. En este caso se tomó el mismo proceso multivariable, pero se cambió para que tuviera un único signo negativo en  $K_{21}$ . Esto se hizo para que tuviera la dinámica relación con un proceso de nivel y temperatura, cuya lógica de funcionamiento es más intuitiva que el caso anterior. Como consecuencia del cambio, los lazos resultantes a la hora de aplicar el desacoplador son diferentes a los anteriores, por lo que se tienen pruebas de un proceso distinto. Esto se hizo con la intención de poder conocer un poco los efectos que tienen estos signos en el comportamiento del proceso.

### Prueba en condiciones ideales.

TABLA A.21: Pruebas En condiciones ideales.1 signo. (Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	17.5796	0	87	0.0694	0.6902
PID RS	9.4175	0	41	0.1004	0.3653
1 Difuso	30.6298	1.9522	121	0.3322	0.1484
2 Difusos	30.947	0	84	0.4214	0.4994
3 Difusos	9.1387	0	37	0.1295	0.386

TABLA A.22: Pruebas En condiciones ideales. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	17.9027	0	64	0.1081	0.0638
PID RS	14.2061	0	29	0	0.096
1 Difuso	14.4356	4.8506	40	0.0524	0.0714
2 Difusos	37.398	0	84	0.4477	0.18
3 Difusos	13.4687	0	25	0.0161	0.1107



TABLA A.23: Pruebas de Robustez. Aumento de 50% en las ganancias del proceso. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	13.2782	0	71	0.0826	0.8472
PID RS	9.3506	0	53	0.2832	0.4858
1 Difuso	32.5184	2.9718	168	0.4454	0.1598
2 Difusos	33.7978	0	84	0.5381	0.5168
3 Difusos	9.5798	0	51	0.3225	0.5041

TABLA A.24: Pruebas de Robustez. Aumento de 50% en las ganancias del proceso. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	16.236	0	63	0.2335	0.0718
PID RS	12.24	0	48	0.1335	0.117
1 Difuso	13.7146	5.9995	46	0.1957	0.0864
2 Difusos	38.4778	0	84	0.5741	0.1788
3 Difusos	12.5283	0	46	0.1671	0.1243

TABLA A.25: Pruebas de Robustez. Disminución de 50% en las ganancias del proceso. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	27.0282	0	67	0.0195	0.6103
PID RS	16.2424	0	86	0	0.261
1 Difuso	34.7921	1.1165	162	0.1575	0.168
2 Difusos	30.4193	0	84	0.2477	0.4596
3 Difusos	14.272	0	72	0	0.2863

TABLA A.26: Pruebas de Robustez. Disminución de 50% en las ganancias del proceso. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	26.1893	0	60	0.0175	0.0734
PID RS	28.0642	0	98	0	0.0948
1 Difuso	22.7725	4.2571	54	0.004	0.0788
2 Difusos	39.0009	0	84	0.2663	0.1769
3 Difusos	25.7297	0	79	0	0.1132

TABLA A.27: Pruebas de Robustez. Aumento de 50% en las constantes de tiempo. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	23.5504	0.506	103	0.1369	0.6304
PID RS	9.6634	0.5198	29	0.0598	0.3176
1 Difuso	44.0333	3.3167	199	0.4164	0.21
2 Difusos	40.6876	18.8762	124	0.4455	0.7393
3 Difusos	9.6904	0.5621	30	0.0855	0.3366

TABLA A.28: Pruebas de Robustez. Aumento de 50% en las constantes de tiempo. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	24.6162	0.7055	115	0.1923	0.0803
PID RS	18.5431	0.3538	76	0.0667	0.1011
1 Difuso	20.1883	7.3058	70	0.1421	0.08
2 Difusos	50.083	4.003	124	0.489	0.2682
3 Difusos	18.5734	0.3289	77	0.0854	0.1207

TABLA A.29: Pruebas de Robustez. Disminución de 50% en las constantes de tiempo. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	12.8634	0.2618	45	0	1.1144
PID RS	9.2297	0.5758	68	0.1843	0.4684
1 Difuso	18.5915	1.2614	86	0.1849	0.092
2 Difusos	17.8142	3.1052	40	0.3589	0.2272
3 Difusos	8.0772	0.4998	44	0.2245	0.5026

TABLA A.30: Pruebas de Robustez. Disminución de 50% en las constantes de tiempo. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	12.6029	0.3522	46	0	0.0534
PID RS	13.9216	0.2061	69	0	0.1104
1 Difuso	13.9928	3.1539	63	0	0.0748
2 Difusos	21.8257	0.1336	40	0.3545	0.0837
3 Difusos	12.003	0.1868	44	0	0.1186

TABLA A.31: Pruebas de Robustez. Aumento de 50% en el tiempo muerto. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	18.1259	1.4375	87	0.0784	0.5958
PID RS	10.6664	3.3296	64	0.1662	0.3998
1 Difuso	33.7448	4.5377	164	0.358	0.1763
2 Difusos	32.999	18.5643	84	0.4315	0.5103
3 Difusos	10.6725	3.7022	62	0.1985	0.4347

TABLA A.32: Pruebas de Robustez. Aumento de 50% en el tiempo muerto. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	22.2138	0.8658	89	0.2069	0.0797
PID RS	16.4951	0.5503	46	0.0781	0.1051
1 Difuso	20.1472	7.9548	67	0.2301	0.0957
2 Difusos	39.331	8.1621	84	0.5835	0.1804
3 Difusos	16.7775	0.6155	47	0.1102	0.1227

TABLA A.33: Pruebas de Robustez. Aumento de 50% en todos los parámetros. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	19.9006	2.7507	119	0.1663	0.6092
PID RS	11.3653	4.8714	54	0.27	0.4386
1 Difuso	51.2188	9.9368	292	0.5645	0.2672
2 Difusos					Inestable
3 Difusos	11.8015	5.6967	54	0.3043	0.4547

TABLA A.34: Pruebas de Robustez. Aumento de 50% en todos los parámetros. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	30.6572	1.1751	129	0.4301	0.1139
PID RS	20.3909	0.6516	77	0.2748	0.1267
1 Difuso	28.0245	14.307	132	0.455	0.1293
2 Difusos					Inestable
3 Difusos	21.5509	0.756	77	0.3083	0.1405

TABLA A.35: Pruebas de Robustez. Disminución de 50% en todos los parámetros. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	25.1782	0.6376	96	0	0.6139
PID RS	16.1982	1.2687	108	0	0.2917
1 Difuso	23.0684	0.2382	82	0.0288	0.112
2 Difusos					Inestable
3 Difusos	13.5749	1.4828	92	0	0.2922

TABLA A.36: Pruebas de Robustez. Disminución de 50% en todos los parámetros. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	24.8705	0.851	93	0	0.0682
PID RS	27.8176	0.4154	127	0	0.0983
1 Difuso	22.3719	1.6602	89	0	0.0679
2 Difusos					Inestable
3 Difusos	25.4287	0.4113	107	0	0.1186

TABLA A.37: Pruebas de Robustez. El peor caso. (1 signo,Lazo1).

	IAE	IAED	Ts	Mp	SU
PID gene	9.8837	3.9566	62	0.0635	1.2695
PID RS	14.0881	12.0461	97	0.2688	0.8181
1 Difuso	20.8491	8.0168	128	0.3289	0.1651
2 Difusos					Inestable
3 Difusos	15.0694	13.0108	122	0.2931	0.0177

TABLA A.38: Pruebas de Robustez. El peor caso. (1 signo,Lazo2).

