

Sistema de control neuro-difuso basado en relaciones booleanas

Helbert Eduardo Espitia Cuchango

Universidad de Oviedo
Doctorado en Energía y Control de Procesos
Oviedo, España
2021



RESUMEN DEL CONTENIDO DE TESIS DOCTORAL

1.- Título de la Tesis	
Español: Sistema de control neuro-difuso basado en relaciones booleanas.	Inglés: Neuro-fuzzy control system based on boolean relations
2.- Autor	
Nombre: Helbert Eduardo Espitia Cuchango	Pasaporte:
Programa de Doctorado: Doctorado en Energía y Control de Procesos	
Órgano responsable: Centro Internacional de Postgrado	

RESUMEN (en español)

En sistemas de control, las redes neuronales son una buena alternativa en casos donde resulta difícil el diseño de un sistema de control por métodos convencionales debido a la complejidad del sistema, no linealidades y variación de sus parámetros, entre otros aspectos.

Una de las desventajas de las redes neuronales es su inicialización, convencionalmente de forma aleatoria, lo cual no garantiza tener la misma representación cada vez que se entrene. Otra dificultad consiste en establecer una arquitectura de red neuronal adecuada. Ante esto, una alternativa son los sistemas de lógica difusa y su integración con redes neuronales teniendo los sistemas neuro-difusos. En el área de la lógica difusa existen estudios sobre esquemas simplificados que permiten disminuir la complejidad del sistema tal como es el planteamiento de los sistemas de lógica difusa basados en relaciones booleanas. Con este tipo de sistema se puede establecer una estructura compacta y una fácil inicialización del sistema difuso.

Adicionalmente, los sistemas de control adaptativo son adecuados para el monitoreo del desempeño del sistema de control con parámetros desconocidos y variantes. Estos sistemas requieren de estructuras adecuadas que permitan efectuar la identificación de la planta y la sintonización del controlador, lo cual se puede lograr con los sistemas neuro-difusos basados en relaciones booleanas.

Este documento propone un sistema de control adaptativo empleando sistemas neuro-difusos basados en relaciones booleanas. En relación con los sistemas a controlar, se tiene un sistema de generación distribuida de energía ya que su integración con la red de distribución presenta desafíos, siendo uno de estos el control del flujo de potencia para regular el voltaje en los nodos del sistema de distribución. También se considera el control de sistemas hidráulicos SISO (Single Input Single Output); primero para el control de un tanque cónico y luego para dos tanques esféricos acoplados; finalmente, se realiza el control para un sistema MIMO (Multiple Input Multiple Output) acoplado. Estos sistemas se caracterizan por presentar diferentes no linealidades que dificultan su control.

El documento está organizado de la siguiente forma. En primer lugar, se presenta la justificación y formulación del problema. En el siguiente capítulo se describen los objetivos propuestos. Posteriormente, se muestran los antecedentes del proyecto realizando una revisión sobre propuestas y trabajos relacionados. En el capítulo siguiente se presenta el marco teórico. Luego se describe el desarrollo de la propuesta presentando el proceso de diseño de los sistemas difusos basados en relaciones booleanas. Con estos conceptos se establecen dos estructuras compactas utilizadas para el desarrollo de un sistema de control adaptativo; posteriormente se realiza la aplicación de control adaptativo en un sistema de distribución de energía, también se desarrollan dos aplicaciones para sistemas hidráulicos SISO y una para MIMO. Finalmente se presentan las conclusiones de los resultados obtenidos.



RESUMEN (en Inglés)

In control systems, neural networks are a suitable alternative in cases where it is difficult to design a control system using conventional methods, due to the complexity of the system, non-linearities, and variation of its parameters, among other aspects.

One of the disadvantages of neural networks is their conventionally random initialization, which does not guarantee to have the same representation every time it is trained. Another difficulty is establishing a suitable neural network architecture. An alternative to this consists of fuzzy logic systems and their integration with neural networks, having neuro-fuzzy systems. In the area of fuzzy logic, there are studies on simplified schemes which allow reducing the complexity of the system, such as the approach to fuzzy logic systems based on boolean relationships. With this type of system, a compact structure and easy initialization of the fuzzy system can be established.

Additionally, adaptive control systems are suitable for monitoring the performance of the control system with unknown parameters and variants. These systems require adequate structures that allow plant identification and controller tuning, which can be achieved with neuro-fuzzy systems based on boolean relationships.

In this document, an adaptive control system is proposed using neuro-fuzzy systems based on boolean relationships. Regarding the systems to be controlled, there is a distributed power generation system since its integration with the distribution network presents challenges, one of these being the control of the power flow to regulate the voltage at the nodes of the distribution system. It is also considered the control of hydraulic systems SISO (Single Input Single Output) first for the control of a conical tank and then for two coupled spherical tanks; finally, the control is carried out for a coupled MIMO (Multiple Input Multiple Output) system. These systems are characterized by presenting different non-linearities which make their control difficult.

The document is organized as follows. First, the justification and problem formulation are presented. The proposed objectives are described in the following chapter. Subsequently, the background of the project is shown, making a review on proposals and related works. The following chapter presents the theoretical framework. Then the development of the proposal is described where the design process of fuzzy systems based on boolean relationships is presented, with these concepts two compact structures are established which are used for the development of an adaptive control system; later, the adaptive control application is carried out in a power distribution system, two applications are also carried out for SISO hydraulic systems and one for MIMO. Finally, the conclusions of the results obtained are presented.

**SR. PRESIDENTE DE LA COMISIÓN ACADÉMICA DEL PROGRAMA DE DOCTORADO
EN ENERGÍA Y CONTROL DE PROCESOS**

Sistema de control neuro-difuso basado en relaciones booleanas

Helbert Eduardo Espitia Cuchango

Tesis presentada como requisito para optar al título de:
Doctor en Energía y Control de Procesos

Director:
PhD. Iván Machón González

Universidad de Oviedo
Doctorado en Energía y Control de Procesos
Oviedo, España
2021

Agradecimientos

En primer lugar quiero expresar gratitud a mis padres y familia cuyo apoyo incondicional, fue fundamental para poder continuar y finalizar este complejo trabajo.

Por su colaboración en la primera parte de este trabajo expreso un agradecimiento a los profesores Guzmán Díaz y Susana Díaz por su orientación en temas de distribución de energía y lógica difusa respectivamente.

También por toda su colaboración y apoyo agradezco al director de la tesis Iván Machón y también al profesor Hilario López, quienes fueron parte fundamental para lograr los objetivos de este trabajo, al proporcionar observaciones y orientaciones acertadas.

Estoy igualmente agradecido con el profesor Ignacio Díaz y a todos los miembros de las comisiones de seguimiento quienes proporcionaron observaciones objetivas sobre el avance de la tesis.

Por sus sugerencias y colaboración, a los profesores Lilian Bejarano, Joaquín Meza y José Soriano, como también a los revisores que intervinieron en las publicaciones relacionadas con la tesis.

Finalmente expreso mi cariño y gratitud por su apoyo y comprensión a Oscar Martínez, Cesar González, Luisa Pinzón y los amigos omitidos, que proporcionaron su aliento en los diferentes momentos del desarrollo de este trabajo.

Contenido

Resumen	1
Introducción	3
I Descripción del proyecto	5
1. Formulación del proyecto	6
1.1. Justificación	6
1.1.1. Sistemas neuro-difusos	7
1.1.2. Sistemas de control adaptativo	8
1.2. Formulación del problema	10
2. Objetivo general y objetivos específicos	11
2.1. Objetivo general	11
2.2. Objetivos Específicos	11
3. Antecedentes	12
3.1. Introducción	12
3.2. Sistemas de lógica difusa basados en relaciones booleanas	12
3.3. Aplicaciones de control neuro-difuso adaptativo	14
3.3.1. Aplicaciones varias	14
3.3.2. Control de motores, vehículos eléctricos y robótica	15

3.3.3.	Control de invernaderos	16
3.3.4.	Control de modo deslizante	16
3.3.5.	Control de sistemas no lineales	17
3.3.6.	Control de sistemas MIMO	18
3.4.	Sistemas de control MIMO mediante redes neuronales	19
3.4.1.	Enfoque de control predictivo	19
3.4.2.	Enfoque de control adaptativo	20
3.4.3.	Enfoque adaptativo para sistemas con incertidumbre	21
3.4.4.	Enfoque superficie de control	22
3.4.5.	Aplicación control PID	23
3.4.6.	Aplicación con lógica difusa	23
3.4.7.	Aplicación control de llenado de tanques	23
3.5.	Técnicas de control en sistemas de generación distribuida de energía	24
3.5.1.	Enfoques sobre el control en sistemas de generación distribuida	26
3.5.2.	Control PID	27
3.5.3.	Control robusto	27
3.5.4.	Control predictivo	28
3.5.5.	Control difuso	28

II Marco teórico 31

4. Elementos de lógica difusa 32

4.1.	Introducción	32
4.2.	Álgebras De Morgan, Kleene y Boole	32
4.2.1.	Álgebra de Kleene y cálculo proposicional difuso	33
4.3.	Conjuntos concretos y conjuntos difusos	34
4.4.	Conjunto difuso	35
4.5.	Funciones de pertenencia	36
4.6.	Propiedades de los conjuntos difusos	38

4.7.	Operaciones sobre conjuntos difusos	41
4.8.	Relaciones difusas	43
4.8.1.	Implicaciones difusas	44
5.	Sistemas de lógica difusa	46
5.1.	Introducción	46
5.2.	Modelos para la implementación de sistemas de lógica difusa	46
5.3.	Elementos de un sistema de inferencia difusa	47
5.4.	Sistemas tipo Mamdani	48
5.4.1.	Base de reglas difusas	48
5.4.2.	Defusificación	49
5.5.	Sistemas tipo Takagi-Sugeno	50
5.5.1.	Base de reglas difusas	50
5.5.2.	Cálculo de la salida	50
5.5.3.	Sistemas difusos Takagi-Sugeno no normalizados	52
6.	Sistemas de control mediante redes neuronales	53
6.1.	Introducción	53
6.2.	Redes neuronales	53
6.2.1.	Modelo biológico de una neurona	54
6.2.2.	Perceptrón	54
6.2.3.	Redes neuronales	55
6.2.4.	Entrenamiento y ejecución	55
6.3.	Sistemas de control mediante redes neuronales	56
6.4.	Control predictivo	57
6.5.	Control NARMA-L2	58
6.6.	Controlador neuronal con modelo de referencia	59
7.	Identificación supervisada mediante sistemas neuronales	61
7.1.	Introducción	61

7.2.	Identificación de sistemas	61
7.3.	Identificación mediante redes neuronales	63
7.4.	Identificación empleando sistemas de lógica difusa	63
7.5.	Identificación mediante sistemas neuro-difusos	64
7.6.	Proceso general en un algoritmo de entrenamiento	65
 III Desarrollo		66
 8. Sistemas de inferencia difusa basados en relaciones booleanas		67
8.1.	Introducción	67
8.2.	Sistemas de inferencia difusa basados en relaciones booleanas	68
8.2.1.	Elementos del sistema de inferencia difusa	69
8.2.2.	Salida virtual	69
8.3.	Proceso de inferencia difusa en un FIS-BBR	71
8.4.	Metodología de diseño	73
8.4.1.	Condición de regularidad	75
8.4.2.	Extensión de una tabla de Boole a una de Kleene	76
8.4.3.	Consideraciones para la selección de conjuntos difusos	76
8.5.	Aplicación demostrativa para el control de nivel en un tanque	78
8.5.1.	Modelo del sistema	78
8.5.2.	Diseño booleano	79
8.5.3.	Diseño FIS-BBR	80
8.5.4.	Conjuntos booleanos	80
8.5.5.	Implementación FIS-BBR	83
8.5.6.	Resultados	84
8.6.	Simplificación de términos en forma normal disyuntiva	87
8.6.1.	Forma normal disyuntiva obtenida a partir de tablas de Kleene	87
8.7.	Sistemas difusos compactos basados en relaciones booleanas	89
8.7.1.	Generación de términos en forma normal disyuntiva	90

8.7.2.	Expresiones con álgebra de Boole para FIS-BBR compactos	91
8.8.	Extensión a la tabla de Kleene	91
8.8.1.	Función de activación con dependencia unitaria	92
8.8.2.	Función de activación con dependencia parcial	95
8.8.3.	Función de activación con dependencia total	97
8.9.	Sistemas difusos compactos basados en relaciones booleanas	99
8.9.1.	Sistema compacto con arquitectura I	100
8.9.2.	Sistema compacto con arquitectura II	101
9.	Sistema de control neuro-difuso adaptativo	104
9.1.	Introducción	104
9.2.	Características del sistema de control adaptativo neuro-difuso	104
9.3.	Arquitectura del sistema de control neuro-difuso adaptativo	105
9.3.1.	Descripción del proceso de control adaptativo	107
9.4.	Arquitectura general para la identificación de la planta	108
9.5.	Arquitectura del controlador	109
9.6.	Aplicaciones para identificación y control	111
9.6.1.	Arquitectura I	111
9.6.2.	Arquitectura II	113
9.7.	Configuraciones de los sistemas neuro-difusos	117
9.7.1.	Arquitectura II con un conjunto difuso sigmoideal	118
9.7.2.	Arquitectura II con dos conjuntos difusos sigmoideales	118
9.7.3.	Arquitectura II con dos conjuntos sigmoideales y uno gaussiano	118
9.7.4.	Arquitectura II con cuatro conjuntos difusos sigmoideales	119
9.7.5.	Arquitectura II con dos conjuntos difusos gaussianos	119
9.7.6.	Arquitectura I con cuatro conjuntos difusos gaussianos	120
9.7.7.	Arquitectura I con cuatro conjuntos difusos sigmoideales	120
9.8.	Proceso general para la optimización de sistemas difusos	120
10.	Control del flujo de potencia en un sistema de generación	122

10.1. Introducción	122
10.2. Cálculo del flujo de carga	123
10.3. Operador de la red de distribución	123
10.4. Métodos de regulación	124
10.4.1. Regulador automático de voltaje	124
10.4.2. Control de conmutadores en carga	125
10.5. Trabajos realizados para el control de flujo de potencia	126
10.6. Control adaptativo para un sistema de distribución de energía	127
10.6.1. Proceso de control adaptativo	127
10.7. Sistema de distribución de energía	131
10.7.1. Resultados experimentales	132
10.7.2. Metodología para el análisis estadístico de resultados	133
10.7.3. Métrica utilizada	135
10.7.4. Configuración de experimentos	135
10.7.5. Resultados para un generador	137
10.7.6. Análisis estadístico de resultados para un generador	139
10.7.7. Resultados para tres generadores	143
10.7.8. Análisis estadístico de resultados para tres generadores	145
11. Control de sistemas hidráulicos SISO	149
11.1. Introducción	149
11.2. Aplicación del adaptativo para regular el llenado de un tanque cónico	149
11.2.1. Esquema utilizado para la identificación de la planta	149
11.2.2. Esquema de tiempo discreto utilizado para el controlador	150
11.3. Arquitectura para la identificación de la planta	150
11.4. Ecuaciones de entrenamiento para la identificación de la planta	152
11.5. Arquitectura del controlador	153
11.6. Ecuaciones para el entrenamiento de parámetros del controlador	155
11.6.1. Derivadas de los parámetros del controlador con respecto a la planta	156

11.6.2. Derivadas de los parámetros del controlador respecto al controlador	157
11.6.3. Proceso para el entrenamiento del controlador	159
11.7. Modelo del tanque cónico	161
11.8. Resultados	162
11.9. Aplicación del adaptativo para regular el llenado de tanques esféricos	165
11.9.1. Arquitectura del sistema neuro-difuso para identificación y control	166
11.10 Ecuaciones de entrenamiento para la identificación de la planta	168
11.10.1 Ecuaciones de entrenamiento para el controlador	170
11.10.2 Derivadas de los parámetros del controlador respecto a la planta	171
11.10.3 Derivadas de los parámetros el controlador respecto al controlador	172
11.10.4 Proceso para el entrenamiento del controlador	174
11.11 Modelo del sistema hidráulico	175
11.12 Configuración de experimentos	178
11.12.1 Método empleado para el análisis estadístico de resultados	179
11.13 Resultados experimentales	179
11.13.1 Resultados del comportamiento del sistema	181
12. Control de un sistema hidráulico MIMO acoplado	184
12.1. Introducción	184
12.2. Sistema hidráulico MIMO	184
12.3. Sistema de control neuro-difuso adaptativo MIMO	188
12.4. Sistema difuso compacto basado en relaciones booleanas	189
12.5. Configuraciones MIMO con sistemas basados en relaciones booleanas	191
12.5.1. Configuraciones MIMO utilizadas	194
12.6. Ecuaciones para el entrenamiento del controlador	194
12.7. Ecuaciones de entrenamiento para la identificación de la planta	201
12.7.1. Ecuaciones de entrenamiento para el módulo de salida	202
12.7.2. Ecuaciones de entrenamiento para el módulo de entrada	204
12.8. Resultados experimentales	207

13. Conclusiones, aportes originales y trabajos futuros	214
13.1. Conclusiones	214
13.2. Aportes originales	215
13.3. Trabajos futuros	216
13.4. Productos académicos	216
Bibliografía	220

Resumen

En sistemas de control, las redes neuronales son una buena alternativa en casos donde resulta difícil el diseño de un sistema de control por métodos convencionales debido a la complejidad del sistema, no linealidades y variación de sus parámetros, entre otros aspectos.