	IAE	IAED	Ts	Mp	SU
PID gene	17.4155	1.3797	68	0.3467	0.0929
PID RS	15.5142	2.4267	84	0.3208	0.1654
1 Difuso	32.5775	15.2064	202	0.5141	0.2311
2 Difusos					Inestable
3 Difusos	15.7323	2.7011	84	0.368	0.1668

## Apéndice B

### Gráficas de pruebas de desempeño

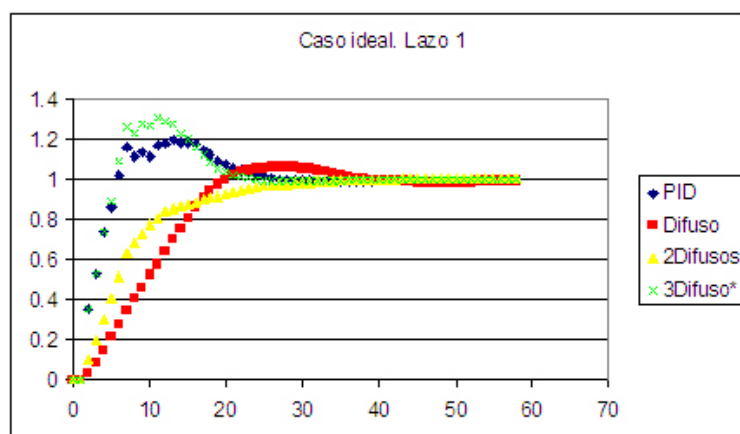


FIGURA B.1: Respuesta de caso ideal. Lazo 1

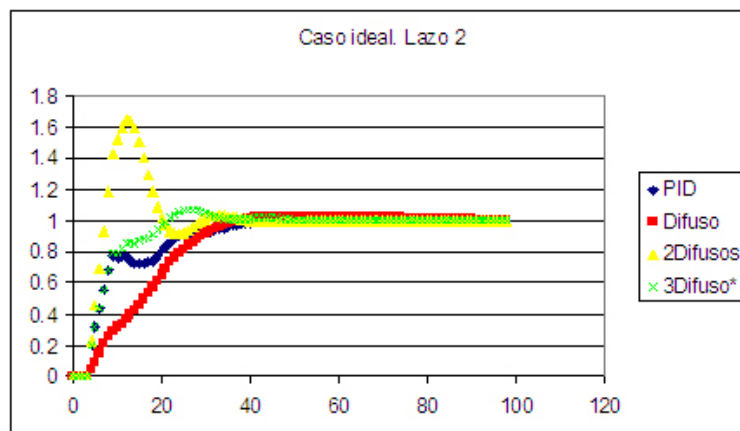


FIGURA B.2: Respuesta de caso ideal. Lazo 2

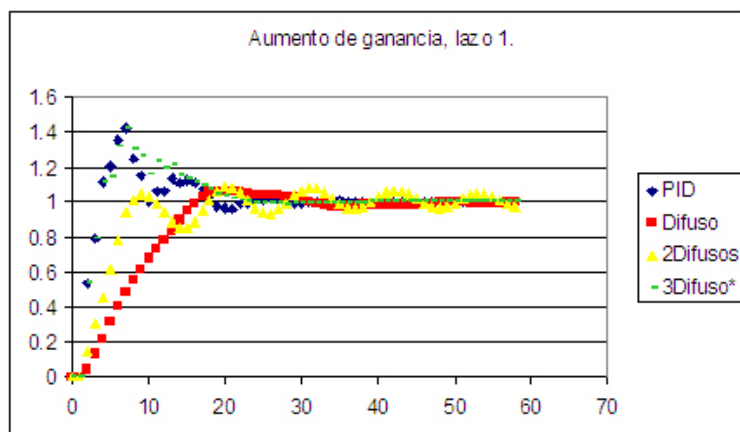


FIGURA B.3: Respuesta al aumento de ganancias. Lazo 1

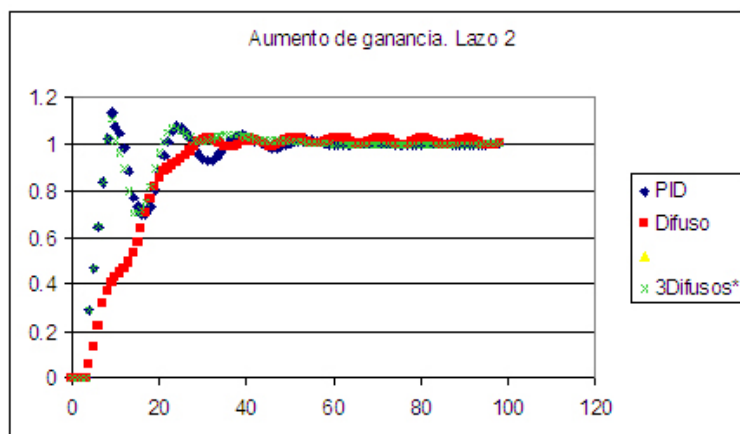


FIGURA B.4: Respuesta al aumento de ganancias. Lazo 2

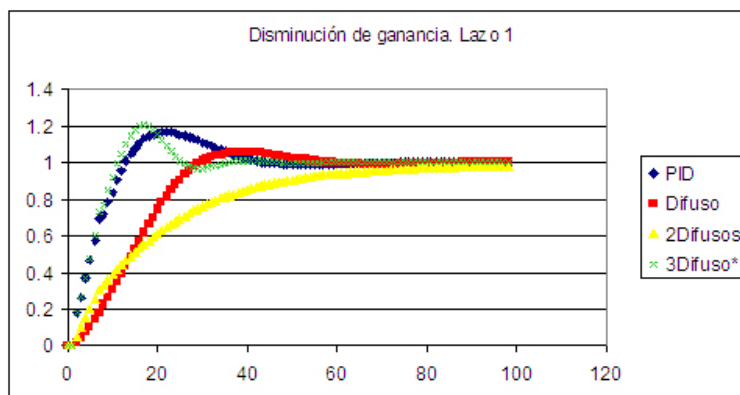


FIGURA B.5: Respuesta a la disminución de ganancias. Lazo 1

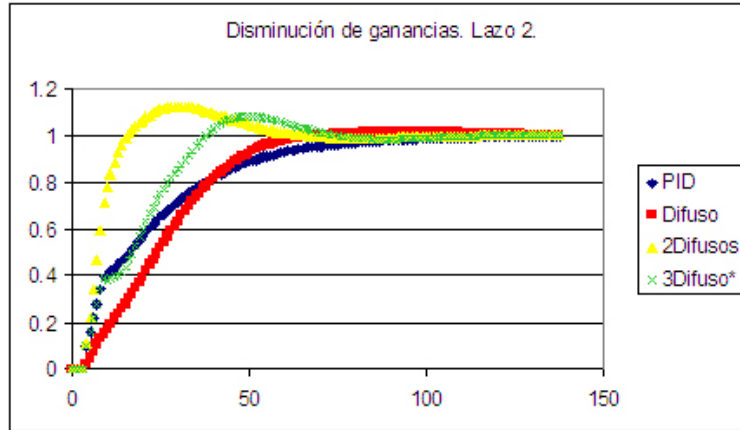


FIGURA B.6: Respuesta a la disminución de ganancias. Lazo 2

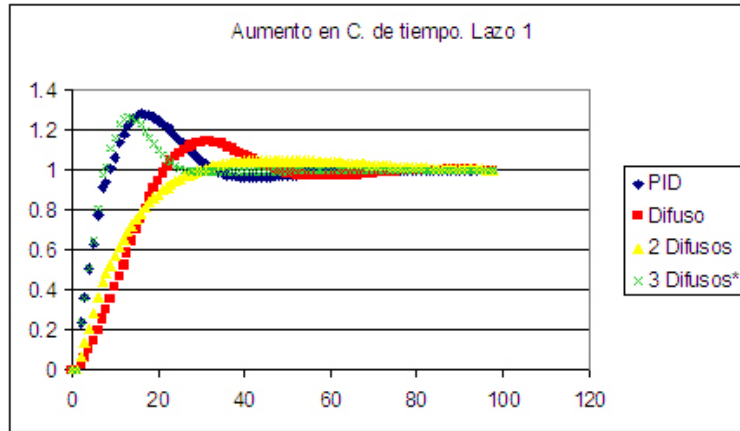


FIGURA B.7: Respuesta al aumento en las constantes de tiempo. Lazo 1

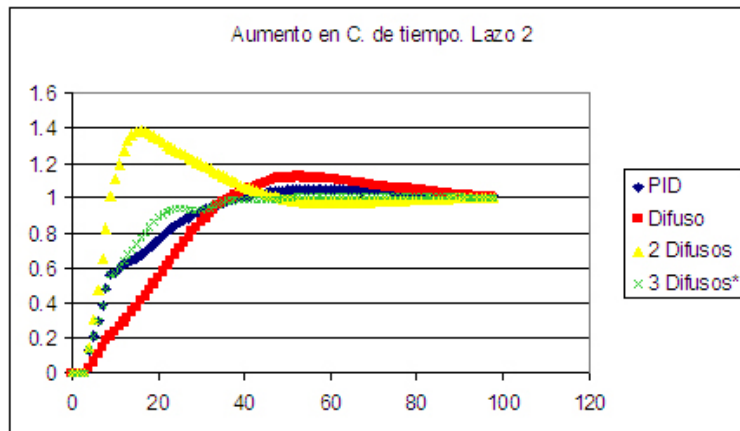


FIGURA B.8: Respuesta al aumento en las constantes de tiempo. Lazo 2

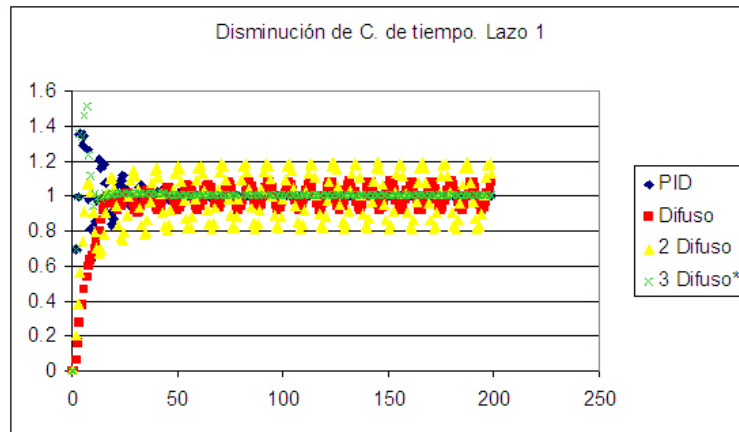


FIGURA B.9: Respuesta a la disminución en las constantes de tiempo. Lazo 1

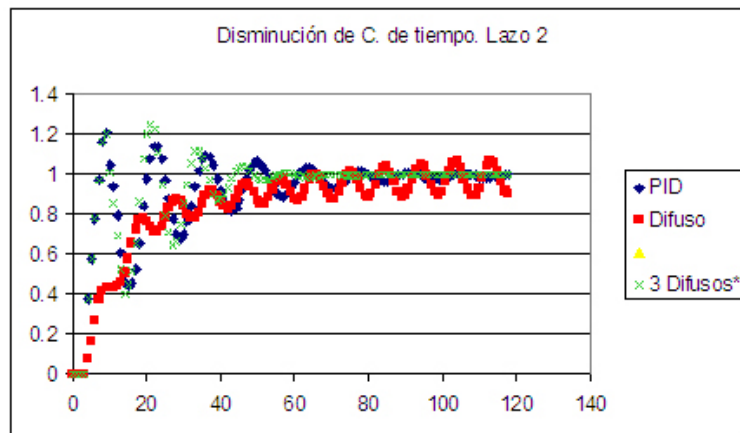


FIGURA B.10: Respuesta a la disminución en las constantes de tiempo. Lazo 2

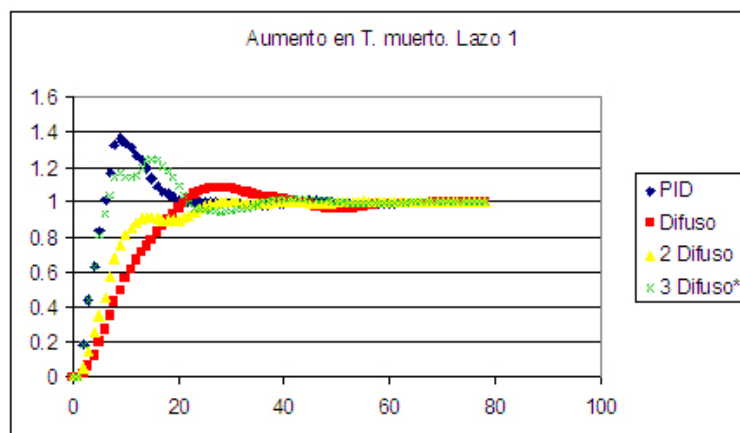


FIGURA B.11: Respuesta al aumento en los tiempos muertos. Lazo 1



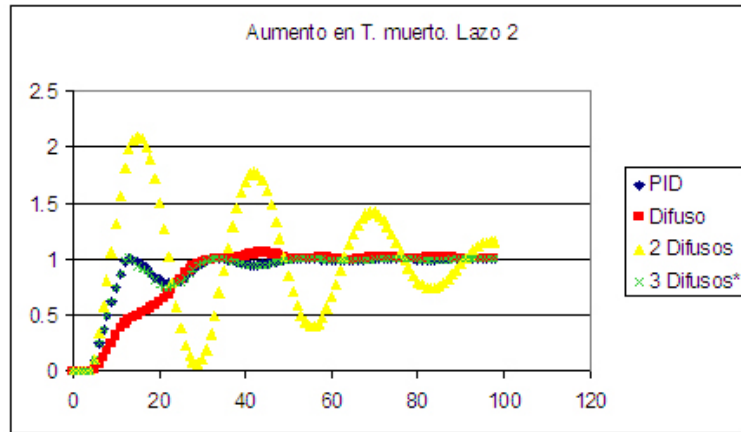


FIGURA B.12: Respuesta al aumento en los tiempos muertos. Lazo 2

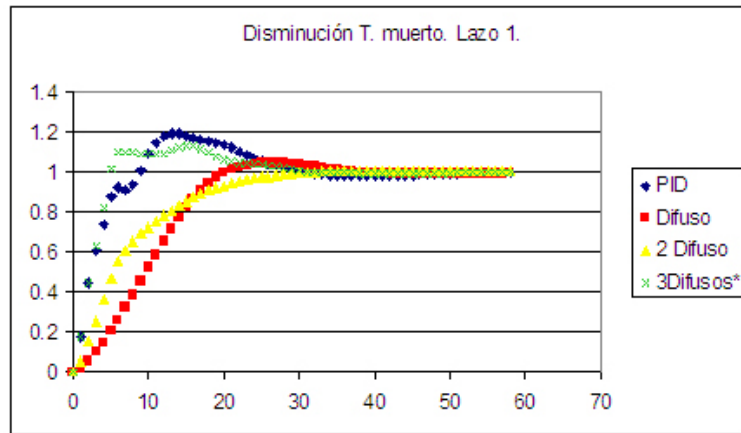


FIGURA B.13: Respuesta a la disminución en los tiempos muertos. Lazo 1

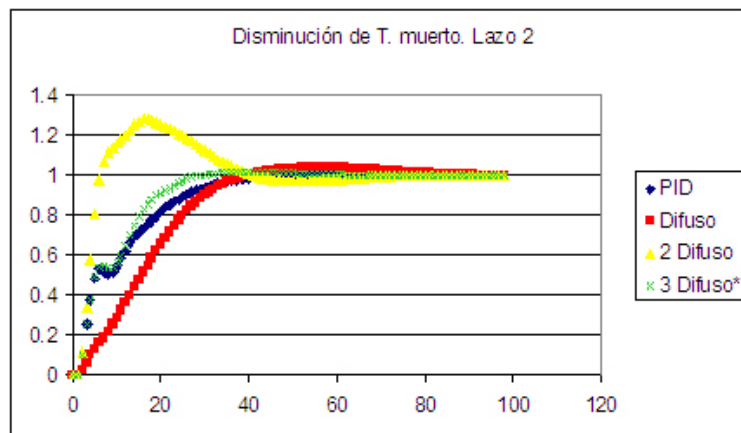


FIGURA B.14: Respuesta a la disminución en los tiempos muertos. Lazo 2

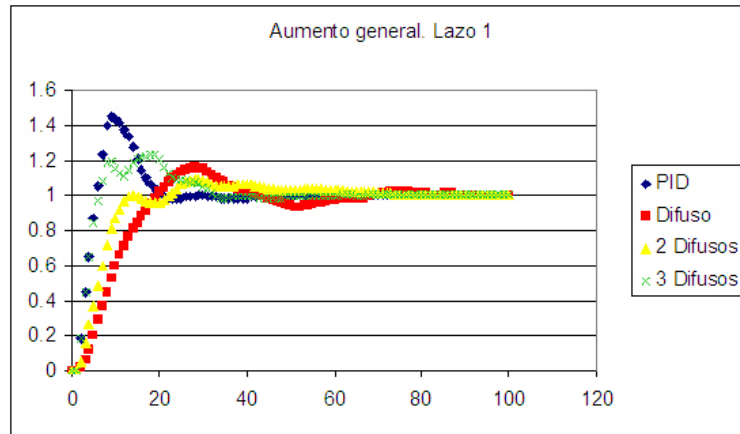


FIGURA B.15: Respuesta al aumento en todos los parámetros. Lazo 1

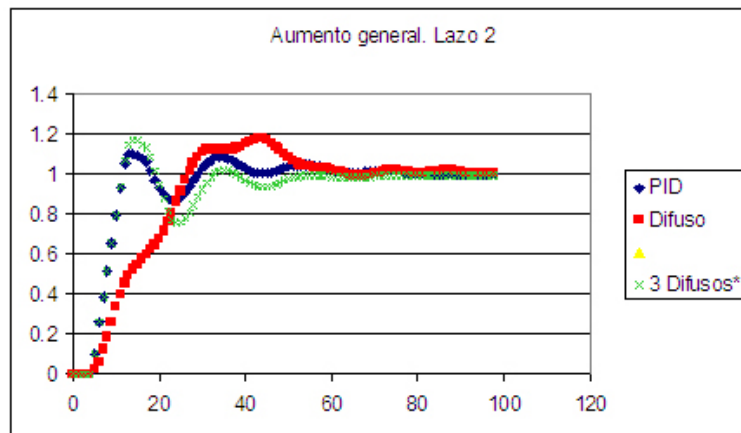


FIGURA B.16: Respuesta al aumento en todos los parámetros. Lazo 2

## Apéndice C

# Interfaz Humano-Máquina y código programado para control multivariable

### C.1. Introducción

Con el fin de generar las simulaciones necesarias para investigar y probar los métodos descritos se programó una interfase con la ayuda de LabWindows CVI. Ésta es una herramienta de la compañía National Instruments que permite al programador relacionar de manera sencilla interfases que trabajan en base a eventos con código programado en C. Como ha sido mencionado en ocasiones anteriores, se optó por este enfoque para las simulaciones con el fin de obtener un algoritmo que fuera de la mayor velocidad posible. Entre otras ventajas dignas de ser mencionadas se tienen la posibilidad de ofrecer una interfaz gráfica con el usuario, flexibilidad de funcionamiento, acceso al código y la posibilidad de desarrollar un programa ejecutable en cualquier computadora personal.

Tanto el código como las pruebas realizadas fueron desarrolladas en una computadora modelo HP Pavilion ze4900, procesador Celeron de 1.3 GHz, 224MB RAM y 30 GB de disco duro. El objetivo de esta sección es el de mostrar brevemente la interfaz desarrollada así como la parte significativa del código programado.

### C.2. Interfaz gráfica con el usuario

La interfaz gráfica fue desarrollada con el fin de poder simular, monitorear y analizar el comportamiento de procesos multivariables de dos entradas y dos salidas ante la aplicación de los algoritmos de control previamente mencionados. Se desarrollaron diferentes pantallas donde se le permite al usuario de forma simple introducir los datos necesarios para las simulaciones, cargar los datos de los controladores, correr los algoritmos de sintonización, guardar y graficar los datos, así como su análisis en base a la función de costo definida para el problema de optimización. Las diferentes pantallas serán descritas a continuación.

### C.2.1. Pantalla principal

Esta es la pantalla, mostrada en la figura C.1 donde se realiza la simulación, así como se tiene acceso al resto de los paneles del programa. En el primer graficador se aprecian los valores de las variables de proceso, en el segundo se aprecian las manipulaciones y en último, las señales de error. En este panel se pueden modificar los valores de referencia en modo automático y de manipulación en modo manual. Tiene la opción para guardar los datos en un archivo de texto. La función de los elementos de la pantalla se enumeran a continuación:

- Indicadores de variable de proceso: Marcados como %Nivel y %Temperatura, muestran el valor de las variables de salida de la planta multivariable.
- Indicador de Tiempo de muestreo: Marcado en la interfase como T, modifica el tiempo de muestreo que se utiliza en la simulación.
- Indicadores de referencia: Marcado como Ref Nivel y Ref Temp. En el modo automático modifica el valor de la referencia a seguir por el controlador.
- Indicadores de error del proceso: Denotados como Error Nivel y Error Temp. Muestran el valor del error de las variables de salida.
- Selector de algoritmo de control: Denotado como ALG. Permite seleccionar de alguno de los algoritmos de control disponibles.
- Botón de Evaluación de respuesta: Denotado como Comparar. Invoca a la pantalla donde se puede evaluar la calidad de las respuestas del proceso.
- Botón de panel de proceso: Nombrado como Proceso en la interfase. Invoca el panel donde es posible modificar los valores del proceso a simular.
- Botón de panel de algoritmo: Nombrado como Alg en la interfase. Invoca el panel del controlador elegido con el selector de algoritmo de control.
- Botón de guardado de datos: Denotado como Salvar en el panel. Invoca la rutina para guardar el archivo de las simulaciones realizadas con la interfase.