Una de las desventajas de las redes neuronales es su inicialización, convencionalmente de forma aleatoria, lo cual no garantiza tener la misma representación cada vez que se entrene. Otra dificultad consiste en establecer una arquitectura de red neuronal adecuada. Ante esto, una alternativa son los sistemas de lógica difusa y su integración con redes neuronales teniendo los sistemas neuro-difusos. En el área de la lógica difusa existen estudios sobre esquemas simplificados que permiten disminuir la complejidad del sistema tal como es el planteamiento de los sistemas de lógica difusa basados en relaciones booleanas. Con este tipo de sistema se puede establecer una estructura compacta y una fácil inicialización del sistema difuso.

Adicionalmente, los sistemas de control adaptativo son adecuados para el monitoreo del desempeño del sistema de control con parámetros desconocidos y variantes. Estos sistemas requieren de estructuras adecuadas que permitan efectuar la identificación de la planta y la sintonización del controlador, lo cual se puede lograr con los sistemas neuro-difusos basados en relaciones booleanas.

Este documento propone un sistema de control adaptativo empleando sistemas neuro-difusos basados en relaciones booleanas. En relación con los sistemas a controlar, se tiene un sistema de generación distribuida de energía ya que su integración con la red de distribución presenta desafíos, siendo uno de estos el control del flujo de potencia para regular el voltaje en los nodos del sistema de distribución. También se considera el control de sistemas hidráulicos SISO (*Single Input Single Output*); primero para el control de un tanque cónico y luego para dos tanques esféricos acoplados; finalmente, se realiza el control para un sistema MIMO (*Multiple Input Multiple Output*) acoplado. Estos sistemas se caracterizan por presentar diferentes no linealidades que dificultan su control.

El documento está organizado de la siguiente forma. En primer lugar, se presenta la justificación y formulación del problema. En el siguiente capítulo se describen los objetivos propuestos. Posteriormente, se muestran los antecedentes del proyecto realizando una revisión sobre propuestas y trabajos relacionados. En el capítulo siguiente se presenta el marco teórico. Luego se describe el desarrollo de la propuesta presentando el proceso de diseño de los sistemas difusos basados en relaciones booleanas. Con estos conceptos se establecen dos estructuras compactas utilizadas para el desarrollo de un sistema de control adaptativo; posteriormente se realiza la aplicación de control adaptativo en un sistema de distribución de energía, también se desarrollan dos aplicaciones para sistemas hidráulicos SISO y una para MIMO. Finalmente se presentan las conclusiones de los resultados obtenidos.

Palabras clave: Adaptativo, control, lógica difusa, neuro-difuso.

Abstract

In control systems, neural networks are a suitable alternative in cases where it is difficult to design a control system using conventional methods, due to the complexity of the system, non-linearities, and variation of its parameters, among other aspects.

One of the disadvantages of neural networks is their conventionally random initialization, which does not guarantee to have the same representation every time it is trained. Another difficulty is establishing a suitable neural network architecture. An alternative to this consists of fuzzy logic systems and their integration with neural networks, having neuro-fuzzy systems. In the area of fuzzy logic, there are studies on simplified schemes which allow reducing the complexity of the system, such as the approach to fuzzy logic systems based on boolean relationships. With this type of system, a compact structure and easy initialization of the fuzzy system can be established.

Additionally, adaptive control systems are suitable for monitoring the performance of the control system with unknown parameters and variants. These systems require adequate structures that allow plant identification and controller tuning, which can be achieved with neuro-fuzzy systems based on boolean relationships.

In this document, an adaptive control system is proposed using neuro-fuzzy systems based on boolean relationships. Regarding the systems to be controlled, there is a distributed power generation system since its integration with the distribution network presents challenges, one of these being the control of the power flow to regulate the voltage at the nodes of the distribution system. It is also considered the control of hydraulic systems SISO (Single Input Single Output) first for the control of a conical tank and then for two coupled spherical tanks; finally, the control is carried out for a coupled MIMO (Multiple Input Multiple Output) system. These systems are characterized by presenting different non-linearities which make their control difficult.

The document is organized as follows. First, the justification and problem formulation are presented. The proposed objectives are described in the following chapter. Subsequently, the background of the project is shown, making a review on proposals and related works. The following chapter presents the theoretical framework. Then the development of the proposal is described where the design process of fuzzy systems based on boolean relationships is presented, with these concepts two compact structures are established which are used for the development of an adaptive control system; later, the adaptive control application is carried out in a power distribution system, two applications are also carried out for SISO hydraulic systems and one for MIMO. Finally, the conclusions of the results obtained are presented.

Keywords: Adaptive, control, fuzzy logic, neuro-fuzzy.

Introducción

Para realizar la implementación de sistemas de control, las redes neuronales son una buena alternativa en casos donde resulta difícil el diseño de un controlador por métodos convencionales, debido a la complejidad del sistema, no linealidades y variación de sus parámetros, entre otros aspectos.

Una de las desventajas de las redes neuronales es su inicialización, convencionalmente de forma aleatoria, lo cual no garantiza tener la misma configuración de parámetros cada vez que se realiza su entrenamiento; otra dificultad consiste en establecer la arquitectura adecuada de red neuronal. Ante esto, una alternativa es la integración de los sistemas de lógica difusa con redes neuronales teniendo así los sistemas neuro-difusos; los cuales permiten tener la configuración del sistema de control basado en un conocimiento preliminar de la planta a controlar. Adicionalmente, en el área de la lógica difusa existen estudios sobre esquemas simplificados que permiten disminuir la complejidad del sistema, tal como es el planteamiento de los sistemas de lógica difusa basados en relaciones booleanas. Este tipo de sistemas permiten establecer una estructura compacta y una fácil inicialización del sistema neuro-difuso.

Cuando se tienen sistemas complejos, no lineales y variantes en el tiempo, una alternativa para su control consiste en las estrategias adaptativas, por cuanto este enfoque es adecuado para el monitoreo del sistema de control con parámetros desconocidos y variantes. Este esquema de control puede implementarse mediante sistemas neuro-difusos ya que permiten establecer la estructura del controlador mediante el conocimiento previo del sistema. Posteriormente, utilizando de forma iterativa datos medidos y algoritmos de optimización, se efectúa el respectivo ajuste del sistema de control logrando así su adaptabilidad.

En este documento se propone un esquema de control adaptativo empleando sistemas difusos basados en relaciones booleanas. Para esto se formula el proceso de diseño y se establecen estructuras compactas utilizadas en la implementación del sistema de control adaptativo. Posteriormente este esquema de control se aplica en un sistema de distribución de energía donde se considera la variación de la carga y la conexión de varios generadores. También se realiza la aplicación del control de dos sistemas hidráulicos SISO (*Single Input Single Output*) y un sistema MIMO (*Multiple Input Multiple Output*).

El enfoque propuesto utiliza sistemas difusos compactos basados en relaciones booleanas ya que permiten establecer la estructura a utilizar y la inicialización de parámetros.

El sistema de control adaptativo emplea dos sistemas neuro-difusos, uno para la identificación y otro para control. De forma iterativa, el modelo obtenido de la identificación de la planta durante cada proceso adaptativo se emplea para realizar el entrenamiento del controlador. Los resultados muestran un buen desempeño del sistema de control propuesto logrando de forma efectiva el proceso de adaptación en la mayoría de los casos experimentales considerados.

Sobre las plantas a controlar se tiene un sistema de distribución de energía ya que la implementación de los sistemas de generación distribuida y su integración con la red de distribución presentan desafíos para su funcionamiento, siendo uno de estos el control del flujo de potencia para regular el voltaje en los nodos del sistema de distribución. También se realiza el control de sistemas hidráulicos SISO, el primero consiste en un tanque cónico y el segundo compuesto por dos tanques esféricos acoplados, finalmente se muestra el control de un sistema hidráulico MIMO acoplado. Estos sistemas se caracterizan por presentar diferentes no linealidades las cuales dificultan su control.

El documento está organizado en tres partes fundamentales:

- Parte 1:** Contiene información sobre la formulación del proyecto; descripción, justificación, objetivos y estado del arte.
- Parte 2:** Dispuesta para el marco teórico del proyecto compuesto por una revisión sobre lógica difusa, identificación y control supervisado.
- Parte 3:** En este grupo de capítulos se presenta el desarrollo del proyecto, donde primero se formula el proceso de diseño de los sistemas difusos basados en relaciones booleanas. Con estos conceptos se diseñan dos estructuras compactas utilizadas para el desarrollo del sistema de control adaptativo; posteriormente, se efectúa la aplicación de control adaptativo en un sistema de distribución de energía, también se realizan dos aplicaciones para sistemas hidráulicos SISO y una para MIMO. Finalmente, se presentan las conclusiones de los resultados del sistema propuesto.

Parte I

Descripción del proyecto

Capítulo 1

Formulación del proyecto

1.1. Justificación

En la actualidad, debido a las necesidades existentes sobre el control de procesos, se requieren sistemas con un buen desempeño en tiempo real tanto para tareas de monitorización como de control.

Gracias a las investigaciones en inteligencia computacional, el desarrollo de los sistemas basados en el conocimiento y los sistemas expertos, se han producido grandes avances en el tratamiento del conocimiento, factor importante para el desarrollo de sistemas empleados en el control de procesos [1, 2].

Una de las herramientas que emplean los sistemas basados en conocimiento es la lógica difusa, desarrollándose sobre esta teoría nuevas áreas de investigación, con lo cual se espera tener un mejor desempeño de estos sistemas. En el área de la lógica difusa también se aprecian estudios sobre esquemas simplificados que permiten disminuir la complejidad del sistema. Tal es el caso de los sistemas de lógica difusa basados en relaciones booleanas [3, 4].

Por su parte, las redes neuronales, debido principalmente a su capacidad de aprendizaje y adaptación, resultan apropiadas para sistemas que procesan y adquieren automáticamente información numérica. Sin embargo, su modo de procesar la información resulta difícil de expresar en términos de lenguaje natural, lo que muchas veces dificulta emplear un conocimiento heurístico en las tareas de diseño. Los sistemas difusos, debido a su habilidad para incorporar a su estructura información lingüística tomada del experto humano, son útiles para el diseño de sistemas de control que por su complejidad, imprecisiones e incertidumbres de su entorno, resultan difíciles de modelar; sin embargo, su principal dificultad radica en su falta de adaptabilidad [5, 6, 7].

Según [8, 9], las redes neuronales emulan el *hardware* del cerebro, y los sistemas difusos el *software*. Los sistemas neuro-difusos son el resultado de la simbiosis de estas dos técnicas, estos son empleados en el diseño de sistemas inteligentes y teoría de control, mostrando buenos resultados en el caso de sistemas en entornos ruidosos, cambiantes y con cierto grado de imprecisión e incertidumbre.

Los sistemas neuro-difusos buscan aprovechar las características comunes como el mapeo no lineal de funciones, la tolerancia a fallos y la operación en tiempo real, entre otras. Con este enfoque se utiliza la interpretabilidad propia de los sistemas difusos mejorando la comprensión de la red neuronal, como también la capacidad de adaptación de las redes neuronales.

Tanto las redes neuronales como los sistemas de inferencia difusa son técnicas que se pueden considerar como aproximadores generales de funciones, teniendo la capacidad de estimar el valor de la función que caracteriza su relación entrada-salida [5, 6]. Estos sistemas son usualmente construidos desde un sistema difuso basado en un conocimiento previo [10].

Dada la dificultad para establecer una estructura determinada de una red neuronal, este aspecto se puede abordar considerando un conocimiento previo de la información como un sistema de inferencia difusa. Los sistemas de inferencia difusa basados en relaciones booleanas son una técnica en desarrollo que permite establecer sistemas de inferencia compactos y simplificados que pueden ser empleados para aplicaciones supervisadas.

1.1.1. Sistemas neuro-difusos

Las redes neuronales son una buena alternativa cuando se tienen sistemas y fenómenos complejos donde su descripción o aplicación de técnicas formales es limitada o no arroja buenos resultados. Esto se logra ya que las redes neuronales proporcionan una buena adaptabilidad de sus parámetros cuando se dispone de un conjunto de datos asociados al fenómeno o sistema que se quiera representar.

Particularmente en sistemas de control, las redes neuronales resultan una muy buena alternativa en situaciones donde resulta difícil el diseño de un sistema de control por métodos convencionales, debido a la complejidad del sistema, no linealidades y variación de sus parámetros, entre otros aspectos.

Una de las desventajas de las redes neuronales es su inicialización, la cual se suele realizar de forma aleatoria, lo que no garantiza tener la misma configuración de parámetros cada vez que se inicialice y entrene la red neuronal. Otra dificultad yace en establecer una arquitectura de red neuronal adecuada para el problema que se esté considerando [11].

Por su parte, la lógica difusa es una técnica de computación flexible que permite describir de forma lingüística el modelo del sistema, esto es muy útil cuando existe vaguedad e incertidumbre en las relaciones internas que definen el sistema [12].

De esta forma, los sistemas difusos proporcionan una alta interpretabilidad, destacándose los de tipo Mamdani; por su parte, los sistemas Takagi-Sugeno tienen menor interpretabilidad pero permiten una representación y entrenamiento como redes neuronales (sistema neuro-difuso). Con esto se logra una inicialización dada por la interpretabilidad del sistema difuso y la adaptabilidad de las redes neuronales.

Las reglas de un sistema neuro-difuso permiten incorporar conocimiento previo del sistema de interés, lo cual puede ser un instrumento útil para abordar la identificación y control de sistemas no lineales inciertos. Además, la lógica difusa generalmente se consi-

dera aplicable a plantas donde la experiencia de operadores humanos está disponible para proporcionar reglas cualitativas. Según el teorema de aproximación universal, los controladores difusos son lo suficientemente generales para realizar cualquier acción de control no lineal [13].

Por su parte, los sistemas difusos basados en relaciones booleanas permiten disminuir la complejidad del sistema difuso, adicionalmente, como se encuentran fundamentados en el diseño de automatismos se logra una interpretabilidad directa para sistemas de control. De esta forma, se aprecia la posibilidad de aplicar la adaptabilidad de un sistema neuro-difuso basado en relaciones booleanas para la implementación de un sistema de control adaptativo. Con este enfoque, mediante sistemas neuro-difusos, y de forma adaptativa, se identifica la planta la cual se emplea para la optimización del controlador.

1.1.2. Sistemas de control adaptativo

Desde una perspectiva biológica, la adaptabilidad consiste en la capacidad que tienen los organismos para sobrevivir en un determinado ambiente, al lograrlo, prosperan y crean descendencia, pero en caso contrario, pueden desaparecer [14, 15]. Este principio según [16] y [17], se puede aplicar tanto en optimización como en sistemas de control inteligente adaptativo.