- Botón de salida de programa: Nombrado Salir en la pantalla. Termina la interfaz de simulación.
- Interruptor de almacenamiento de datos: Marcado como Log. En cuanto es activado, los datos de la simulación en curso son guardados en un buffer para después ser almacenado al presionar el botón de guardado de datos.

- Interruptor de modo de control: Selecciona entre el modo manual y automático del controlador.
- Interruptor de ejecución del programa: Al estar activado el programa realiza todas sus funciones. De lo contrario, queda en pausa.
- Indicador de saturación de espacio de almacenamiento. Marcado como OverF en la interfase. Avisa al usuario que el buffer de almacenamiento de los datos de una simulación en particular ha sido excedido para detener su funcionamiento.

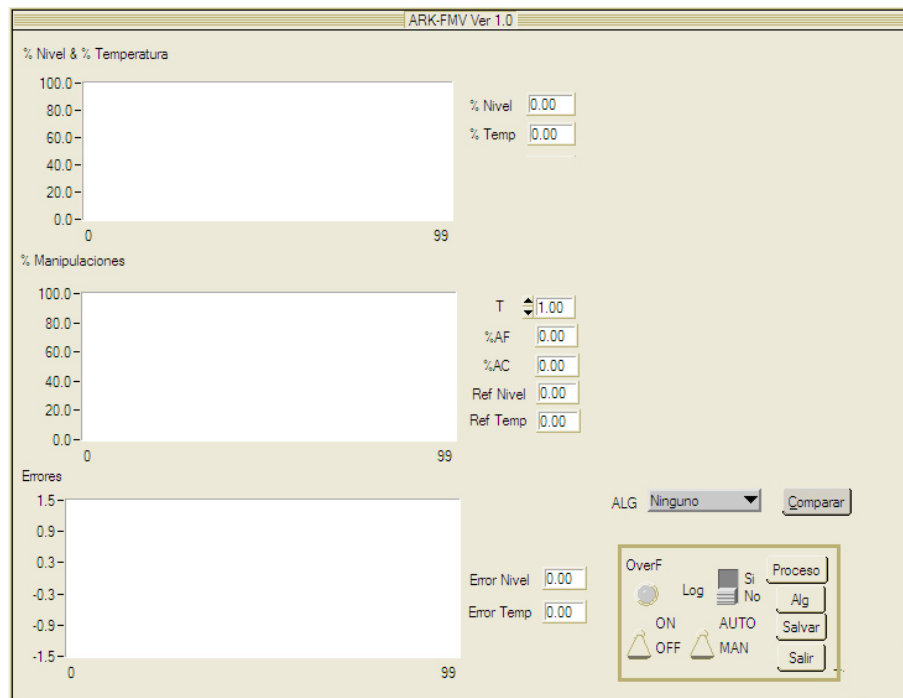


FIGURA C.1: Pantalla principal de la interfaz Hombre-Máquina

### C.2.2. Pantalla de proceso

En esta pantalla es posible, mostrada en la figura C.2, introducir los datos para configurar al proceso multivariable a simular con el programa, también es posible introducir condiciones iniciales de operación en el caso deseado. Cuenta a su vez con opciones para guardar estos datos en un archivo y poder abrirlos más tarde.

- Indicadores de las ganancias del proceso: Denotados como K en la interfase. Determinan la ganancia de cada una de las ganancias de los modelos que conforman al sistema de  $2 \times 2$ . Siguiendo el orden matricial, de arriba hacia abajo tenemos los datos para  $G_{11}$ ,  $G_{12}$ ,  $G_{21}$  y  $G_{22}$ , respectivamente.

- Indicadores de las constantes de tiempo del proceso: Denotados como Tau en la interfase y siguiendo el orden estipulado, determinan los valores respectivos de este parámetro en los modelos del proceso.
- Indicadores de tiempo muerto de proceso: Denotados como T0 en la interfase, determinan los tiempos muertos.
- Indicadores de Salida en estado estable: Marcados como YSS en la interfase, determinan el valor inicial de cada una de los procesos constitutivos. El valor de salida dependerá de la suma algebraica de estos valores.
- Indicadores de Entrada en estado estable: Marcados como USS en la interfase, determina la condición inicial de entrada de los procesos constitutivos.
- Selector de signo del proceso. Marcados como Negativo en la interfase, determinan si el valor de ganancia es positivo o negativo en la interacción del sistema.
- Botón de aceptación de datos: Con la etiqueta de Hecho en la interfaz, carga los datos introducidos en el programa.
- Botón de guardado de datos: Con la etiqueta de Guardar, almacena los datos del proceso en un archivo.
- Botón de cargado de datos: Con la etiqueta de Cargar, recupera los datos de un proceso guardado con anterioridad.
- Botón de cerrado de panel: Con la etiqueta de Salir, cierra el panel y vuelve a la pantalla principal.