De acuerdo con [18, 19], la mayoría de las técnicas empleadas para diseñar sistemas de control se basan en el conocimiento y el entorno de la planta. Sin embargo, la planta puede ser notablemente compleja dificultando su modelado. De esta forma se pueden mejorar las técnicas de control con un método de identificación para conocer mejor la planta. Si la identificación del sistema es recursiva, donde el modelo de la planta se actualiza periódicamente utilizando estimaciones previas y nuevos datos, el control y la identificación se pueden realizar simultáneamente. Bajo este enfoque, el controlador se sintoniza durante el funcionamiento de la planta utilizando el modelo que se ajusta de forma recursiva. En la práctica, estos controladores se emplean en plantas con incertidumbre que varían lentamente en el tiempo. La aplicación de dichos esquemas surge en contextos como sistemas avanzados de control de vuelo para aeronaves y naves espaciales, manipuladores de robots, controles de sistemas de potencia, entre otros.

Para que un sistema de control adaptativo sea viable desde un punto de vista práctico, la estructura del sistema de control y los métodos de entrenamiento deben permitir el ajuste de parámetros en el tiempo requerido.

Al respecto, en [16] se propone utilizar algoritmos de optimización bio-inspirados en sistemas de control adaptativo. Este mismo autor en [17], señala que el control inteligente es el estudio para lograr el control automático a través de la emulación de sistemas inteligentes biológicos (biomimética). Considerando lo descrito en [20], los sistemas de control adaptativo son adecuados para el monitoreo del desempeño del sistema de control con parámetros desconocidos y variantes.

Según [20] y [21], el control adaptativo incluye un conjunto de técnicas que proporcionan un enfoque sistemático para el ajuste del controlador en tiempo real buscando mantener el rendimiento deseado del sistema de control cuando los parámetros de la plan-

ta son desconocidos o también variantes en el tiempo. De esta forma, para el control de sistemas no lineales inciertos, el control adaptativo es una técnica adecuada para manejar las incertidumbres del sistema utilizando un estimador de parámetros, al utilizar este enfoque, si el ajuste del controlador es satisfactorio, el efecto de la adaptación desaparece con el paso del tiempo [22]. Además, el control adaptativo ha mostrado ser aplicable en diferentes sistemas, desarrollando técnicas como el control adaptativo con modelo de referencia (*Model Reference Adaptive Control* MRAC), siendo esta un área importante a nivel teórico y práctico [23].

Sobre las técnicas de entrenamiento asociadas a los métodos de optimización, los algoritmos evolutivos han mostrado ser muy buena herramienta para acercarse a un valor óptimo global, sin embargo, requieren de varias ejecuciones y un número elevado de evaluaciones de la función objetivo [24, 25]. Por su parte, los métodos basados en el cálculo de gradientes presentan una rápida convergencia, no obstante, son susceptibles al punto inicial de búsqueda teniendo convergencia a mínimos locales [26, 27].

De acuerdo con [16], las técnicas de gradiente ofrecen métodos prácticos y efectivos para realizar la optimización (los métodos de gradiente se pueden utilizar en línea para ajustar todos los parámetros del sistema de control). El enfoque básico consiste en ajustar iterativamente los parámetros para minimizar el error de aproximación. Sin embargo, se suelen presentar mínimos locales ya que, en general, la función de objetivo que caracteriza el error de aproximación no es convexa.

En [16], se propone el entrenamiento mediante algoritmos bio-inspirados, los cuales permiten encontrar mínimos globales pero con una carga computacional alta, además que cada vez que se ejecutan estos algoritmos no se tienen los mismos resultados. Según [28], el cálculo de gradientes es muy utilizado en los algoritmos para el ajuste de sistemas neuronales, particularmente el método de gradiente descendente se suele emplear en el algoritmo *Back Propagation* (BP) para el entrenamiento de redes neuronales [29]. Considerando lo anterior, una alternativa para mejorar el desempeño del sistema de control adaptativo empleando algoritmos basados en gradientes, consiste en una adecuada configuración preliminar de los sistemas empleados tanto para la identificación de la planta como para el control.

Por su parte, los sistemas difusos tienen la capacidad de modelar procesos no lineales, como también adquirir información de un conjunto de datos empleando algoritmos de aprendizaje [8, 9]. De esta manera, un sistema difuso permite establecer una estructura y configuración preliminar del sistema de control. Esto también permite abordar la dificultad que tienen las redes neuronales para establecer esta estructura como también la inicialización de sus parámetros [11]. Con el sistema difuso se puede establecer un modelo preliminar que luego se ajusta con datos tomados durante el funcionamiento del sistema.

Al utilizar redes neuronales, la configuración inicial suele ser aleatoria, mientras que un sistema difuso permite establecer una configuración previa basada en un conocimiento preliminar del sistema. Los resultados con redes neuronales dependen de la estructura de la red, como también de una buena identificación y de la configuración inicial de los parámetros (aleatoria en un intervalo) [30].

En los sistemas de control con redes neuronales para el ajuste del modelo de la planta y el controlador sólo se requieren datos de entrenamiento, siendo esta una característica

importante en sistemas altamente complejos, con incertidumbre y variabilidad [17, 20]. Sin embargo, una mala inicialización de las redes neuronales influye drásticamente en el sistema de control. Al utilizar un sistema difuso en lugar de una red neuronal, se busca aliviar este problema al incluir información preliminar para tener una inicialización y configuración adecuadas. Según [31], al incluir la información lingüística de la planta se puede acelerar el proceso de adaptación.

Un sistema difuso adaptativo está equipado con un algoritmo de entrenamiento donde un controlador adaptativo se sintetiza a partir de una colección de reglas difusas, y los parámetros de las funciones de pertenencia cambian de acuerdo con alguna ley adaptativa para controlar una planta y de este modo seguir una trayectoria de referencia [13].

El sistema de control adaptativo neuro-difuso basado en relaciones booleanas permite tener una configuración inicial de parámetros y arquitecturas simples establecidas por un conocimiento preliminar. De esta forma, se pueden tener estructuras compactas de los sistemas difusos para lograr entrenamientos con pocas iteraciones ya que se pueden utilizar algoritmos de entrenamiento basados en gradientes, los cuales presentan una rápida convergencia a mínimos locales.

1.2. Formulación del problema

Muchos de los sistemas de control requieren de una adecuada identificación de la planta para el diseño del controlador. En caso de presentarse incertidumbre, no linealidades y variación en los parámetros del modelo, se puede ver comprometido el desempeño del sistema de control. Una alternativa consiste en un enfoque supervisado ajustando el modelo durante el funcionamiento de la planta. Como posibilidad, se pueden emplear redes neuronales debido a su capacidad de adaptación, sin embargo, aplicaciones de control supervisado con redes neuronales presentan dificultades para definir la arquitectura de la red y su configuración inicial. Los sistemas neuro-difusos permiten mejorar esta situación ya que se puede establecer una estructura y configuración inicial con este enfoque.

Adicionalmente, considerando las no linealidades en ciertas plantas, se requieren estrategias y arquitecturas adecuadas que asimilen estas características para lograr un buen desempeño del sistema de control.

Mediante los sistemas difusos basados en relaciones booleanas se busca suplir las anteriores situaciones dado que estos sistemas permiten el modelado no lineal de relaciones de entrada salida, y al mismo tiempo proporcionan un esquema simplificado para el cálculo del sistema difuso, permitiendo el desarrollo de un sistema de control adaptativo.

Capítulo 2

Objetivo general y objetivos específicos

2.1. Objetivo general

Desarrollar un marco de referencia para la aplicación de los sistemas neuro-difusos basados en relaciones booleanas para control adaptativo.

2.2. Objetivos Específicos

- Establecer un marco de referencia para los principios de diseño de los sistemas difusos basados en relaciones booleanas.
- Obtener un marco de referencia para el control adaptativo con sistemas neuro-difusos basados en relaciones booleanas.
- Desarrollo de casos de aplicación empleando sistemas neuro-difusos supervisados basados en relaciones booleanas.
- Establecer la efectividad de la propuesta realizada para los casos de aplicación considerados.

Capítulo 3

Antecedentes

3.1. Introducción

Este capítulo presenta los trabajos relacionados con el desarrollo del sistema de control neuro-difuso adaptativo basado en relaciones booleanas. En primer lugar, se efectúa una revisión de los trabajos previos sobre sistemas de inferencia difusa basados en relaciones booleanas. Luego se exploran las diferentes aplicaciones del control neuro-difuso adaptativo y su combinación con otras técnicas de control. Posteriormente, se revisan los trabajos realizados sobre sistemas de control MIMO mediante redes neuronales. Finalmente, se hace un recuento sobre técnicas de control en sistemas de generación distribuida de energía puesto que en este trabajo se realiza una aplicación para este tipo de sistemas con la estrategia de control adaptativa propuesta.

3.2. Sistemas de lógica difusa basados en relaciones booleanas

Los sistemas de lógica difusa han probado ser una buena alternativa para la solución de problemas en diversas áreas, especialmente para el control de procesos donde es posible diseñar un controlador de forma heurística [32, 33]. Dependiendo del tipo de problema, existen alternativas para el sistema de inferencia difusa como los modelos Mamdani, Larsen, Takagi-Sugeno-Kang y Tsukamoto [11].

Originalmente, los sistemas de inferencia difusa basados en relaciones booleanas (*Fuzzy Inference Systems - Based on Boolean Relations* FIS-BBR) se plantean como un posible mecanismo para la implementación del proceso asociado a la defusificación denominado DBR (*Defuzification Based on Boolean Relations*) [4, 34], sin embargo, después de varios desarrollos fue apreciable que se podía interpretar como un sistema de inferencia difusa, observando que el término DBR es adecuado solamente para referirse al proceso simplificado asociado a la defusificación. También se utilizó el término CBR (*Concretion Based on Boolean Relations*) para enfatizar el proceso de diseño basado en tablas de verdad. En trabajos previos se pueden encontrar estos términos, sin embargo, los FIS-BBR incorpo-

ran tanto el mecanismo de implementación como el proceso de diseño. En la figura 3.1 se puede apreciar la relación de estos conceptos.

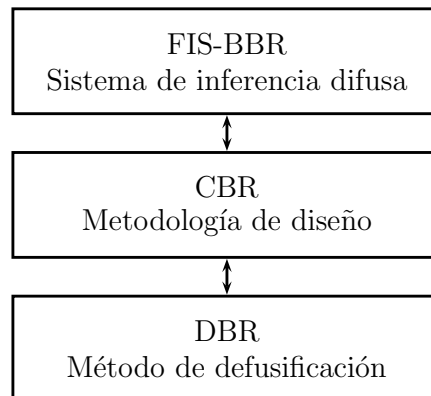


Figura 3.1: Relación entre FIS-BRB, CBR y DBR.

Los FIS-BBR buscan plantear un forma de implementación para los sistemas de lógica difusa tomando como referencia el diseño de automatismos con álgebra booleana, ya que son muy empleados en control de procesos por su facilidad de implementación. Esta técnica de diseño considera los sensores, actuadores y las relaciones booleanas empleadas en las estrategias de control [3]. Los FIS-BBR se puede considerar como una variación de los sistemas de inferencia difusa tipo Mamdani y Takagi-Sugeno.

Las características que se buscan tener con este enfoque son:

- Facilitar la implementación computacional del sistema de inferencia difusa.
- Tener un mejor desempeño en cuanto al tiempo de procesamiento del sistema de inferencia difusa.
- Proporcionar una metodología de diseño para sistemas de lógica difusa.

El esquema de inferencia sobre el cual se encuentra fundamentado los FIS-BBR consiste en una forma de lógica argumentativa [3], siendo esta la utilizada en el diseño de circuitos digitales y de automatismos. Por esta razón, uno de los aspectos de importancia en estos sistemas es la posibilidad de realizar la simplificación de las reglas de inferencia. Un trabajo que muestra el acercamiento a este proceso de simplificación de reglas se tiene en [35].

Aplicaciones iniciales del concepto de diseño de sistemas difusos basados en relaciones booleanas se pueden apreciar en [36] y [37], en estos trabajos se busca tener un método simplificado para llevar a cabo el proceso de defusificación. Sobre algunos trabajos realizados en el desarrollo de sistemas de inferencia difusa basados en relaciones booleanas, en [34] se lleva a cabo el diseño e implementación de un controlador difuso auto-sintonizado sobre microcontroladores, aplicado al control del péndulo invertido. En [38, 39], figura una propuesta para la predicción de series de tiempo caóticas, por su parte, en [40] se formulan los sistemas de inferencia difusa tipo II basados en relaciones booleanas. En [41] se muestra una primera presentación de la arquitectura de un sistema de inferencia difusa basado en relaciones booleanas. Sobre aplicaciones prácticas de control, en [42] tiene

lugar la implementación para un sistema de conversión de energía DC-DC de tipo *Buck Boost*. Así mismo, en [43] se presenta el diseño de un controlador para la orientación de un satélite CUBESAT. Por su parte, en [44] se muestra el diseño e implementación de un controlador basado en relaciones booleanas tipo II para estabilizar un péndulo invertido. En [45] se comparan controladores de lógica difusa tipo II de intervalo con controladores basados en álgebras de tres elementos. Finalmente, en [46] se analizan sistemas de inferencia difusa de dos y tres elementos, este trabajo se enfoca en el diseño y análisis de sistemas dinámicos con histéresis (*Hysteresis-Backlash*).

En relación con proceso de diseño, en [47] se emplea álgebra de Kleene para transformar ecuaciones lógicas binarias en funciones de lógica difusa. Con este mismo enfoque, en [48] se describe un método para simplificar fórmulas usando álgebras finitas. Además, el trabajo presentado en [49] ofrece el modelado de la vaguedad, la ambigüedad y la contradicción con la lógica difusa trivalente y tetravalente, empleando estructuras algebraicas de Kleene y De Morgan, centrándose en las tablas y la representación gráfica.

3.3. Aplicaciones de control neuro-difuso adaptativo

El control adaptativo es una alternativa adecuada y robusta para hacer frente a las incertidumbres relacionadas con la predicción de los parámetros de un sistema dinámico. Para la implementación de este tipo de sistemas se debe contar con las estructuras adecuadas para el controlador y el modelo de la planta, como también de los métodos de entrenamiento que permitan ajustar los parámetros de forma efectiva. Al respecto, para la identificación de la planta y la optimización del controlador se pueden utilizar algoritmos evolutivos, sin embargo, el número de iteraciones es mayor que en un algoritmo basado en gradiente [26, 50].

Sobre la estructura, los sistemas difusos permiten modelar procesos no lineales y adquirir información de un conjunto de datos empleando algoritmos de aprendizaje. A diferencia de las redes neuronales, los sistemas de lógica difusa permiten utilizar fácilmente el conocimiento de los expertos en forma directa o como punto de partida para su optimización [8, 9]. Las estructuras de control neuro-difusas, a pesar de todas las ventajas de las características de las redes neuronales y los sistemas de inferencia difusa, presentan limitaciones al incrementar la cantidad de reglas difusas asociadas al orden del sistema o al número de conjuntos difusos utilizados para cada entrada [51].

Sobre los principales aspectos identificados en la literatura relacionados con el control neuro-difuso adaptativo, en la figura 3.2 se relacionan los temas que se abordan en esta sección.

3.3.1. Aplicaciones varias

Sobre aplicaciones relacionadas con el desarrollo de sistemas de control neuro-difuso adaptativo, en [52] se propone un esquema para atenuar las respuestas sísmicas para puentes atirantados, considerando diferentes escenarios paramétricos, así como las condiciones del sitio y características sísmicas. En el estudio, el desempeño de los métodos adaptati-

vos se compara con los esquemas pasivos y no controlados antes y después de considerar variaciones paramétricas. El esquema de control adaptativo simple brinda una reducción general exitosa para variaciones de los parámetros del puente.

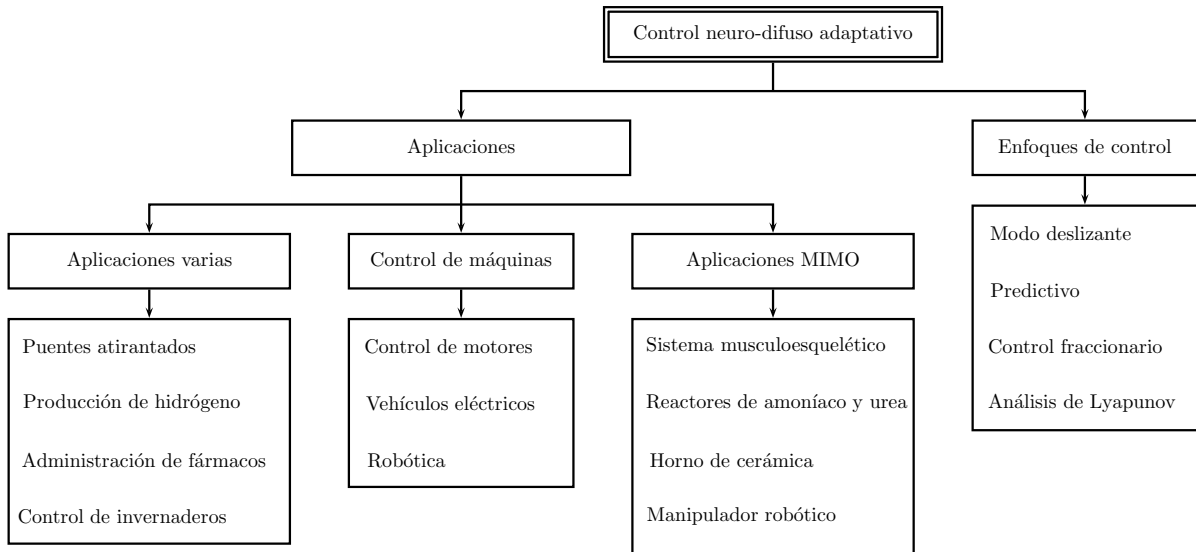


Figura 3.2: Principales aspectos identificados del control neuro-difuso adaptativo.

Otra aplicación industrial se puede observar en [53], donde se estudió el control de un reformador a escala piloto para la producción de hidrógeno (H_2) mediante el reformado de metano (CH_4) con CO_2 . Para modelar el proceso se utilizó un enfoque ANFIS (*Adaptive Neuro Fuzzy Inference System*) mediante conjuntos y sistemas difusos de tipo II, permitiendo una linealización por partes del modelo no lineal. Las funciones de membresía (pertenencia) en forma de campana generalizada se optimizan utilizando datos de entrenamiento empíricos. Este modelo se utiliza dentro de un marco MPC (*Model Based Predictive Control*) para diseñar estrategias de control óptimas. Se implementa un proceso de optimización sin restricciones usando un modelo cuasi-lineal por partes mediante la adopción de pesos apropiados en las entradas de control en la función objetivo. Estos pesos se ajustan para evitar la saturación del actuador.

Una aplicación médica, se puede apreciar en [54], donde se desarrollan estrategias de control en lazo cerrado para la infusión y administración de fármacos. Esto se puede utilizar especialmente en anestesia durante diferentes cirugías para estabilizar al paciente en la condición de conciencia deseada. Los autores proponen un controlador adaptativo neuro-difuso para superar los desafíos actuales en el control de lazo cerrado de la anestesia, como son la variabilidad inter e intrapaciente, dinámicas complejas y no lineales, ruidos de medición y alteraciones quirúrgicas, así como la presencia de subimpulso y rebasamiento en la fase de inducción.

3.3.2. Control de motores, vehículos eléctricos y robótica

Otra aplicación del control neuro-difuso adaptativo consiste en el control de motores eléctricos, al respecto, en [55] se propone un esquema de accionamiento para un motor de inducción controlado por un sistema neuro-difuso. El esquema de control propuesto utiliza la amplitud del flujo del estator y los errores de par electromagnético a través de un

sistema de inferencia neuro-difuso adaptativo (ANFIS), para actuar tanto en la amplitud como en el ángulo del voltaje de referencia deseado.

Otro trabajo relacionado se puede apreciar en [56], donde se busca la minimización del consumo total de energía (batería eléctrica y combustible) de vehículos híbridos (hidráulicos-eléctricos) a través de un enfoque combinado de gestión energética que incorpora elementos de lógica difusa, redes neuronales y algoritmos basados en reglas. En este trabajo, la eficiencia global del vehículo se calcula teniendo en cuenta el motor eléctrico, la batería, el motor, la bomba hidráulica, el motor hidráulico y la transmisión. Para lograr un control en tiempo real se integra en el vehículo un algoritmo neuronal difuso adaptativo.

Un trabajo de robótica se puede observar en [57], donde se proponen algoritmos híbridos adaptativos de control neuro-difuso para el control de un manipulador con incertidumbres. Las salidas de los controladores se utilizan para producir la señal de actuación (acción) final basada en la posición actual y los errores de velocidad. De esta forma, el control del robot industrial incluye no linealidades, incertidumbres y perturbaciones externas en el diseño de las leyes de control.

3.3.3. Control de invernaderos

La regulación del clima en invernaderos es una aplicación común de los sistemas neuro-difusos, ya que un invernadero es un sistema complejo no lineal que proporciona a las plantas las condiciones ambientales adecuadas para crecer. El problema del control climático del invernadero es crear un entorno favorable para el cultivo con el fin de alcanzar resultados predeterminados de alto rendimiento, calidad y bajos costos. Al respecto, en [58] se presenta el diseño de un sistema de control para un invernadero que utiliza energía geotérmica para el sistema de calefacción. Se plantean cuatro técnicas: control PI (Proporcional Integral), control difuso, control con red neuronal artificial y el control neuro-difuso adaptativo, empleados para ajustar la temperatura interior del invernadero al valor requerido. Otro trabajo relacionado se puede apreciar en [59], que presenta la aplicación de un sistema ANFIS para hacer frente a la gran cantidad de incertidumbres en tales sistemas, y con el fin de mejorar el rendimiento, se utiliza un algoritmo genético (*Genetic Algorithm* GA) para adaptar los parámetros del controlador, como son el número y la forma de las funciones de pertenencia y los factores de escala.

3.3.4. Control de modo deslizante

El desarrollo de controladores de modo deslizante neuro-difuso adaptativo ha sido también de interés, al respecto, en [60] se investiga el problema del control de modo deslizante neuro-difuso adaptativo de orden fraccional (*Fractional Order* FO) para una clase de sistemas difusos singularmente perturbados, sujetos a las incertidumbres y perturbaciones externas. En este trabajo se presenta una superficie de modo deslizante difuso FO utilizando una función de Lyapunov para el análisis de la estabilidad robusta de la dinámica del modo deslizante. Además, se propone un sistema híbrido neuro-difuso (*Hybrid Neuro-Fuzzy Network System* HNFNS) para estimar la incertidumbre. El controlador de modo

deslizante difuso adaptativo FO está diseñado para impulsar las trayectorias de estado de sistemas difusos perturbados singularmente a la superficie del modo deslizante FO predefinido dentro de un tiempo finito.

Un trabajo relacionado se puede apreciar en [61], donde se presenta un sistema de control de modo deslizante desacoplado adaptativo neuro-difuso (*Decoupled Adaptive Neuro Fuzzy DANF*), para el problema de control del caos en una planta sin información precisa de su comportamiento. El sistema tiene capacidad de aprendizaje en línea para afrontar la incertidumbre y perturbación ajustando los parámetros del controlador. Los autores establecen que se requiere conocimiento previo de la planta en el proceso de diseño y desarrollan un controlador de modo deslizante adaptativo desacoplado para controlar el sistema caótico de Lorenz.

Otra aplicación se observa en [62], con un método de control adaptativo para regular la velocidad de motores de inducción. Para este sistema se emplea un esquema de aprendizaje en línea basado en un algoritmo de entrenamiento de modo deslizante y sistemas difusos de tipo II. Mediante el ajuste de los parámetros de control se puede abordar el problema asociado a las incertidumbres y perturbaciones paramétricas. El mecanismo adaptativo de modo deslizante ajusta los parámetros de las funciones de pertenencia de tipo II de acuerdo con las entradas correspondientes al error de velocidad y su derivada. Dado que los parámetros del motor de inducción pueden variar y la información que se utiliza para construir las funciones de pertenencia y las reglas del sistema de lógica difusa es incierta, se selecciona la estructura neuro-difusa de tipo II como controlador.

3.3.5. Control de sistemas no lineales

En términos generales, los sistemas de control neuro-difuso se utilizan para plantas no lineales. Al respecto, en [63] se sugiere el diseño de control adaptativo neuro-difuso para el seguimiento en los sistemas dinámicos de control con no linealidades desconocidas. La planta se describe mediante un modelo difuso Takagi-Sugeno (TS), donde los submodelos locales se realizan a través de mapeos dinámicos de entrada y salida no lineales. La propuesta se basa en la aproximación de ciertos términos que involucran la derivada de la función de Lyapunov y las no linealidades del sistema desconocido. Además, se propone un esquema de reinicio que asegura la validez de la entrada de control. El análisis de estabilidad proporciona la ley de control y las reglas de adaptación para los pesos de la red.

Otro trabajo se puede apreciar en [64], donde se propone un método de control predictivo adaptativo para sistemas no lineales a través del sistema de inferencia adaptativo neuro-difuso (ANFIS). En el método propuesto, se utiliza un controlador predictivo generalizado no lineal (*Generalized Predictive Controller GPC*), el modelo de la planta se obtiene utilizando un sistema adaptativo. La dinámica del sistema se clasifica en dos partes, lineal y no lineal. La parte lineal se aproxima usando la técnica de estimación de mínimos cuadrados y la parte no lineal mediante un identificador basado en ANFIS. El controlador se actualiza utilizando de forma adaptativa la predicción obtenida con estos dos modelos. El método propuesto se puede implementar en tiempo real, además no hay necesidad de una fase de pre-entrenamiento de la red. Como caso de aplicación, se considera un sistema de control del nivel de líquido y un reactor (tanque) con agitado continuo

(*Continuous Stirred Tank Reactor CSTR*).

Por su parte, en [51], se propone un controlador adaptativo neuro-difuso basado en cuaterniones, buscando así reducir el problema asociado al número de reglas grandes utilizando el concepto de retropropagación de cuaterniones. Además, se utiliza aprendizaje por refuerzo mediante la evaluación de la respuesta del sistema por un crítico. Como caso de aplicación, se considera el control de un disco giratorio caótico.

Un trabajo adicional se puede observar en [65], donde se desarrolla un método de control dinámico de superficie con retroceso de orden fraccional (*Fractional Order Backstepping Dynamic Surface Control DSC*), para hacer frente al problema de estabilización de sistemas no lineales de orden fraccionario con incertidumbres y perturbaciones externas. En cada paso adaptativo, se emplea un sistema neuro-difuso para aproximar la función no lineal desconocida existente en el subsistema de orden fraccional. Además, se diseña un filtro de orden fraccionario modificado para evitar el problema de explosión de complejidad causado por el procedimiento recursivo. Luego, con base en la teoría de estabilidad de Lyapunov de orden fraccionario, se propone el controlador DSC de retroceso adaptativo para garantizar la estabilidad de los sistemas de bucle cerrado.

Finalmente, en [66] se presenta un controlador no lineal híbrido para el seguimiento de la velocidad y la altitud de naves espaciales. Sobre la base del aumento de la dinámica y la cinemática de las naves espaciales, se deriva una formulación pseudo-lineal para el desarrollo del controlador, la cual sigue un esquema MSDRE (*Modified State-Dependent Riccati Equation*), donde un controlador neuro-difuso está diseñado utilizando un sistema ANFIS que utiliza las soluciones fuera de línea del MSDRE. El esquema de control combinado se aplica de acuerdo con grandes intervalos de tiempo de las soluciones MSDRE para obtener los pares de control óptimos, mientras que a lo largo de cada intervalo de tiempo, el controlador ANFIS proporciona la señal de control requerida. La estabilidad asintótica global del MSDRE se investiga utilizando el teorema de Lyapunov y se verifica mediante simulaciones de Monte Carlo.

3.3.6. Control de sistemas MIMO

Sobre aplicaciones de sistemas MIMO, en [67] se investiga un modelo musculoesquelético de múltiples entradas y múltiples salidas (MIMO) del brazo humano con seis músculos, y se propone un controlador neuro-difuso adaptable óptimo para controlar el punto final del brazo. La adaptabilidad y la optimización de la fuerza muscular son características importantes del controlador neuro-difuso propuesto.

Sobre otra aplicación MIMO, en [68] se presentan tres estrategias de control para reactores de amoníaco y urea. El primero es un controlador predictivo adaptativo basado en modelo (*Adaptive Model Predictive Controller*), el segundo es el control predictivo del modelo de red neuronal adaptativa (*Adaptive Neural Network Model Predictive Control*). El tercero es un controlador de modo deslizante neuro-difuso adaptativo (*Adaptive Neuro Fuzzy Sliding Mode Controller*). El objetivo principal de estos controladores es lograr la estabilización de la concentración de salida de amoníaco y urea, una velocidad de reacción estable, un aumento en la conversión de monóxido de carbono (CO) en dióxido de carbono (CO₂) para reducir el efecto de contaminación y un aumento en las producciones

de amoníaco y urea, manteniendo la relación NH_3/CO_2 igual a 3 para reducir el CO_2 y NH_3 sin reaccionar, y la temperatura de los dos reactores en los rangos de operación adecuados por cambio de parámetros del reactor o perturbación externa.

Por su parte, en [69] se presenta un estudio comparativo utilizando ANN y un sistema de inferencia neuro-difuso para el control de temperatura no lineal MIMO de un horno de rodillos utilizado en la fabricación de baldosas cerámicas.

Finalmente, en [70] se presenta un sistema de control adaptativo de modo deslizante terminal (*Terminal Sliding Mode TSMC*) utilizando lógica difusa para un manipulador robótico rígido de dos eslabones. El control de modo deslizante terminal (TSM) se desarrolló para proporcionar una convergencia más rápida y un control de mayor precisión que el control deslizante lineal basado en hiperplano. El controlador propuesto combina un TSM continuo con un algoritmo de aprendizaje adaptativo y un sistema de lógica difusa para estimar la dinámica de la planta controlada. De este modo, se busca garantizar la estabilidad en bucle cerrado y la convergencia en tiempo finito de los errores de seguimiento.

3.4. Sistemas de control MIMO mediante redes neuronales

De acuerdo con [71], resolver problemas cooperativos para sistemas de agentes múltiples es un desafío importante. Los problemas se vuelven más complejos si los agentes controlados son sistemas no lineales de múltiples entradas y múltiples salidas, donde se carece de conocimiento de la dinámica del sistema interno y se ven afectados por perturbaciones externas. Al respecto, un ejemplo se puede apreciar en [72], donde se desarrolla un esquema de control de red neuronal simple para una clase de sistemas no lineales MIMO de tiempo discreto con entradas de forma triangular y perturbaciones. El sistema estudiado es representado mediante un modelo NARMAX (*Nonlinear Autoregressive Moving Average Model with Exogenous Inputs*). Las redes neuronales de una sola capa se utilizan como emuladores de los controles deseados. La estabilidad del sistema de lazo cerrado se analiza mediante el método de Lyapunov.

De forma gráfica, en la figura 3.3 se describen los principales aspectos identificados en la literatura relacionados con sistemas de control MIMO utilizando redes neuronales. Estos aspectos se presentan a continuación.

3.4.1. Enfoque de control predictivo

Con relación al desarrollo de sistemas de control predictivo, en [71] se propone un esquema de control con redes neuronales (*Neural Networks NN*) para sistemas MIMO no lineales que incorpora una ley de actualización y un algoritmo de control en línea que actualiza los parámetros en un paso iterativo. Con la ayuda de la teoría de Lyapunov, los autores muestran que los errores de aproximación del peso y los errores de seguimiento cooperativo están uniformemente acotados en última instancia (*Uniformly Ultimately*

Bounded UUB), y los parámetros del sistema convergen a los valores aproximadamente óptimos.