### **C.2.3. Pantalla de evaluación de respuesta del sistema**

Aunque es posible realizar simulaciones y grabar sus datos para análisis posteriores, el programa cuenta con una opción que puede ser útil para este fin. En el panel descrito se cuenta con la opción de aplicar un escalón unitario a cada una de las entradas para observar el comportamiento de las salidas. Automáticamente ofrecerá la opción de guardar en un archivo la respuesta resultante a los experimentos. En los numéricos superiores muestra cada uno de los parámetros evaluados en la función de costo que se suma para obtener el valor total. Se tiene la opción de elegir el tiempo de simulación como múltiplos enteros de la mayor constante de tiempo del proceso a evaluar. La pantalla descrita se muestra en la figura C.3.

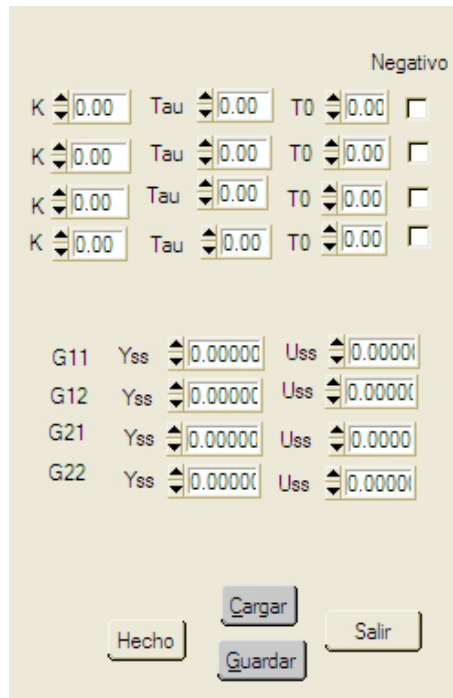


FIGURA C.2: Pantalla de introducción de datos del proceso multivariable.

- Indicadores de desempeño: Muestra cada uno de los valores evaluados en la función de costo. Se evaluarán los parámetros de suma de errores absoluta (IAE), suma de errores del acoplamiento (IAED), tiempo de establecimiento (T), sobretiro (Mp), error final (err), suavidad de manipulación (SU) y el total del costo. Esto está disponible para ambas entradas, marcado con el número correspondiente siguiendo al nombre del parámetro de evaluación.
- Indicador de tiempo de simulación: Etiquetado como TauSIM. Indica cuantas constantes de tiempo máximas serán evaluadas durante la simulación.
- Selector de algoritmo: Sin etiqueta. Elige el algoritmo, previamente configurado, para evaluar como sistema de control.
- Botón de inicio de simulación: Etiquetado como Probar. Arranca la simulación y el almacenamiento de los archivos resultantes.
- Cerrado de panel. Etiquetado como Salir. Cierra la pantalla, volviendo a la principal.

Comparaciones												
Desempeño												
IAE1	IAE2	IAED1	IAED2	T1	T2	Mp1	Mp2	enf1	enf2	SU1	SU2	Total
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Ninguno   
 TauSIM

FIGURA C.3: Pantalla de evaluación de función de costo.

#### C.2.4. Pantallas de configuración de controlador

En las pantallas restantes se tienen las opciones necesarias para ajustar el controlador correspondiente, guardar o cargar los datos de sintonizaciones previas y opciones para correr el algoritmo de ajuste explicado con anterioridad. Todas las pantallas son similares, cambiando solamente las interfaces destinadas a la introducción de parámetros específicos. Las pantallas aparecen en las figuras.

Los controles generales de las pantallas son los siguientes:

- Botón de ejecución de algoritmo de ajuste: Etiquetado como Sintonizar. Corre el algoritmo de ajuste en base a las condiciones actuales del controlador ajustado.
- Botón de cargado de datos: Etiquetado como Cargar. Lee del archivo los datos de una sintonía disponible y los coloca en las interfaces adecuadas en la pantalla de configuración.
- Botón de guardado de datos: Etiquetado como Guardar. Almacena los datos del controlador en particular en un archivo para uso posterior.
- Botón de confirmación de configuración: Etiquetado como Aceptar o Hecho. Carga la configuración actual a las variables correspondientes dentro del programa.
- Botón de cierre de pantalla: Etiquetado como Salir. Cierra el panel para volver a la pantalla principal de la interfaz.
- Indicador de temperatura inicial: Etiquetado como T0. Muestra al usuario el valor inicial de temperatura que el algoritmo de ajuste toma para seleccionar los estados obtenidos durante su ejecución.
- Indicador de temperatura actual: Etiquetado como TAnn o TAnnIn. Muestra al usuario el valor actual utilizado por el algoritmo.
- Indicador de temperatura final: Etiquetado como Tf. Muestra al usuario el valor final de temperatura que será evaluado.



- Probabilidad de aceptación: Etiquetado como P(E). Muestra el valor de probabilidad con la que se aceptará un nuevo estado propuesto.
- Indicador de Costo inicial: Etiquetado como J0. Muestra al usuario el valor de costo obtenido de la primer iteración del algoritmo.
- Indicador de Costo actual: Etiquetado como J1. Muestra al usuario el costo del estado actual que el algoritmo ha seleccionado como el óptimo global.
- Indicador de Costo propuesto: Etiquetado como J2. Muestra al usuario el costo del nuevo estado propuesto por el algoritmo de ajuste.
- Indicador de modo de optimización: Etiquetado como Optimizar con max(). Modifica la función de costo para que considere solo el máximo de los indicadores de cada una de las salidas. Útil solo en casos que el usuario lo considere necesario. Ofrece menores temperaturas iniciales y mejores costos a evaluar.
- Indicadores de tipo de acción de control: Etiquetados como Inversa 1 e Inversa 2. Disponibles en los casos de los controladores univariados solamente. Determinan la acción de control a seguir en función al signo de la ganancia del proceso a controlar.

Para el panel del controlador difuso multivariable, los controles particulares de esta interfaz son los siguientes:

- Indicador de número de entradas de controlador: Etiquetado como N.E. Máximo cuatro salidas posibles.
- Indicador de Número de conjuntos difusos de entrada: Etiquetado como C.D.E. Permite introducir como máximo 5 conjuntos difusos. Se entiende que los de las orillas serán rampas S y Z y los del centro, triángulos.
- Indicador de número de salidas: Marcado N.S. Ofrece un máximo de 2 salidas.
- Indicador de número de conjuntos difusos de salida: Marcado C.D.S. Ofrece la opción de introducir hasta 9 conjuntos. Estos son del tipo Singleton.
- Tabla de parámetros de conjuntos difusos de entrada: Marcado como ENTRADAS. Introduce los valores de  $\alpha$ ,  $\beta$  y  $\gamma$  que describe a los conjuntos difusos de entrada. Poner 0 si alguno de ellos no viene al caso.
- Tabla de parámetros de conjuntos difusos de salida: Marcado SALIDAS. Permite introducir las posiciones de los singleton de cada salida.

- Tabla de ampliaciones de entrada y salida: Marcado como AMPS. Permite introducir los valores de las ampliaciones. El primer valor corresponde a la salida. El resto corresponde al valor al valor de ampliación de cada entrada en particular.
- Tabla de base de reglas: Marcado como REGLAS. Ofrece el espacio necesario para introducir las reglas de cada implicacion. Se debe colocar el valor de la etiqueta que aparece en la tabla de salidas.