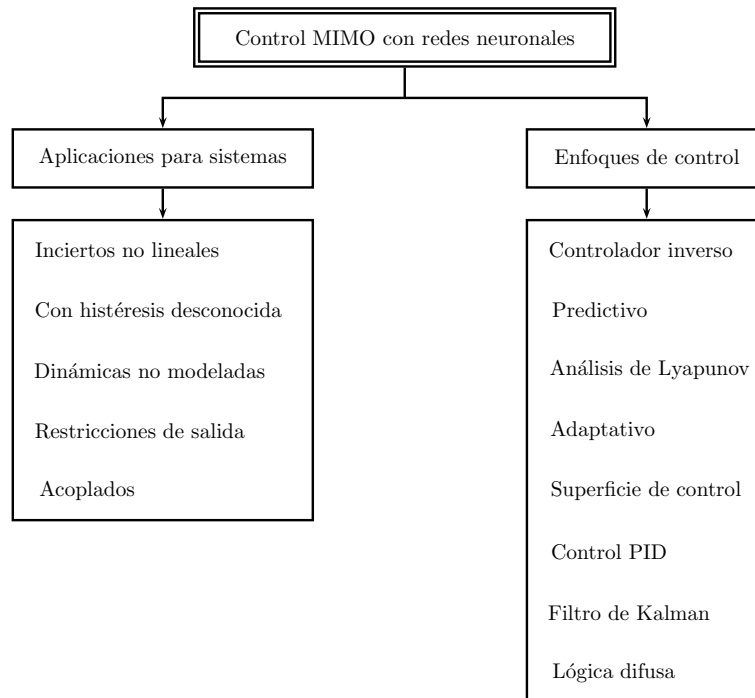


Figura 3.3: Principales aspectos identificados de sistemas de control MIMO mediante redes neuronales.

Otro trabajo relacionado se puede observar en [73], donde se estudia la identificación del sistema y el control predictivo generalizado de una clase de modelos MIMO. El problema de control predictivo generalizado con parámetros desconocidos se aborda encontrando una secuencia de las acciones de control. El esquema de control predictivo generalizado se formula como un problema de programación cuadrática con restricciones de desigualdad lineal, el cual se resuelve a través de una red neuronal de proyección generalizada con estructura simple y un número reducido de neuronas. Los parámetros desconocidos de los modelos se estiman mediante un algoritmo de gradiente estocástico.

Adicionalmente, en [74] se presenta un algoritmo de control predictivo basado en redes neuronales de restricción como modelo interno no lineal. El algoritmo de ajuste está basado en un filtro de Kalman que utiliza un algoritmo de programación cuadrática secuencial para calcular la siguiente acción de las variables manipuladas del proceso.

3.4.2. Enfoque de control adaptativo

Sobre el desarrollo con el enfoque adaptativo, en [75] se investiga el control neuronal adaptativo en tiempo finito y los problemas de desacoplamiento para sistemas no lineales MIMO con perturbaciones y estructura de retroalimentación no estricta. En el procedimiento de diseño se emplean redes neuronales para estimar las no linealidades desconocidas y se utiliza la desigualdad de Young para hacer frente a los términos de perturbación derivados de todos los subsistemas. Para caracterizar el rendimiento de atenuación de perturbaciones del control adaptativo de tiempo finito, se desarrolla un criterio denominado

desacoplamiento de cuasi perturbaciones en tiempo finito. El controlador neuronal adaptativo se diseña mediante el método de retroceso y la selección adecuada de la función de Lyapunov.

Por su parte, en [76] se investiga el control de seguimiento neuronal adaptativo basado en observadores de perturbaciones de una clase de sistemas MIMO en presencia de dinámicas no modeladas, incertidumbres del sistema, perturbación variable en el tiempo y zona muerta. Se propone un método de control neuronal adaptativo combinado con la técnica de retroceso y las redes neuronales de función de base radial (*Radial Basis Function Neural Network* RBFNN). En los diseños de retroceso recursivo, se emplea la red neuronal para la aproximación de la incertidumbre. El observador de perturbaciones está desarrollado para proporcionar un aprendizaje eficiente de la perturbación compuesta que incluye el efecto de la perturbación variable en el tiempo y el error de aproximación de la red neuronal.

En [77] se desarrolla una estrategia de control adaptativo que combina el controlador inverso de red neuronal (*Neural Network Inverse Controller* NNIC) con el observador de perturbaciones (*Radial Basis Function Network Disturbance Observer* RBFNDOB) para un sistema MIMO con perturbaciones internas y externas de fase no mínima. Dado que el modelo inverso del sistema es inestable debido a la fase no mínima, se construye una pseudo-planta, luego se usa el RBFN para identificar el modelo inverso de pseudo-planta que puede rastrear las variaciones de parámetros del sistema. Al copiar la estructura y los parámetros del identificador se obtiene la NNIC. Conectando en cascada la NNIC con la planta original, el sistema MIMO se puede desacoplar y linealizar en sistemas SISO independientes. Para el sistema desacoplado independiente, el RBFNDOB emplea un RBFN para observar las perturbaciones externas, este valor estimado se usa como un término de compensación de retroalimentación en el controlador.

3.4.3. Enfoque adaptativo para sistemas con incertidumbre

Sobre trabajos que consideran la incertidumbre, en [78] se propone un esquema de control de retroalimentación para sistemas inciertos no lineales MIMO, que combina un controlador no lineal basado en inversión con un compensador basado en redes neuronales. El controlador no lineal basado en inversión actúa como controlador principal, y se utiliza una red neuronal con una ley de actualización adaptativa para modelar la dinámica del sistema desconocido. Se emplea un controlador de estructura variable para eliminar el efecto de los errores de aproximación de la red neuronal y garantizar la estabilidad del sistema.

Adicionalmente, en [79] se propone un enfoque de control de seguimiento de retroalimentación de salida de redes neuronales adaptativas para una clase de sistemas no lineales MIMO con histéresis desconocida, y estados incommensurables. Las redes neuronales de función de base radial se utilizan para aproximar las funciones no lineales desconocidas de los sistemas controlados, y se diseña un observador para estimar los estados no medidos. Las señales filtradas se introducen para eludir el problema de bucle algebraico encontrado en la implementación del controlador, además se emplea una técnica de compensación adaptativa para resolver el problema de histéresis.

Un trabajo adicional se puede observar en [80], donde se presenta un control adaptativo indirecto estable con redes neuronales recurrentes para plantas no lineales multivariables con dinámica desconocida. El esquema de control se compone de un modelo neuronal instantáneo adaptativo, un controlador neuronal basado en redes de aprendizaje recurrente en tiempo real (*Real Time Recurrent Learning* RTRL) completamente conectadas, y una ley de actualización de parámetros en línea. Los rendimientos de bucle cerrado, así como las condiciones suficientes para la estabilidad asintótica, se derivan del enfoque de Lyapunov de acuerdo con el parámetro de tasa de actualización adaptativa.

En [81], se aborda el problema de control de los sistemas estocásticos no lineales MIMO con ganancias desconocidas de alta frecuencia. Este trabajo propone un algoritmo de red neuronal adaptativa que permite que las ganancias de alta frecuencia varíen en el tiempo. El enfoque basado en la ganancia de Nussbaum, y el mecanismo de red neuronal adaptativa se combinan para que todas las señales en el sistema de bucle cerrado estén acotadas.

Un trabajo relacionado se presenta en [82], que propone una ley de control de seguimiento continuo para una clase de sistemas dinámicos no lineales inciertos MIMO de alto orden con perturbaciones externas, y una matriz de dirección de control variable desconocida. El controlador propuesto consta de retroalimentación de alta ganancia, selector de matriz de ganancia Nussbaum, modelo de aproximación en línea y un término robusto. El modelo de aproximación en línea está representado por una red neuronal de dos capas. Se garantiza la continuidad de la señal de control para relajar los requisitos del ancho de banda del actuador y evitar el efecto de vibración. El rendimiento del seguimiento asintótico se logra teóricamente mediante el análisis estándar de Lyapunov.

3.4.4. Enfoque superficie de control

Acerca de estrategias que consideran superficies de control, en [83] se analiza el problema del control de superficie dinámico neuronal adaptativo (*Adaptive Neural Dynamic Surface Control* DSC), con los parámetros mínimos ajustables para una clase de sistemas no lineales de retroalimentación pura MIMO, con dinámicas no modeladas y restricciones de salida. Una señal auxiliar diseñada por las características de dinámica no modelada se utiliza para manejar las incertidumbres dinámicas. Las funciones continuas desconocidas de caja negra, producidas en el proceso de diseño del controlador, se aproximan mediante el uso de redes neuronales de función de base radial. Basado en un mapeo no lineal uno a uno, el sistema no lineal MIMO con restricciones de salida se transforma en un sistema no lineal MIMO de estructura de bloques sin restricciones de salida. Basado en el sistema transformado y la DSC, se desarrolla un esquema de control de seguimiento neuronal adaptativo robusto.

Otro trabajo relacionado se puede observar en [84], donde se investiga un método de control de superficie dinámico neuronal adaptativo para estabilizar sistemas no lineales MIMO con estados que varían en el tiempo y perturbaciones externas. Se propone un compensador, que combina redes neuronales de función de base radial adaptativas basadas en errores de aproximación y un observador de perturbaciones no lineales (*Nonlinear Disturbance Observer* NDOB) para mejorar el rendimiento de compensación de dinámicas no modeladas y perturbaciones externas. Para evitar que se violen las restricciones de

estado completo que varían en el tiempo, el controlador se diseña mediante la función de Lyapunov de barrera asimétrica que varía en el tiempo.

3.4.5. Aplicación control PID

En relación con las aplicaciones que consideran control de tipo PID (*Proportional Integral Derivative*), en [85] se propone un diseño de controlador conmutado adaptativo (*Adaptive Switch Controller ASC*) para un sistema no lineal MIMO, que comprende la conmutación en línea entre un controlador PID adaptativo de red neuronal y un controlador adaptativo indirecto de red neuronal. La ley de adaptación de la red neuronal ha sido desarrollada por el método de descenso de gradiente (*Gradient Descent GD*). El controlador PID adaptativo se basa en la red neuronal que combina el control PID y la estructura neuronal explícita. La estrategia de entrenamiento consiste en el ajuste en línea de los pesos del controlador neuronal usando el algoritmo de retropropagación para seleccionar la combinación adecuada de ganancias PID, de manera que el error entre la señal de referencia y la salida real del sistema converja a cero.

En [86], se propone el ajuste adaptativo de un controlador PID. Este enfoque está desarrollado para abordar una clase de sistemas no lineales MIMO. El controlador PID adaptativo está construido sobre la base de redes neuronales que combinan el control PID y la estructura neuronal explícita. La estrategia de entrenamiento consiste en el ajuste en línea de los pesos del controlador neuronal utilizando el algoritmo de retropropagación (*Back Propagation BP*), para seleccionar la combinación adecuada de ganancias PID de manera que el error entre la señal de referencia y la salida real del sistema converja a cero. El esquema de control se basa en un modelo de red neuronal, utilizando una tasa de aprendizaje variable, del sistema que se adapta mediante el método de gradiente descendente.

3.4.6. Aplicación con lógica difusa

Sobre aplicaciones que incorporan lógica difusa, en [87] se desarrolla un controlador híbrido de lógica difusa y red neuronal para sistemas MIMO. El sistema consta de un controlador de lógica difusa diseñado para controlar cada grado de libertad (*Degree Of Freedom DOF*) de un sistema MIMO de forma individual, y una red neuronal de acoplamiento adicional incorporada al controlador difuso para compensar los efectos de acoplamiento dinámico entre cada grado de libertad del sistema MIMO.

3.4.7. Aplicación control de llenado de tanques

Finalmente, en relación con el control de llenado de tanques, en [88] se presenta el diseño de un control tolerante a fallas usando dos controladores PI descentralizados, más controladores difusos para regular el nivel de los dos tanques inferiores, en un proceso de nivel de tanque cuádruple no lineal (*Quadruple Tank Level Process QTLP*). Se muestran algunos aspectos básicos del diseño de control descentralizado con respecto a la estabilidad y el rendimiento utilizando un modelo virtual de un proceso de tanque cuádruple.

La selección de la estructura de control se fundamenta en la matriz de ganancia relativa de rendimiento (*Performance Relative Gain Array* PRGA). Los controladores están basados en una estructura PI convencional más un sistema de inferencia difusa. La relación entre entradas/salidas se comprueba mediante una matriz de ganancia relativa (*Relative Gain Array* RGA), y luego, se divide el sistema de tanque cuádruple en dos subsistemas, controlando cada uno de ellos por separado.

3.5. Técnicas de control en sistemas de generación distribuida de energía

En la actualidad, la demanda de energía eléctrica se encuentra en aumento en todo el mundo, por lo cual, las centrales eléctricas y líneas de transmisión se deben diseñar para satisfacer la demanda de los clientes.

Por su parte, mediante la generación distribuida (*Distributed Generation* DG) los costos son menores que los requeridos para una central eléctrica, así como la expansión de los sistemas de distribución y transmisión. Para satisfacer el crecimiento de la carga, como también la calidad y confiabilidad del suministro de energía, se deben tener estrategias adecuadas para interconectar las mini-redes.

De la misma forma, se aprecia un rápido desarrollo en los sistemas de generación distribuida que utilizan una fuente de energía renovable. Bajo este enfoque se pueden usar sistemas híbridos que combinan más fuentes de energía. Pese a las ventajas que se logran con la generación distribuida se pueden tener problemas en el sistema de distribución como lo son: oscilaciones de frecuencia del sistema, violaciones de capacidad de las líneas de energía y el aumento de tensión. Al respecto, existen varias estrategias de control de tensión, en particular, los métodos de control de potencia reactiva han demostrado ser una buena alternativa para lograr tensiones dentro de un rango admisible sin reducir la producción de potencia activa.

En esta sección se hace un recuento sobre las diferentes estrategias de control empleadas para los sistemas de generación distribuida. Algunos estudios de revisión relacionados con los sistemas de generación distribuida se pueden observar en [89, 90, 91, 92, 93, 94].

Sobre la revisión de los sistemas de conversión de energía mediante inversores, en [89] se lleva a cabo un recuento del funcionamiento en paralelo de inversores y filtros de potencia activa en la generación distribuida. Adicionalmente, en [90] se revisan las topologías y estrategias de control de inversores conectados a redes multi-funcionales para mejorar la calidad de potencia.

Por su parte, en [91] se hace una revisión de la literatura sobre el control de carga-frecuencia para sistemas de generación convencional y distribuida. El objetivo del control de carga (*Load-Frequency Control* LFC) en un sistema interconectado de energía consiste tanto en mantener la frecuencia de cada área dentro de los límites, como en mantener los flujos de energía dentro de algunas tolerancias preestablecidas mediante el ajuste de las salidas de los generadores.

Otro trabajo referencial se puede apreciar en [92], donde se lleva a cabo una recopi-

lación sobre la generación eléctrica basada en un sistema integrado de energía renovable, observando de forma particular las configuraciones, opciones de almacenamiento, tamaño y sistemas de control. Por otra parte, en [93, 94, 95], se lleva a cabo un recuento sobre el control de los sistemas de generación de energía eléctrica distribuida para aplicaciones de microrredes. Una revisión sobre las técnicas de inteligencia computacional empleadas para detectar el funcionamiento en modo isla de unidades de generación distribuida se puede apreciar en [96].

Otro aspecto considerado en los sistemas de generación distribuida consiste en los métodos para la localización de generadores. Al respecto, en [97] se efectúa una revisión sobre el problema de asignación de DG desde el punto de vista de los algoritmos de optimización utilizados, los objetivos, las variables de decisión, el tipo de DG, las restricciones aplicadas y el tipo de modelado de incertidumbre empleado.

Adicional a estos artículos de revisión, se encuentran diferentes orientaciones de los sistemas de control aplicados a la generación distribuida. Un primer grupo de trabajos se enfoca en las diferentes estrategias empleadas para el control de la potencia activa y reactiva. Desde la perspectiva de las técnicas de control, se aprecian principalmente enfoques del tipo PID, predictivo, robusto y difuso. En estos trabajos se observan diferentes estrategias y aplicaciones de los sistemas de control siendo notable dos orientaciones, una se enfoca propiamente en el lazo de control del sistema de conversión de energía eléctrica DC-AC, es decir del inversor. La otra orientación se enfoca en el sistema de control para la interconexión de las diferentes unidades de generación del sistema distribuido. En la figura 3.4 se puede apreciar la relación de los sistemas de control aplicados a los sistemas de generación distribuida.