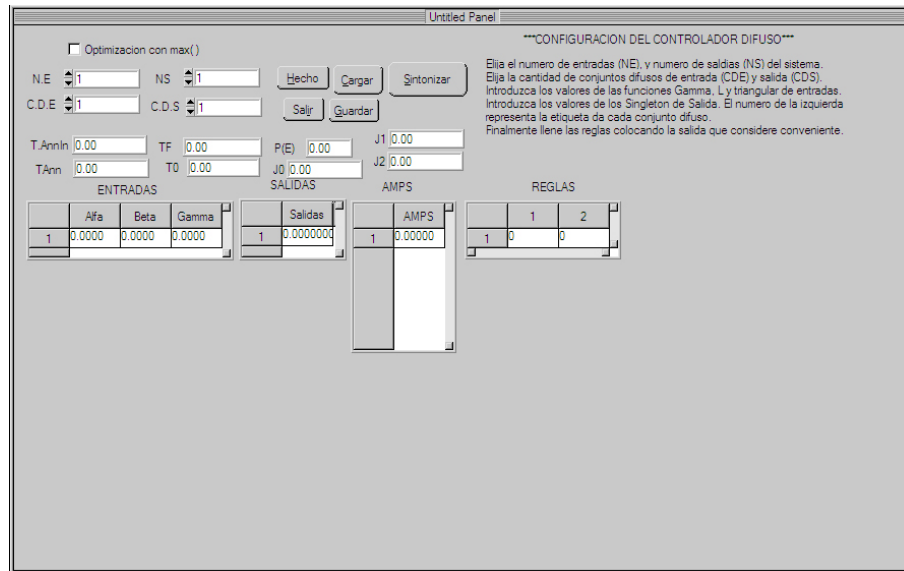


FIGURA C.4: Pantalla de configuración de controlador difuso multivariable.

Para el panel de la estrategia de 2 difusos con desacoplador, los controles particulares de esta interfaz son los siguientes:

- Tablas de conjuntos difusos de entrada: Marcadas como Difuso 1 y Difuso 2. Se ofrece espacio para introducir los valores  $\alpha$ ,  $\beta$  y  $\gamma$  de los conjuntos difusos. Se entiende que son tres conjuntos difusos.
- Tabla de reglas: Marcada como Reglas. Indica solamente el orden de las reglas consideradas, sin modificación posible. 9 reglas en total.
- Tabla de predicado de reglas: Marcada como Predicados. Se colocan las etiquetas de las salidas para determinar el predicado correspondiente a cada implicación.
- Tabla de salidas de controlador difuso: Marcada como Salidas. Ofrece espacio para determinar las posiciones de los singleton de cada salida. 9 como máximo.

- Tabla de ganancias: Marcada como Ganancias. Ofrece las ganancias de ampliación. Cada casilla, de izquierda a derecha corresponde a la entrada 1, salida 1, entrada 2 y salida 2, respectivamente.

The screenshot shows a configuration window for a fuzzy control strategy. It includes the following data tables:

	1	2	3
4	0.00000	0.00000	0.00000
5	0.00000	0.00000	0.00000
6	0.00000	0.00000	0.00000

	1	2	3
4	0.00000	0.00000	0.00000
5	0.00000	0.00000	0.00000
6	0.00000	0.00000	0.00000

	1	2
1	-1	-1
2	-1	0
3	-1	1
4	0	-1
5	0	0
6	0	1
7	1	-1
8	1	0
9	1	1

	1	2
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0

	1	2
1	0.00000	0.00000
2	0.00000	0.00000
3	0.00000	0.00000
4	0.00000	0.00000
5	0.00000	0.00000
6	0.00000	0.00000
7	0.00000	0.00000
8	0.00000	0.00000
9	0.00000	0.00000

	1	2	3	4
1	0.0000	0.0000	0.0000	0.0000

FIGURA C.5: Pantalla de configuración de estrategia de controlador difuso univariable con desacoplador.

Para el panel de la estrategia de 3 difusos con desacoplador, los controles particulares de esta interfaz son los siguientes:

- Tablas de configuración de conjuntos difusos de entrada. Marcadas como Entrada. Introducen datos de estos conjuntos como ha sido descrito anteriormente. Las tablas de la izquierda corresponden al controlador del lazo 1. Se le asignó a la acción proporcional el número 1 y al integral, el número 2. Ofrece un máximo de 9 conjuntos.
- Tablas de configuración de conjuntos difusos de salida. marcadas como Salida 1 y Salida 2. Determinan las posiciones de cada singleton de salida. En cada tabla se concentran las salidas de las tres acciones de cada lazo. 9 conjuntos como máximo por salida.
- Indicador de Número de conjuntos utilizado: Sin etiqueta. Determina cuantos conjuntos de entrada y salida serán considerados. Máximo 9 de ellos.
- Tabla de ganancias: Etiquetado como Ganancias. Muestra los valores de ganancia proporcional, integral y derivativo de cada controlador, en el orden señalado.

### C.3. Código del programa

En esta sección se destina espacio para agregar código del programa desarrollado. Debe de entenderse que no se enlistará todo el código generado dado que eso, por un

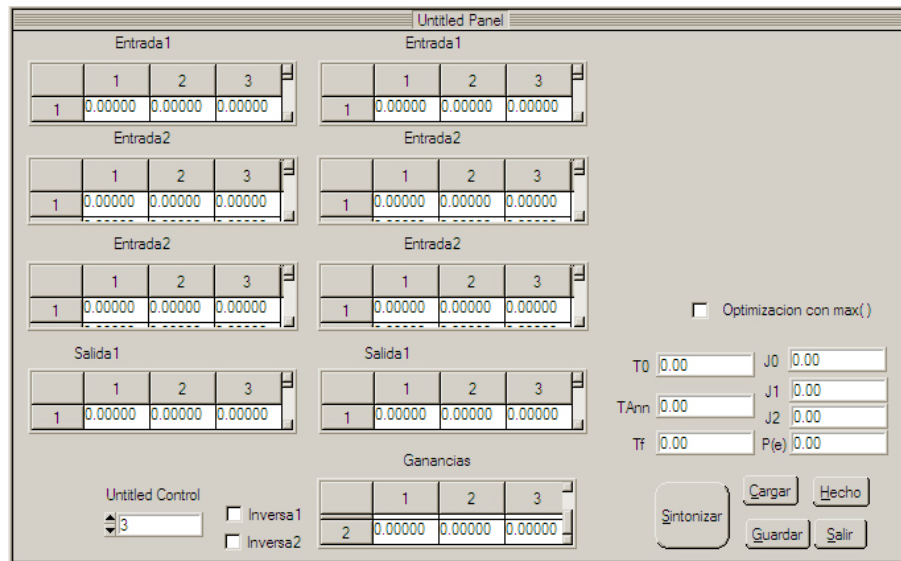


FIGURA C.6: Pantalla de configuración de estrategia de controlador PID de 3 difusos con desacoplador.

lado, no se piensa como útil para el lector y consume espacio de forma innecesaria. El objetivo de esta sección es mostrar secciones significativas del código que puedan servir de guía para alguien que desee programar algoritmos similares. Se presenta el código necesario para correr la técnica de ajuste y una rutina de Matlab que se diseñó con el fin de poder obtener las reglas del controlador difuso multivariable.

### C.3.1. Código del lazo de control

```
int CVICALLBACK SETPIDSINTONIZAR (int panel, int control, int event,
void *callbackData, int eventData1, int eventData2)
{
double PIDmenos[6];
double Jmin;
double J0,; //Para ajuste consantes de F costo.
double CJs[13];
double TAnn,Tin;//Temperatura RS.
double T0=1000000,TF=0;
double J1,J2;// Valor de la funcion de optimizacion, el valor del
double N0;
double epsAnn;// Factor de cambio amio del nuevo estado.
double TAnnIn;
int Nstate;// Cantidad de estados evaluados RS.
int tiempoJ;//Duracion de la prueba para obtener J.
int Nout;
```

```

    int conti, contj,contk, contTS;
double SPONV, SPOTP;
int OPTMAX;
switch (event)
{
case EVENT_COMMIT:
    autotune=1;
    SuspendTimerCallbacks(); /// Se detiene la accion de control.
    //1.Captura de datos de la funcion.
    TAnn=T0;
    NO=36; //Grados de libertad del sistema.
    Nout=20;
    contj=0;
    alfaTAnn=0.9;

    epsilon=.1; //Minima temperatura
    epsAnn=.1; //salto aleatorio.
    GetCtrlVal(screen1,PANEL_MUESTREO,&T);
    // Elige el tiempo de simulacion.
    (conti=0;conti<=3;conti++)
    {if(planta[2+conti*4]>tiempoJ){tiempoJ=planta[2+conti*4];}}
    tiempoJ=tiempoJ*3/T;

    //1.Estimacion de parametros de la funcion de optimizacion.
    ClearArraysAAM();

    for(conti=0;conti<13;conti++){Js[conti]=0;CJs[conti]=1;}
    for(conti=0;conti<tiempoJ;conti++)
    {
        sptp[0]=SPOTP; //Escalon
        spnv[0]=SPONV+1;
        AaM(0,1,1);
        Js[1]=Js[1]+CJs[1]*absu(errnv[0]);
        Js[3]=Js[3]+CJs[3]*absu(errtp[0]);
        Js[11]=Js[11]+CJs[11]*absu(opaf[0]-opaf[2]);
        //if(Js[11]<absu(opaf[0]-opaf[2])){Js[11]=absu(opaf[0]-opaf[2]);}
        if(errnv[0]<0){if(-1*errnv[0]>Js[7]){Js[7]=-1*errnv[0];}}
    }
    Js[5]=CJs[5]*CalculaTS(tiempoJ,errnv);
    Js[9]=CJs[9]*absu(errnv[0]);