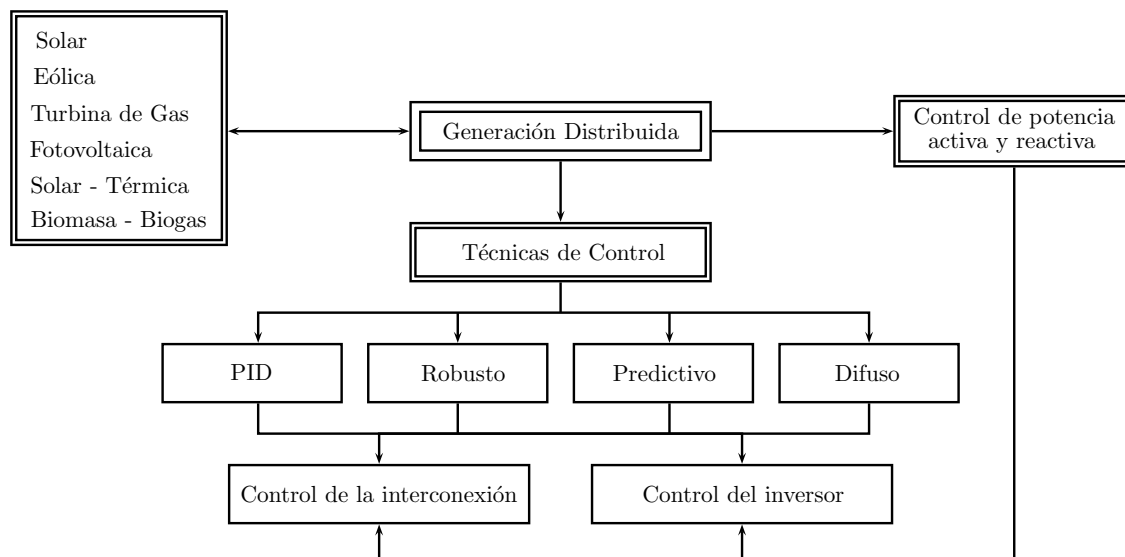


Figura 3.4: Principales características identificadas.

3.5.1. Enfoques generales sobre el control en sistemas de generación distribuida

En términos generales, los enfoques para el control de los sistemas de generación distribuida son: el control de la potencia activa y reactiva, interconexión de las unidades de generación, la detección del funcionamiento en isla (*island detection*) y particularmente, el control de los lazos propios del inversor DC-AC.

Las unidades de generación distribuida se interconectan con la red de suministro eléctrico mediante inversores trifásicos, por lo cual, las aplicaciones para el control de potencia activa y reactiva se enfocan en las estrategias para el control del inversor. Algunos trabajos donde se presentan estrategias para el control del flujo de potencia se pueden apreciar en [98, 99, 100, 101, 102]. Por su parte, trabajos relacionados con el control de la potencia reactiva se pueden observar en [103, 104, 105, 106, 107]. En términos generales, estos sistemas de control presentan tres lazos, siendo el lazo más interno el control de corriente, siguiendo por el control de voltaje y en el bucle exterior se tiene el control de potencia (activa y reactiva).

Otro aspecto de importancia en los sistemas de generación distribuida es la interconexión de las diferentes unidades de generación, donde se busca disminuir los efectos de la distorsión de las señales de voltaje buscando tener un sistema estable, en [108, 109, 110, 111, 112] se pueden apreciar trabajos relacionados. Un aspecto de particular interés es la interconexión de sistemas híbridos, es decir, unidades de generación de diferente naturaleza. Estos trabajos también se enfocan en el despacho de energía, para lo cual se suelen emplear técnicas de predicción (*forecast*) para establecer la potencia a suministrar en ciertos momentos.

Un enfoque para coordinar la inyección de potencia de los generadores distribuidos se puede apreciar en [113], donde se propone un método de control de voltaje de una red de distribución basado en la matriz de sensibilidad de voltaje. Para coordinar la inyección de potencia compleja de los generadores distribuidos se utiliza la matriz de sensibilidad, la cual se determina observando el efecto que tiene cada generador sobre los nodos de la red de distribución.

En funcionamiento en modo isla (*islanding*), es la situación en la cual un sistema de distribución se encuentra eléctricamente aislado del resto del sistema de generación y sin embargo, en este se conserva tensión durante un determinado tiempo. Aplicaciones enfocadas a la detección y control de este fenómeno se pueden apreciar en [114, 115, 116].

Finalmente, siendo el inversor el elemento con el cual se realiza el acople y la conversión de energía, se pueden apreciar varios trabajos relacionados con el desarrollo de estrategias de control aplicadas a este elemento. Con estos desarrollos se busca regular la corriente, voltaje y potencia suministradas. Algunos trabajos sobre el sistema de control del inversor se pueden observar en [117, 118, 119, 120, 121].

3.5.2. Control PID

Sobre aplicaciones del esquema PID, se presenta el control de diferentes sistemas de generación, también el control de la interconexión de las unidades de generación y el control de los lazos propios del inversor. Sobre la aplicación en particular de ciertas plantas de generación se tiene: turbinas hidráulicas [122], sistemas térmicos [123], turbinas de viento [124] y sistemas fotovoltaicos [125], entre otros.

En relación con el control de la interconexión de los sistemas de generación distribuida se pueden apreciar los trabajos desarrollados en [126, 127, 128, 130]. En estos trabajos también se suele emplear algoritmos de optimización para realizar el ajuste de los parámetros del controlador.

Sobre el control del inversor, en [131] se muestra el diseño de un controlador resonante y en [132] se presenta el diseño de una interfaz de electrónica de potencia para la integración en la red de sistemas de generación solar fotovoltaica.

Finalmente en [133], se presenta un esquema mejorado de control orientado a voltaje (*Voltage Oriented Control* VOC) para controlar un inversor conectado a la red trifásica. El inversor se considera la parte central del sistema de generación distribuida. Se utiliza un método de optimización para ajustar recursivamente los parámetros del controlador PID buscando lograr el rendimiento óptimo de la unidad DG conectada a la red.

3.5.3. Control robusto

Sobre aplicaciones de control robusto en sistemas de generación distribuida, se identificaron principalmente la interconexión de las unidades de generación y el control del elemento de conversión de energía (inversor).

Algunas aplicaciones de control robusto para la interconexión de unidades de generación se pueden apreciar en [134] para controlar voltaje, en [135, 136, 137] para controlar la frecuencia y en [138] para condiciones de desbalance. Particularmente, diseños basados en H_∞ se presentan en [139, 140, 141].

Sobre aplicaciones de control robusto que se enfatizan sobre los lazos de control del inversor, en [142] se describe un esquema predictivo robusto con sincronismo intrínseco para el control directo de potencia. Por su parte, en [143] se muestra el diseño de un control robusto para la generación distribuida en microrredes mediante el control directo del voltaje. En [144] se aplica el método de Lyapunov para el funcionamiento estable de generación distribuida basada en un convertidor multinivel. Adicionalmente, en [145] se presenta el diseño de un regulador de modo deslizante para el suministro de la máxima potencia. Finalmente, en [146] se presenta un control robusto híbrido para un inversor DC-AC de una sola fase con variaciones en el voltaje de entrada.

3.5.4. Control predictivo

Sobre esta técnica, en [147] se aprecia el diseño de un sistema de control predictivo para un inversor de cuatro brazos. Una aplicación adicional que se tiene del control predictivo para la generación distribuida consiste en el despacho de energía, lo cual se puede apreciar en [148].

Otra de las aplicaciones que se observa de la técnica de control predictivo consiste en la interconexión de las unidades de generación distribuida. Al respecto, en [149] se realiza la comparación de los esquemas de control predictivo centralizados y distribuidos para la regulación de oscilaciones amortiguadas electromecánicas. Adicionalmente, en [150] y [151] se presentan propuestas de sistemas de control predictivo para el control de carga-frecuencia (*Load Frequency Control* LFC). Al respecto, los sistemas de LFC buscan que después de un cambio de carga en cada área se elimine el error de frecuencia en régimen permanente, cada área debe además mantener en su valor programado el flujo de potencia. Un estudio sobre la gestión de la energía en las microrredes locales basado en estrategias de control predictivo se puede apreciar en [152]. Por su parte, en [153] se presenta el diseño de un sistema de control predictivo estocástico para el despacho de energía de un parque eólico.

Otro trabajo a considerar es [154], donde se incorpora el despacho económico (*Economic Dispatch* ED) para la operación de una microrred, proponiéndose una metodología de control predictivo. El sistema de control está completamente distribuido y cada sistema de generación distribuida puede comunicarse con otros nodos para el cálculo iterativo del proceso de optimización. Para establecer el desempeño del sistema se simula una microrred con diferentes fuentes de energía realizando una comparación con una estrategia de control centralizado.

3.5.5. Control difuso

El control difuso proporciona una alternativa cuando se tienen sistemas muy complejos, no lineales y con incertidumbre en sus modelos. En relación con los sistemas de inferencia difusa, se encuentran diferentes aplicaciones incluyendo:

- Control difuso PID.
- Detección de perturbaciones.
- Detección de operación en modo isla.
- Despacho de energía.
- Interconexión de sistemas de DG.
- Control del inversor.

Control difuso PID

Estas aplicaciones combinan la estrategia de control convencional PID con el esquema de razonamiento aproximado de los sistemas difusos. Particularmente, en [155] y [156] se muestra el diseño de sistemas de control PID con ganancia programada (*Gain Scheduling* GS). Mediante la información suministrada al sistema difuso se realiza el ajuste de los parámetros del controlador PID. Un trabajo similar se puede observar en [157] y [158],

donde se propone un esquema supervisado basado en un sistema difuso con el que se ajustan los parámetros de un controlador PID. Otra aplicación donde se tiene un controlador PID difuso adaptativo se puede apreciar en [159], que detalla el lazo de control del inversor para el control de potencia y el sincronismo con la red. Por su parte en [160], se optimiza un controlador PID difuso empleando el algoritmo de evolución diferencial. Sobre aplicaciones de sistemas difusos tipo II, en [161] se lleva a cabo la optimización de un controlador PID difuso tipo II de intervalo.

Detección de perturbaciones

Sobre este tipo de aplicaciones, en [162] se estudia la detección de las perturbaciones de los parques eólicos empleando un sistema neuro-difuso. Por su parte, en [163] se propone una metodología neuro-difusa para determinar el margen de carga cuando se tiene en cuenta la intermitencia de las fuentes.

Detección del funcionamiento en modo isla

Un acercamiento al problema de la detección del modo isla en generación distribuida usando la técnica de lógica difusa se puede apreciar en [164]. Por su parte en [165], se desarrolla un sistema de lógica difusa con enfoque híbrido tiempo-frecuencia para la detección del funcionamiento en modo isla para fuentes de generación distribuida. Finalmente en [166], se propone un esquema para la detección del modo isla utilizando filtrado *wavelet* y un sistema neuro-difuso.

Despacho de energía

Una aplicación para el despacho de energía empleando lógica difusa se pueden apreciar en [167] y [168], utilizando un algoritmo de programación cuadrática secuencial para obtener una solución óptima de la distribución de energía entre múltiples unidades de generación distribuida. Posteriormente, se implementa un sistema difuso para poner en práctica las estrategias óptimas encontradas previamente. Adicionalmente, los parámetros del sistema difuso se ajustan utilizando un algoritmo genético. Por su parte, una estrategia que emplea la predicción de series de tiempo para regular el despacho de energía se puede apreciar en [169].

Interconexión de sistemas de generación distribuida

Sobre trabajos enfocados a la interconexión de las unidades de generación distribuida, en [170] se diseña un sistema difuso orientado al control de carga-frecuencia para un sistema de generación multi-área. Por otra parte, en [171, 172, 173] se estudia la interconexión de unidades de generación de diferente naturaleza como celdas de combustible y bancos de baterías.

Otras estrategias de control difuso para interconexión de sistemas híbridos se pueden

apreciar en [174] y [175]. Como aspecto particular de estos trabajos se tiene la optimización de los sistemas de control empleando algoritmos evolutivos. Adicionalmente, en [176] se presenta el diseño de un controlador difuso robusto para un generador aislado conectado a un barraje infinito, en este trabajo, mediante una técnica de análisis de series de tiempo se obtiene información para anticiparse a los cambios en la carga.

Finalmente en [177], se presenta un sistema de control para la interconexión de recursos de generación distribuida a redes eléctricas mediante convertidores electrónicos de potencia. Se propone un sistema de control adaptativo difuso basado en la teoría de estabilidad de Lyapunov para el bucle de corriente del convertidor. Con el el esquema de control propuesto se busca un seguimiento adecuado de la corriente de referencia y proporcionar una respuesta dinámica apropiada.

Control del inversor

Sobre estrategias enfatizadas en los lazos de control del inversor en [178, 179, 180] se puede apreciar la aplicación de sistemas neuro-difusos en diferentes partes del sistema de conversión de energía.

En relación de otras aplicaciones, en [181] se presenta el diseño de un controlador difuso para una micro-turbina en un sistema de generación distribuida. Una aplicación para el control de una celda de combustible se puede observar en [182], en este trabajo se realiza la regulación del voltaje. Finalmente, en [183] se presenta el diseño de un controlador difuso para un generador de inducción con doble bobinado en el estator.

Parte II

Marco teórico

Capítulo 4

Elementos de lógica difusa

4.1. Introducción

La lógica difusa y los conjuntos difusos son utilizados para representar la información de un sistema en forma lingüística. Por lo general, esta información es dada por el conocimiento que tiene un experto acerca del sistema; de esta forma, se puede establecer el modelo de un sistema mediante observaciones cualitativas de su funcionamiento.

Para lograr estos modelos es necesario revisar los conceptos de conjuntos difusos y sistemas de lógica difusa, lo cual es de atención en el presente capítulo. Primero, se revisan las definiciones de las álgebras de Boole, Kleene y De Morgan, utilizadas para el cálculo proposicional. Luego se realiza la descripción de conjuntos booleanos y difusos como también sus propiedades. Posteriormente se presentan las operaciones de t-norma, t-conorma y complemento utilizadas en los sistemas difusos. Finalmente se revisan las diferentes formas de realizar una implicación difusa.

4.2. Álgebras De Morgan, Kleene y Boole

Las álgebras de Boole y de Kleene se utilizan para el diseño de los sistemas difusos compactos basados en relaciones booleanas; por lo tanto, los conceptos requeridos se detallan a continuación. Las siguientes definiciones se basan en [184, 185, 186, 187].

Una álgebra De Morgan $\mathcal{M} = (M, \vee, \wedge, \bar{\cdot}, 0, 1)$, está compuesta por un conjunto de elementos M con dos operaciones binarias \wedge (conjunción, intersección, o *meet*), \vee (disyunción, unión, o *join*), una operación unitaria $\bar{\cdot}$ (complemento) y las constantes 0 y 1. Considerando $A, B, C \in M$ se definen las propiedades de (4.1) hasta (4.9).