```

```

    for(conti=0;conti<tiempoJ;conti++)
    {
        sptp[0]=SP0TP+1;
        spnv[0]=SP0NV;
        AaM(0,1,1);
        Js[2]=Js[2]+CJs[2]*absu(errtp[0]);
        Js[4]=Js[4]+CJs[4]*absu(errnv[0]);
        Js[12]=Js[12]+CJs[12]*absu(opac[0]-opac[2]);
        if(errtp[0]<0){if(-1*errtp[0]>Js[8])
            {Js[8]=-1*errtp[0];}}
        }
        Js[6]=CJs[6]*CalculaTS(tiempoJ,errtp);
        Js[10]=CJs[10]*absu(errtp[0]);
        Js[0]=0;
        for(conti=1;conti<13;conti++)
            {if(Js[conti]>Js[0]){Js[0]=Js[conti];}}
            ClearArraysAAM();
for(conti=0;conti<13;conti++){Js[conti]=0;}
J1=0;J2=0;
    for(conti=0;conti<tiempoJ;conti++)
    {
        sptp[0]=SP0TP;
        spnv[0]=SP0NV+1;
        AaM(0,1,1);

        Js[1]=Js[1]+CJs[1]*absu(errnv[0]);
        Js[3]=Js[3]+CJs[3]*absu(errtp[0]);
        Js[11]=Js[11]+CJs[11]*absu(opaf[0]-opaf[2]);
        if(errnv[0]<0){if(-1*errnv[0]>Js[7]){Js[7]=-1*errnv[0];}}
        }

Js[5]=CJs[5]*CalculaTS(tiempoJ,errnv);
Js[9]=CJs[9]*absu(errnv[0]);
for(conti=1;conti<13;conti++){J1=J1+Js[conti];}
for(conti=0;conti<13;conti++){Js[conti]=0;}

ClearArraysAAM();
    for(conti=0;conti<tiempoJ;conti++)
        {spnv[0]=SP0NV;
        sptp[0]=SP0TP+1;

```

```

AaM(0,1,1);

Js[2]=Js[2]+CJs[2]*absu(errtp[0]);
Js[4]=Js[4]+CJs[4]*absu(ernv[0]);
Js[12]=Js[12]+CJs[12]*absu(opac[0]-opac[2]);
if(errtp[0]<0){if(-1*errtp[0]>Js[8]){Js[8]=-1*errtp[0];}}
}
Js[6]=CJs[6]*CalculaTS(tiempoJ,errtp);
Js[10]=CJs[10]*absu(errtp[0]);

if(OPTMAX==1){JMax(Js);}

for(conti=1;cont<13;cont++){J1=J1+Js[cont];}

//Guarda el estado anterior.(entradas difusas)
for(conti=0;cont<6;cont++){PIDmenos[cont]=PIDs[cont];}
for (cont=0;cont<6;cont++)
{
PIDs[cont]=2*(PIDs[cont]+(rand()/32767.0))*(1-2*rand()/32767.0);

}
PIDs[1]=absu(PIDs[1]);
PIDs[2]=absu(PIDs[2]);
PIDs[4]=absu(PIDs[4]);
PIDs[5]=absu(PIDs[5]);
if(PIDs[1]<0.0001){PIDs[1]=0.0001;}
if(PIDs[4]<0.0001){PIDs[4]=0.0001;}
for(cont=0;cont<13;cont++){Js[cont]=0;}
ClearArraysAAM();
run=1;
autom=1;
Jmin=0;

for(cont=0;cont<tiempoJ;cont++)
{
sptp[0]=SPOTP;
spnv[0]=SPONV+1;
AaM(0,1,1);
Js[1]=Js[1]+CJs[1]*absu(ernv[0]);

```

```

Js[3]=Js[3]+CJs[3]*absu(errtp[0]);
Js[11]=Js[11]+CJs[11]*absu(opaf[0]-opaf[2]);
if(errnv[0]<0){if(-1*errnv[0]>Js[7]){Js[7]=-1*errnv[0];}}

}

Js[5]=CJs[5]*CalculaTS(tiempoJ, errnv);
Js[9]=CJs[9]*absu(errnv[0]);

for(conti=1;cont<13;cont++){Jmin=Jmin+Js[cont];}
for(conti=0;cont<13;cont++){Js[cont]=0;}

ClearArraysAAM();

for(conti=0;cont<tiempoJ;cont++)
{
spnv[0]=SPONV;
sptp[0]=SPOTP+1;
AaM(0,1,1);
Js[2]=Js[2]+CJs[2]*absu(errtp[0]);
Js[4]=Js[4]+CJs[4]*absu(errnv[0]);
Js[12]=Js[12]+CJs[12]*absu(opac[0]-opac[2]);
if(errtp[0]<0)
{if(-1*errtp[0]>Js[8]){Js[8]=-1*errtp[0];}}

}
Js[6]=CJs[6]*CalculaTS(tiempoJ, errtp);
Js[10]=CJs[10]*absu(errtp[0]);

if(OPTMAX==1){JMax(Js);}

for(conti=1;cont<13;cont++){Jmin=Jmin+Js[cont];}

for(contk=0;contk<6;contk++){PIDs[contk]=PIDmenos[contk];}

T0=absu(J1-Jmin)/0.2231436;
TAnn=T0;

```



```

    TF=pow(alfaTAnn,Nout)*T0;
    if(TF>25){TF=25;}
TAnnIn=TAnn;
TAnnPast=TAnn;
////-----

////3.Entramos en el ciclo del recocido.
while(TAnn-TF>epsilon)
{

//// Determinar cantidad a estados a evaluar
    Nstate=N0* (2+8*(1- (log(TAnn-TF)/log(T0-TF)) ));
//Nuevo estado (esto solo modifica los conjuntos difusos de entrada)
for(conti=0;conti<6;conti++){PIDmenos[conti]=PIDs[conti];}
for (conti=0;conti<6;conti++)
{
PIDs[conti]=2*(PIDs[conti]+(rand()/32767.0))*(1-2*rand()/32767.0);
}
PIDs[0]=absu(PIDs[0]);
PIDs[1]=absu(PIDs[1]);
PIDs[2]=absu(PIDs[2]);
PIDs[3]=absu(PIDs[3]);
PIDs[4]=absu(PIDs[4]);
PIDs[5]=absu(PIDs[5]);
//Se evalua de nuevo la funcion de perdidas.
for(conti=0;conti<13;conti++){Js[conti]=0;}
ClearArraysAAM();
run=1;
    autom=1;
    J2=0;
for(conti=0;conti<tiempoJ;conti++)
{
sptp[0]=SP0TP;
spnv[0]=SPONV+1;
AaM(0,1,1);
Js[1]=Js[1]+CJs[1]*absu(errnv[0]);
Js[3]=Js[3]+CJs[3]*absu(errtp[0]);
Js[11]=Js[11]+CJs[11]*absu(opaf[0]-opaf[2]);
if(errnv[0]<0){if(-1*errnv[0]>Js[7]){Js[7]=-1*errnv[0];}}

```

```

    }
    conti=contk;
Js[5]=CJs[5]*CalculaTS(tiempoJ, errnv);
    Js[9]=CJs[9]*absu(errnv[0]);

ClearArraysAAM();

for(conti=0;conti<tiempoJ;conti++)
{
spnv[0]=SPONV;
sptp[0]=SPOTP+1;
AaM(0,1,1);
Js[2]=Js[2]+CJs[2]*absu(errtp[0]);
    Js[4]=Js[4]+CJs[4]*absu(errnv[0]);
    Js[12]=Js[12]+CJs[12]*absu(opac[0]-opac[2]);
    if(errtp[0]<0){if(-1*errtp[0]>Js[8]){Js[8]=-1*errtp[0];}}
        conti=contk;
    Js[6]=CJs[6]*CalculaTS(tiempoJ, errtp);
Js[10]=CJs[10]*absu(errtp[0]);

    if(OPTMAX==1){JMax(Js);}

    for(conti=1;conti<13;conti++){J2=J2+Js[conti];}
    //Decidir si se acepta el estado
if(J2<=J1)
{
    J1=J2;
} //Si bajan las perdidas, aceptar el estado nuevo
if(J2>J1)
{
    if( rand()/32767.0 < exp((J1-J2)/TAnnIn) )
    {
        J1=J2;
    }
    //Si no, hacer un volado, y aceptar mediante Metropolis.
    else{for(contk=0;contk<6;contk++)
        {PIDs[contk]=PIDmenos[contk];}
        epsAnn=epsAnn+0.01;
        }//Si no volvemos al estado anterior.
}
}