Las propiedades (4.1) a (4.4) indican que \mathcal{M} es un retículo (*lattice*), también si cumple con (4.5) el retículo es distributivo; de esta forma, las operaciones \wedge y \vee son asociativas, conmutativas y distributivas (\vee con respecto a \wedge y recíprocamente). La propiedad (4.6) indica que el retículo es acotado. Al cumplirse con las propiedades (4.1) a (4.9), esto implica que \mathcal{M} es una álgebra De Morgan [187].

$$A \wedge A = A \quad A \vee A = A \quad (4.1)$$

$$A \wedge B = B \wedge A \quad A \vee B = B \vee A \quad (4.2)$$

$$A \wedge (B \wedge C) = (A \wedge B) \wedge C \quad A \vee (B \vee C) = (A \vee B) \vee C \quad (4.3)$$

$$A \wedge (A \vee B) = A \quad A \vee (A \wedge B) = A \quad (4.4)$$

$$A \wedge (B \vee C) = (A \wedge B) \wedge (A \wedge C) \quad A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C) \quad (4.5)$$

$$A \wedge 1 = A \quad A \vee 0 = A \quad (4.6)$$

$$\overline{(A \wedge B)} = \bar{A} \wedge \bar{B} \quad \overline{(A \vee B)} = \bar{A} \vee \bar{B} \quad (4.7)$$

$$\bar{\bar{0}} = 1 \quad \bar{\bar{1}} = 0 \quad (4.8)$$

$$\bar{\bar{A}} = A \quad (4.9)$$

Una álgebra de Kleene $\mathcal{K} = (K, \vee, \wedge, \bar{\cdot}, 0, 1)$ es una álgebra De Morgan que cumple con la ecuación (4.10), siendo $K = \{0, u, 1\}$, donde el elemento adicional u pertenece al intervalo $(0, 1)$ [187].

$$A \wedge \bar{A} \leq B \vee \bar{B} \quad (4.10)$$

Por su parte, una álgebra de Boole $\mathcal{B} = (B, \vee, \wedge, \bar{\cdot}, 0, 1)$ es una álgebra de Kleene que satisface las condiciones (4.11) y (4.12), donde $B = \{0, 1\}$ se compone de dos elementos [187].

$$A \vee \bar{A} = 1 \quad (4.11)$$

$$A \wedge \bar{A} = 0 \quad (4.12)$$

En comparación con el álgebra de Boole, en el álgebra de Kleene se cumple con las relaciones (4.13) y (4.14).

$$A \vee \bar{A} \leq 1 \quad (4.13)$$

$$A \wedge \bar{A} \geq 0 \quad (4.14)$$

4.2.1. Álgebra de Kleene y cálculo proposicional difuso

El álgebra de Boole tiene solo dos elementos 0 y 1, permitiendo aplicaciones en lógica de tal forma que 0 significa *FALSE*, 1 *TRUE*, \wedge *AND*, \vee *OR*, y $\bar{\cdot}$ *NOT*. Si las proposiciones están asociadas a conjuntos X en términos de la pertenencia x es A , esto permite representar el conjunto como $A = \{x \in X | x \text{ es } A\}$. De esta manera, la función de pertenencia $A(x)$ corresponde a 1 cuando el elemento pertenece al conjunto A , y 0 cuando no [188].

En los sistemas de lógica difusa, las funciones de pertenencia no tienen únicamente valores 0 o 1 ya que pueden presentar valores en el intervalo $[0, 1]$, de esta forma, se puede definir un conjunto difuso como un conjunto de tuplas $\{\mu_A(x_i), x_i\}$, donde $\mu_A(x_i)$ representa la función de pertenencia de A para x_i . Utilizando álgebra de Kleene es posible representar un conjunto difuso mediante los valores 0, u y 1.

De acuerdo con Gehtke [187], diferentes álgebras (de valores de verdad) pueden proporcionar la misma lógica proposicional (*different truth value algebras can give the same*

propositional logic), en particular, la lógica proposicional difusa es la misma que la lógica proposicional de Kleene de tres valores (*fuzzy propositional logic is the same as the three-valued Kleene propositional logic*).

Según [184], la lógica difusa estándar se puede implementar mediante el álgebra de Kleene de tres elementos \mathcal{K} , así se puede representar un sistema difuso en forma de tablas. De esta manera, el álgebra de Kleene permite la implementación del cálculo proposicional difuso (*fuzzy propositional calculus*).

Entre las álgebras \mathcal{K} y \mathcal{B} existe una relación que se puede confirmar cuando en la tabla 4.1 las filas y columnas que tienen la variable u se suprimen convirtiéndose la tabla de Kleene en una Boole [184].

\vee	0	u	1	\wedge	0	u	1	$-$	
0	0	u	1	0	0	0	0	0	1
u	u	u	1	u	0	u	u	u	u
1	1	1	1	1	0	u	1	1	0

Tabla 4.1: Tabla de verdad de la disyunción, conjunción y complemento para el álgebra de Kleene \mathcal{K} .

4.3. Conjuntos concretos y conjuntos difusos

Desde el punto de vista de Boole [189], se tiene un valor de verdad para la proposición u es A , y se puede expresar como:

$$\nu(A) = \begin{cases} 1 & \text{si } u \in A \\ 0 & \text{si } u \notin A \end{cases} \quad (4.15)$$

Posteriormente, Zadeh propone otra forma de representación de la pertenencia de un elemento a un conjunto que no necesariamente es valorada con 1 o 0 ya que puede existir una pertenencia parcial que podría ser expresada con un valor entre los límites definidos por Boole [184].

Considerando lo expuesto por Boole, para todo elemento de un conjunto universal U en el cual se ha definido el conjunto A se cumple que: $u \in A$ ó que $u \notin A$, la función de pertenencia del conjunto A está definida como $\mu_A(u)$ con los siguientes valores:

$$\mu_A(u) = \begin{cases} 1 & \text{si } u \in A \\ 0 & \text{si } u \notin A \end{cases} \quad (4.16)$$

Como es de apreciar, existe una equivalencia entre el valor de verdad definido por Boole y la función de pertenencia, de tal manera que:

$$\mu_A(u) = \nu(A) \quad (4.17)$$

4.4. Conjunto difuso

De acuerdo con [189], un conjunto concreto (booleano) se define como una agrupación de elementos de un tipo, donde la pertenencia de un elemento depende del cumplimiento de una determinada condición. En este caso, el elemento pertenece o no pertenece al conjunto, por lo tanto, sus fronteras están claramente delimitadas. En contraste, la lógica difusa permite que un elemento pueda pertenecer al conjunto en cierto grado.

Un conjunto difuso $A \in U$ puede ser expresado como un conjunto de parejas ordenadas de un elemento $x \in U$ y su valor de pertenencia para A , de la siguiente forma:

$$A = \int_U \frac{\mu_A(x)}{x} \quad (4.18)$$

El símbolo de integración representa la colección de todos los puntos $x \in U$, y el símbolo de división indica la correspondencia entre el elemento $x \in U$ y su pertenencia $\mu_A(x)$. Cuando U es discreto, se representa de la forma:

$$A = \sum_U \frac{\mu_A(x)}{x} \quad (4.19)$$

Para un contexto lingüístico, teniendo un universo U_i donde se define un valor lingüístico A_{ij} dado para la variable u_i , la función de pertenencia $\mu(u_i)$ asociada a A_{ij} asigna a cada elemento de U_i un valor entre 0 y 1, describiendo la “certeza” que el elemento $u_i \in U_i$ pueda ser descrito lingüísticamente con A_{ij} , donde i indica el i -ésimo universo de discurso y j el j -ésimo valor lingüístico [184].

Como es de apreciar, un conjunto difuso A en un universo (dominio) X se encuentra definido por la función de pertenencia $\mu_A(x)$, la cual corresponde a un mapeo desde el universo X al intervalo unitario de la forma:

$$\mu_A(x) : X \rightarrow [0, 1] \quad (4.20)$$

Los conjuntos difusos permiten una pertenencia parcial de un elemento en un conjunto. Si el valor de la función de pertenencia es uno, el elemento x pertenece completamente al conjunto difuso, mientras que, si es cero, x no pertenece al conjunto. Si el grado de pertenencia está entre 0 y 1, x es un miembro parcial del conjunto difuso.

En un conjunto discreto de elementos $X = \{x_i \mid i = 1, 2, \dots, n\}$ un conjunto difuso A puede ser definido por una lista de pares ordenados:

$$A = \left\{ \frac{\mu_A(x_1)}{x_1}, \frac{\mu_A(x_2)}{x_2}, \dots, \frac{\mu_A(x_n)}{x_n} \right\} \quad (4.21)$$

También se puede tener una representación en forma de dos vectores relacionados:

$$\begin{aligned} x &= [x_1, x_2, \dots, x_n]^T \\ \mu_A(x) &= [\mu_A(x_1), \mu_A(x_2), \dots, \mu_A(x_n)]^T \end{aligned} \quad (4.22)$$

Para dominios continuos, los conjuntos difusos se pueden definir analíticamente por funciones de pertenencia. Estos conjuntos tiene una aplicación directa en los sistemas de inferencia difusa donde las relaciones entre variables se pueden describir mediante reglas de la forma:

Si proposición del antecedente *Entonces* proposición del consecuente.

4.5. Funciones de pertenencia

De forma práctica, las funciones de pertenencia permiten describir los conceptos que se quieren modelar con un sistema de inferencia difusa. Estos conceptos pueden ser valores negativos o positivos de una variable como también cierto rango de estos. Algunas de las funciones más empleadas son:

- Trapezoidal.
- Triangular.
- Gaussiana.
- Campana beta.
- Forma s .
- Forma z .

Las funciones de pertenencia trapezoidal, triangular, gaussiana y campana beta permiten representar conceptos relacionados con rangos de valores. Por su parte, una función con forma s se puede asociar a valores negativos y una con forma z para positivos.

Función de pertenencia trapezoidal

La función de pertenencia trapezoidal se puede describir mediante los parámetros a , b , c y d que corresponden a los valores donde se tienen los vértices del trapecio que describe esta función de pertenencia. La ecuación (4.23) presenta los parámetros para una función de pertenencia trapezoidal.

En la figura 4.1 se puede apreciar una representación de esta función de pertenencia donde es posible observar la disposición de los parámetros a , b , c y d .

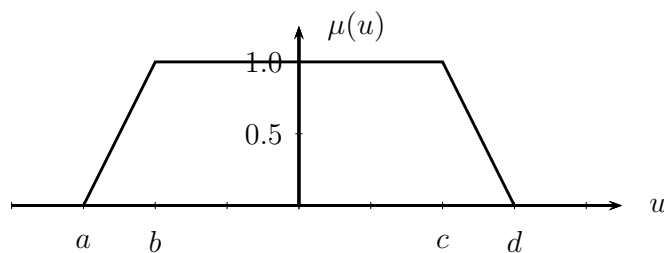


Figura 4.1: Función de pertenencia de tipo trapezoidal.

$$\mu(u) = \begin{cases} 0 & u \leq a \\ \frac{u-a}{b-a} & a < u \leq b \\ 1 & b < u \leq c \\ \frac{d-u}{d-c} & c < u \leq d \\ 0 & u > d \end{cases} \quad (4.23)$$

Función de pertenencia triangular

Esta es una función de pertenencia definida mediante los parámetros a , b y c , donde a y b indican la localización de los vértices inferiores y c el vértice superior. La expresión para esta función de pertenencia es:

$$\mu(u) = \begin{cases} 0 & u \leq a \\ \frac{u-a}{b-a} & a < u \leq b \\ \frac{c-u}{c-b} & b < u \leq c \\ 0 & u > c \end{cases} \quad (4.24)$$

En la figura 4.2 se puede apreciar un ejemplo de la función de pertenencia de tipo triangular.

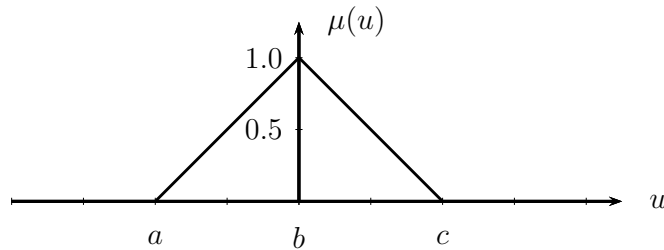


Figura 4.2: Función de pertenencia de tipo triangular.

Función de pertenencia gaussiana

Los parámetros de esta función de pertenencia son c correspondiente al punto central y σ la desviación estándar. La expresión para esta función de pertenencia está dada por la ecuación (4.25). Un ejemplo de la función de pertenencia gaussiana se puede apreciar en la figura 4.3.

$$\mu(u) = \exp\left(-\frac{1}{2}\left(\frac{u-c}{\sigma}\right)^2\right) \quad (4.25)$$

Función de pertenencia campana beta

Esta función de pertenencia es similar a la gaussiana teniendo como parámetros c el punto central y β asociada a la apertura de la campana. Su representación en función de estos parámetros es la siguiente:

$$\mu(u) = \frac{1}{1 + \left(\frac{u-c}{\beta}\right)^2} \quad (4.26)$$

Función de pertenencia con forma s

La función de pertenencia con forma s se compone de los parámetros c y β , donde c está asociado al punto central y β con la pendiente de caída, tal como se observa en la ecuación (4.27).

$$\mu(u) = \frac{1}{2} \left(\frac{e^{\beta(u-c)} - 1}{e^{\beta(u-c)} + 1} + 1 \right) \quad (4.27)$$

Función de pertenencia con forma z

En este caso, los parámetros de la función con forma z son c punto central y β asociada a la pendiente de subida como se muestra en la ecuación (4.28). Un ejemplo de las funciones de pertenencia con forma s y z se presenta en la figura 4.3.

$$\mu(u) = \frac{1}{2} \left(\frac{e^{-\beta(u-c)} - 1}{e^{-\beta(u-c)} + 1} + 1 \right) \quad (4.28)$$

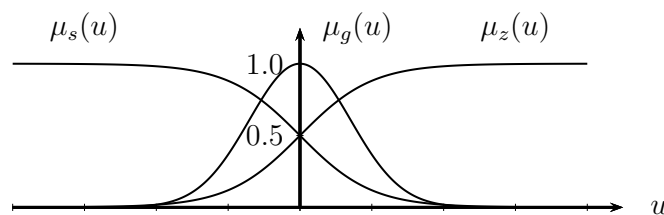


Figura 4.3: Funciones de pertenencia con forma s (μ_s), gaussiana (μ_g) y forma z (μ_z).

4.6. Propiedades de los conjuntos difusos

Considerando un universo de discurso U siendo x cualquier valor de este posible universo de discurso, es decir que $x \in U$, adicionalmente teniendo un conjunto difuso A con la respectiva función de pertenencia $\mu_A(x)$, entonces se pueden establecer las siguientes propiedades de los conjuntos difusos.

Subconjunto difuso

En lógica bivaluada, se dice que el conjunto A es subconjunto de B ($A \subseteq B$) cuando todos los elementos de A se encuentran en B . En lógica difusa, dados los conjuntos difusos A y B definidos en el universo U , con funciones de pertenencia $\mu_A(x)$ y $\mu_B(x)$ respectivamente, entonces se dice que A es un subconjunto difuso de B ($A \subseteq B$) si $\mu_A(x) \leq \mu_B(x)$ para todo $x \in U$.

Altura de un conjunto difuso

La altura de un conjunto difuso A corresponde el valor máximo (supremo) de los grados de pertenencia de los elementos de A .

$$\text{hgt}(A) = \sup\{\mu_A(x)\}, x \in U \quad (4.29)$$

Centro

El centro (*core* o *kernel*) de un conjunto difuso A corresponde al subconjunto abrupto (*crisp* o booleano) en el cual todos sus elementos tienen grado de pertenencia iguales a uno [190].

$$\text{ker}(A) = \{x \in U \mid \mu_A(x) = 1\} \quad (4.30)$$

Soporte

El soporte (*support*) de un conjunto difuso A en el universo de discurso U es un conjunto concreto (*crisp* o booleano) que contiene todos los elementos de U cuyo valor en la función de pertenencia de A es diferente de cero [184], es decir:

$$\text{supp}(A) = \{x \in U \mid \mu_A(x) > 0\} \quad (4.31)$$

Corte alfa de un conjunto difuso

De acuerdo con [191], considerando un valor $\alpha \in [0, 1]$, el α -corte (corte alfa) de un conjunto difuso A consiste en los elementos de U con un valor de pertenencia μ_A igual o mayor que α de tal forma que se tiene:

$$\text{alfa}(A) = \{x \in U \mid \mu_A(x) \geq \alpha\} \quad (4.32)$$

Como es de apreciar, el corte alfa de un conjunto difuso es un conjunto *crisp* o booleano. Por su parte, el centro (*ker*) y soporte (*supp*) de un conjunto difuso pueden ser definidos a partir del corte alfa de la siguiente forma:

$$\text{ker}(A) = \text{alfa}(A), \alpha = 1 \quad (4.33)$$

$$\text{supp}(A) = \text{alfa}(A), \alpha = 0 \quad (4.34)$$

En la figura 4.4 se pueden observar las relaciones entre las propiedades de centro, soporte y α -corte, para un conjunto difuso trapezoidal.