```

```

    contj++;
    if(contj==Nstate)
    {contj=0;TAnn=alfaTAnn*TAnn;
    TAnnIn=TAnn;
    SetCtrlVal(screen4,PIDDECUP_TAnn,TAnn);
    //SetCtrlVal(screen3,PANELFUZZY_TANNOUT,TAnn);
}

}

ClearArraysAAM();
Beep();

break;
}

return 0;
}

```

### C.3.2. Código en Matlab para seleccionar reglas de controlador difuso multivariable

```

function rules=damereglas4(k)
%damereglas(k) genera las 81 reglas de un controlador difuso
% dada una planta de 2 x2.
%Su salida son las reglas listas para usarse.
%El formato de salida es;
%[ 4 columnas de regla (N=-1, Z=0, P=1), 2 columnas de y en ess,
% 2 del valor de la manipulacion, 2 de label de la manipulacion]
P=[-1 -1 -1 0 0 0 1 1 1 ; -1 0 1 -1 0 1 -1 0 1];
q=inv(k)*P
kk=k;
for i=1:length(q)
    for j=1:length(q)
        A=k*[q(1,i),q(2,j)]';
        B(2*i-1,j)=A(1);B(2*i,j)=A(2);
    end
end
end

```

```

label=ones(2,9);
for i=1:length(P)
for j=1:length(P)
    if q(1,i)>q(1,j)
        label(1,i)=label(1,i)+1;
    end
    if q(2,i)>q(2,j)
        label(2,i)=label(2,i)+1;
    end
end
end
label

B;
%extraigamos los pares plausibles de manipulacion.
%1.-partimos los pares

for i=1:9,
    for j= 1:9,
        upper(j,i)=B(2*j-1,i);
        lower(j,i)=B(2*j,i);
    end
end
upper;
lower;
manipup=0;
maniplow=0;
%Separamaos los plausibles Upper
for i=1:9
    switch i
    case {1,4,7}
        data=[upper(1,i) upper(4,i) upper(7,i)];
    case{2, 5, 8}
        data=[upper(2,i) upper(5,i) upper(8,i)];
    case{3,6,9}
        data=[upper(3,i) upper(6,i) upper(9,i)];
    end
    manipup=[manipup data];
end
%Separamaos los plausibles Lower

```

```

for i=1:9
    switch i
    case {1,4,7}
        data=[lower(1,i) lower(4,i) lower(7,i)];
    case{2, 5, 8}
        data=[lower(2,i) lower(5,i) lower(8,i)];
    case{3,6,9}
        data=[lower(3,i) lower(6,i) lower(9,i)];
    end
    maniplow=[maniplow,data];
end

manip=[manipup(2:length(manipup));maniplow(2:length(maniplow))]
clear manip
manip=P
manip
signos=sign(manip);
%Determinamos los signos
%Ordemanos los de igual combinacion de signos.
%Primero se agrupan por el termino superior.
negativos=[0;0];
positivos=[0;0];
ceros=[0;0];
for i=1:length(manip),
    if manip(1,i)<=-1,
        negativos=[negativos,manip(:,i)];
    end
    if manip(1,i)>=1,
        positivos=[positivos,manip(:,i)];
    end
    if manip(1,i)<1 & manip(1,i)>-1,
        ceros=[ceros,manip(:,i)];
    end
end

end
negativos;
ceros;
positivos;
cont3=-1;cont2=-1;cont1=-1;cont0=0;
reglas=[-1 -1 -1 -1];

```

```

for i=1:80,
    reglas=[reglas;[cont3,cont2,cont1,cont0]];
    cont0=cont0+1;
    if cont0==2,
        cont0=-1;
        cont1=cont1+1;
    end
    if cont1==2,
        cont1=-1;
        cont2=cont2+1;
    end
    if cont2==2,
        cont2=-1;
        cont3=cont3+1;
    end
    if cont3==2,
        cont3=-1;
    end
end
reglas;
manip= manip';
x1=0;x2=0;
%Determinan cada miembro de la regla para darle
% una magnitud arbitraria.
%Intentemos asignar finalmente las manipulaciones
    for i=1:81, %control de todos las reglas.
        x1=3*reglas(i,1)+reglas(i,2);
        %Mapea a un numero asignado de -4 a 4 cada
        % una de los enunciados de la regla.
        x2=3*reglas(i,3)+reglas(i,4);
    %Es correcto dado que necesitamos solo el valor de ambas.
        mag(i,:)=[x1,x2];
    end
mag2=mag;
mag2(:,2)=mag2(:,2)*sqrt(-1);
mag3=mag2(:,1) + mag2(:,2);
%Es necesario guardar las magnitudes para reconstruir
%la manipulacion tras tanto destrozo. :s
for i=1:length(mag3),
    %Voy a formar los vectores unitarios de la regla.

```

```

if abs(mag3(i))>0.00001
vecuniregla(i)=mag3(i)/abs(mag3(i));
else
    vecuniregla(i)=0;
end
vecphiregla(i,1)=phase(vecuniregla(i));
vecphiregla(i,2)=mag3(i);
end
vecphiregla;

%Vectores de manipulaciones.
for i=1:length(manip),
%formar los vectores unitarios de las manipulaciones.
    vecmanip(i)=manip(i,1)+manip(i,2)*sqrt(-1);
    if abs(vecmanip(i))>0.00001
    vecunimanip(i)=vecmanip(i)/abs(vecmanip(i));
    else
        vecunimanip(i)=0;
    end
    vecphimanip(i,1)=phase(vecunimanip(i));
    vecphimanip(i,2)=abs(vecmanip(i));
end
vecphimanip;
pi;

for i=1:length(vecphiregla)
%Primero obtengo un vector de los errores de una regla
%determinada con las manip (respecto a la fase)
for j=1:length(vecphimanip)
    error(j,1)=vecphiregla(i,1)-vecphimanip(j,1);
end
%ya que los tengo en el de errores,
% hallo las que son menores respecto a la fase.
apuntador(1,1)=1;
for j=1:length(error)
    for k=1:length(error)
        if abs(error(k,1))<abs(error(apuntador,1))
            apuntador=k;
        end
    end
end

```

```

        end
    end
    %En esta parte lo que se ha hecho es hallar un valor minimo
    % bajo el criterio de fase.
    %Ahora se vera si existen otros valores mas
    cont=2;
    for j=1:length(error)
        if abs(error(j,1))<=abs(error(apuntador(1,1),1))
            if j ~= apuntador
                [error(j,1),error(apuntador(1,1),1)];
                apuntador(cont,1)=j;
                cont=cont+1;
            end
        end
    end
    end
    %Aqui se tendran una serie de manipulaciones candidatas,
    %ahora hay que obtener el minimo de estos coeficientes.
    %por el criterio de magnitud.
    length(apuntador);
    for j=1:length(apuntador)
errorMa(j,1)=abs(vecuniregla(i))-abs(vecunimanip(apuntador(j)));
    end
    %formado el vector de errores de magnitud, hallo el minimo.
    apuntadorM=1;
    for j=1:length(errorMa)
        for k=1:length(errorMa)
            if abs(errorMa(k,1))<abs(errorMa(apuntadorM,1))
                apuntadorM=k;
            end
        end
    end
    ene
    val=apuntador(apuntadorM,1);
    val;
    manipop(i,:)=manip(val,:);
    clear apuntador apuntadorM val error errorMa cont    ;
end
manipop;
OP=inv(kk)*manipop';
OP=OP';
aux1=0;

```



```

aux2=0;
for i=1:length(reglas),
    for j=1:length(q),
        if abs(q(1,j)-OP(i,1))<0.00001
            aux1=label(1,j)-1;
        end
        if abs(q(2,j)-OP(i,2))<0.00001
            aux2=label(2,j)-1;
        end
        end
        OPs(i,:)=[aux1,aux2];
    end
qq=1:1:length(reglas);
qq=qq';
rules=[reglas,manipop,OP,OPs,qq];

```