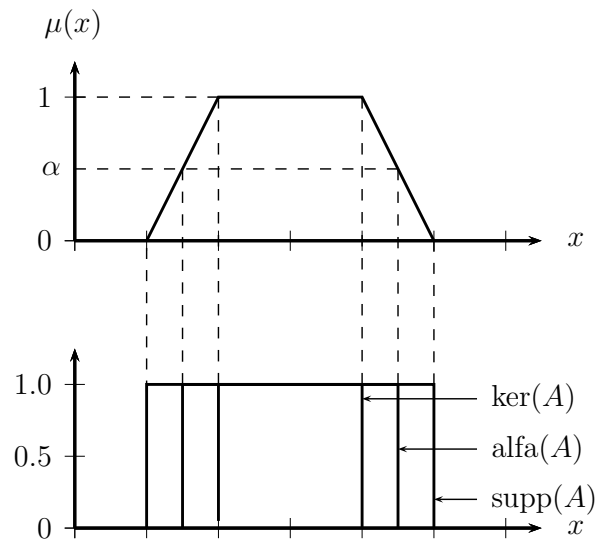


Figura 4.4: Centro, soporte y α -corte.

Conjunto difuso convexo

Un conjunto convexo debe cumplir con la condición dada en la ecuación (4.35), donde $\lambda \in [0, 1]$, [184].

$$\mu_A(\lambda x_1 + (1 - \lambda)x_2) \geq \min[\mu_A(x_1), \mu_A(x_2)] \quad (4.35)$$

En la figura 4.5 se presenta gráficamente la forma para establecer si un conjunto difuso es convexo considerando un determinado α -corte denotado como A_α .

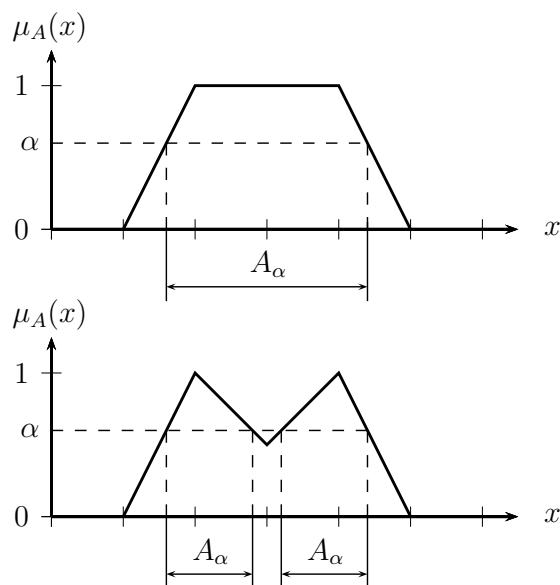


Figura 4.5: Ejemplo de conjuntos difusos convexos y no convexos.

4.7. Operaciones sobre conjuntos difusos

En la teoría del control difuso, se requieren formas útiles para efectuar operaciones entre conjuntos difusos. Los operadores para combinar conjuntos difusos permiten generalizar los métodos comunes empleados con conjuntos booleanos.

En [11], se presenta que existe una equivalencia entre los operadores de conjuntos de unión, intersección y complemento, con los operadores lógicos de *AND*, *OR* y *NOT*, como también con los operadores algebraicos de disyunción, conjunción y complemento.

Aunque el mínimo, máximo y complemento estándar son las operaciones más comunes que se utilizan para *AND*, *OR* y *NOT*, existen otras posibilidades para los conjuntos difusos, las cuales corresponden a t-norma, t-conorma (s-norma según [192]) y complemento difuso [11]. A continuación se describen estos operadores.

Norma triangular

En sistemas de lógica difusa esta operación se denomina t-norma. De manera general se puede representar como:

$$\mathbf{T}[\mu_A(u), \mu_B(u)] = \mu_A(u) \otimes \mu_B(u) \quad (4.36)$$

Una t-norma o norma triangular es una operación binaria $T : [0, 1] \times [0, 1] \rightarrow [0, 1]$ utilizada en lógica multivaluada (lógica difusa). La t-norma es una forma general de la operación de conjunción y es simétrica, asociativa y con 1 como elemento neutral [193, 194]. La t-norma cumple con las siguientes propiedades:

- Conmutativa: $\mu_A \otimes \mu_B = \mu_B \otimes \mu_A, \forall \mu_A, \mu_B \in [0, 1]$.
- Asociativa: $\mu_A \otimes (\mu_B \otimes \mu_C) = (\mu_A \otimes \mu_B) \otimes \mu_C, \forall \mu_A, \mu_B, \mu_C \in [0, 1]$.
- Si $\mu_A \leq \mu_B$ y $\mu_C \leq \mu_D$, entonces, $\mu_A \otimes \mu_C \leq \mu_B \otimes \mu_D$
- Condiciones de frontera:
 - $\mu_A \otimes 0 = 0, \mu_A \in [0, 1]$.
 - $\mu_A \otimes 1 = \mu_A, \mu_A \in [0, 1]$.

Algunas t-normas son: mínimo, producto, Lukasiewicz, drástica, Frank, Yager, Schweizer-Sklar, Hamacher y Sugeno-Weber [193].

Mínimo

Considerando dos conjuntos difusos A y B con funciones de pertenencia $\mu_A(u)$ y $\mu_B(u)$ definidas en el universo u , se tiene:

$$\mu_A(u) \otimes \mu_B(u) = \min\{\mu_A(u), \mu_B(u)\} \quad (4.37)$$

Producto algebraico

En este caso, la t-norma correspondiente al producto algebraico se calcula como:

$$\mu_A(u) \otimes \mu_B(u) = \mu_A(u)\mu_B(u) \quad (4.38)$$

T-norma clase Yager

Este tipo de t-norma se encuentra definida por:

$$\begin{aligned} T_w\{\mu_A(u), \mu_B(u)\} &= \mu_A(u) \otimes \mu_B(u) \\ &= 1 - \min\left\{1, [(1 - \mu_A(u))^w + (1 - \mu_B(u))^w]^{\frac{1}{w}}\right\} \end{aligned} \quad (4.39)$$

Donde el parámetro $w \in (0, \infty)$ permite una variación en la forma de la función de pertenencia resultante.

Conorma triangular

Esta operación es conocida en lógica difusa como t-conorma o también s-norma (según [192]). Una manera general para representar la operación difusa de t-conorma es:

$$\mathbf{S}[\mu_A(u), \mu_B(u)] = \mu_A(u) \oplus \mu_B(u) \quad (4.40)$$

Una función t-conorma $S : [0, 1] \times [0, 1] \rightarrow [0, 1]$, es simétrica, asociativa y neutral con el elemento 0 [193, 194]. La t-conorma cumple con las siguientes propiedades:

- Conmutativa: $\mu_A \oplus \mu_B = \mu_B \oplus \mu_A, \forall \mu_A, \mu_B \in [0, 1]$.
- Asociativa: $\mu_A \oplus (\mu_B \oplus \mu_C) = (\mu_A \oplus \mu_B) \oplus \mu_C, \forall \mu_A, \mu_B, \mu_C \in [0, 1]$.
- Si $\mu_A \leq \mu_B$ y $\mu_C \leq \mu_D$, entonces, $\mu_A \oplus \mu_C \leq \mu_B \oplus \mu_D$
- Condiciones de frontera:
 - $\mu_A \oplus 0 = \mu_A, \mu_A \in [0, 1]$.
 - $\mu_A \oplus 1 = 1, \mu_A \in [0, 1]$.

Algunas t-conormas son: máximo, suma algebraica, Lukasiewicz, drástica, Frank, Yager, Schweizer-Sklar, Hamacher y Sugeno-Weber [193].

Máximo

Considerando A y B dos conjuntos difusos con funciones de pertenencia $\mu_A(u)$ y $\mu_B(u)$ definidas en el universo u , entonces, la t-conorma del máximo se representa como:

$$\mu_A(u) \oplus \mu_B(u) = \max\{\mu_A(u), \mu_B(u)\} \quad (4.41)$$

Suma algebraica

En este caso la t-conorma se puede establecer como:

$$\mu_A(u) \oplus \mu_B(u) = \mu_A(u) + \mu_B(u) - \mu_A(u)\mu_B(u) \quad (4.42)$$

T-conorma clase Yager

Para esta t-conorma el resultado se calcula de la forma:

$$\begin{aligned} S_w\{\mu_A(u), \mu_B(u)\} &= \mu_A(u) \oplus \mu_B(u) \\ &= \min \left\{ 1, [\mu_A(u)^w + \mu_B(u)^w]^{\frac{1}{w}} \right\} \end{aligned} \quad (4.43)$$

Mediante el parámetro $w \in (0, \infty)$ se logra una variación en la forma de la función de pertenencia resultante.

Complemento difuso

Considerando un conjunto difuso A con una función de pertenencia $\mu_A(u)$, la representación general del complemento de un conjunto difuso es:

$$C\{\mu_A(u)\} = \mu_{\bar{A}}(u) \quad (4.44)$$

De forma estándar, el complemento difuso se puede calcular mediante la siguiente expresión:

$$C\{\mu_A(u)\} = 1 - \mu_A(u) \quad (4.45)$$

Otra forma para realizar la operación de complemento consiste en la denominada Sugeno definida como C_λ , la cual utiliza un parámetro λ donde $-1 < \lambda < \infty$ permitiendo variar la forma del conjunto resultante [192]. Este tipo de complemento se puede calcular como:

$$C_\lambda\{\mu_A(u)\} = \frac{1 - \mu_A(u)}{1 + \lambda\mu_A(u)} \quad (4.46)$$

Un complemento adicional corresponde al denominado Yager donde se emplea el parámetro $w \in (0, \infty)$. La expresión para el cálculo de este complemento es:

$$C_w\{\mu_A(u)\} = [1 - \mu_A(u)^w]^{\frac{1}{w}} \quad (4.47)$$

4.8. Relaciones difusas

Con la finalidad de representar el conocimiento mediante conjuntos difusos, es necesario tener presente las relaciones que se pueden generar entre estos de forma lingüística. Una relación difusa n -aria es un mapeo del tipo:

$$R : X_1 \times X_2 \times \dots \times X_n \rightarrow [0, 1] \quad (4.48)$$

la cual asigna grados de pertenencia a todas las n -tuplas (x_1, x_2, \dots, x_n) del producto cartesiano $X_1 \times X_2 \times \dots \times X_n$.

De esta manera, una relación difusa corresponde a un conjunto difuso definido en el producto cartesiano $X_1 \times X_2 \times \dots \times X_n$. Los valores de pertenencia representan el grado de asociación (correlación) entre los elementos de los diferentes dominios X_i [11, 190].

Sobre este tipo de relaciones, son de particular interés aquellas que pueden ser descritas mediante reglas difusas de la forma:

Si proposición (antecedente) **Entonces** proposición (consecuente)

4.8.1. Implicaciones difusas

En los sistemas de lógica difusa resulta de gran importancia la forma como se representan e interpretan las reglas de la forma *Si...Entonces*. En el cálculo proposicional de dos elementos, la expresión *Si p Entonces q* se representa como $p \rightarrow q$ (p implica q), la cual corresponde a $\bar{p} \vee q$, donde $(\bar{})$ y (\vee) representan las operaciones de *NOT* y *AND* [192].

Considerando lo anterior, las reglas difusas de la forma *Si...Entonces*, al ser interpretadas como $\bar{p} \vee q$ se pueden calcular mediante las operaciones difusas de complemento, t-norma y t-conorma. Sin embargo, en el campo de la lógica difusa existen diferentes alternativas de interpretar las reglas difusas (tipos de implicación), como son: Dienes-Rescher, Lukasiewikz, Zadeh, Gödel y Mamdani.

Implicación Dienes-Rescher

Este tipo de implicación emplea la forma $\bar{p} \vee q$ realizando la implementación con los operadores difusos estándar. De esta manera se interpreta *Si A Entonces B* como una relación difusa $Q \in X \times Y$ con una función de pertenencia dada por:

$$\mu_Q(x, y) = \text{máx}\{1 - \mu_A(x), \mu_B(y)\} \quad (4.49)$$

Implicación Lukasiewikz

Para este tipo de implicación se emplea la t-conorma tipo Yager tomando $w = 1$ y el complemento difuso tipo estándar. La relación difusa *Si A Entonces B* corresponde a una relación $Q \in X \times Y$ con una función de pertenencia que se puede establecer como:

$$\mu_Q(x, y) = S_w\{1 - \mu_A(x), \mu_B(y)\} \quad (4.50)$$

Implicación Zadeh

La interpretación de la relación *Si A Entonces B* con la implicación de Zadeh corresponde a la relación difusa $Q \in X \times Y$ con una función de pertenencia calculada como:

$$\mu_Q(x, y) = \max\{\min(\mu_A(x), \mu_B(y)), 1 - \mu_A(x)\} \quad (4.51)$$

Implicación Gödel

Para la implicación de Gödel se tiene la interpretación de la regla difusa *Si A Entonces B* con la relación difusa $Q \in X \times Y$, teniendo la función de pertenencia:

$$\mu_Q(x, y) = \begin{cases} 1 & \text{si } \mu_A(x) \leq \mu_B(y), \\ \mu_B(y) & \text{para otro caso.} \end{cases} \quad (4.52)$$

Implicación Mamdani

Esta implicación es muy utilizada en lógica difusa porque considera que las reglas difusas están descritas de forma local donde cada regla tiene un rango de acción pequeño en comparación con el universo de discurso [192].

La regla difusa *Si A Entonces B* se interpreta como una relación $Q \in X \times Y$, donde la función de pertenencia se puede calcular como:

$$\mu_Q(x, y) = \min\{\mu_A(x), \mu_B(y)\} \quad (4.53)$$

Capítulo 5

Sistemas de lógica difusa

5.1. Introducción

Un sistema de inferencia difusa permite procesar información utilizando un mecanismo que relaciona la entrada y salida de forma lingüística mediante conjuntos y operadores difusos, de esta forma, para un valor concreto de entrada se puede establecer un valor de salida también concreto. Para este proceso existen diferentes modelos siendo los más conocidos:

- Sistemas difusos tipo Mamdani.
- Sistemas difusos tipo Takagi-Sugeno.

En el presente capítulo se revisan conceptos sobre estos tipos de sistemas de inferencia difusa como también los procesos de cálculo asociados.

5.2. Modelos para la implementación de sistemas de lógica difusa

Los modelos que se tienen para la implementación de los sistemas de lógica difusa principalmente dependen de las características para representar la información [32, 190]. Algunos de estos enfoques son:

1. **Modelo Lingüístico:** Este se encuentra basado en la descripción del conocimiento mediante una colección de reglas de tipo *Si...Entonces*, donde tanto los antecedentes como los consecuentes son términos lingüísticos. Un ejemplo es el modelo tipo Mamdani.
2. **Modelo Relacional:** Corresponde a una generalización del modelo lingüístico donde se codifican asociaciones entre términos lingüísticos definidos en las entradas y el dominio de las salidas empleando relaciones difusas, de tal forma que un antecedente se encuentre asociado a diferentes consecuentes.

3. **Modelo Singleton:** Resulta ser un caso especial del modelo lingüístico, donde el consecuente es un conjunto de tipo *singleton*. Este conjunto puede ser representado como valores numéricos.
4. **Modelo Takagi-Sugeno:** Emplea reglas con antecedentes difusos y consecuentes funcionales. Este modelo integra la habilidad de los modelos lingüísticos para representar el conocimiento cualitativo, así como también un efectivo potencial para expresar información cuantitativa. Este modelo permite la implementación de técnicas de identificación a partir de datos.

5.3. Elementos de un sistema de inferencia difusa

La arquitectura general de un sistema de inferencia difusa se encuentra compuesto de los siguientes elementos:

- Fusificador.
- Mecanismo de inferencia difusa.
- Base de reglas difusas.
- Defusificador.

En los sistemas difusos Mamdani y Takagi-Sugeno, los procesos de fusificación e inferencia se implementan de la misma forma. Su principal diferencia se presenta en la forma que se describe la base de reglas y como se efectúa el proceso asociado a la defusificación puesto que en los sistemas tipo Takagi-Sugeno este se considera integrado al proceso de agregación [192]. En un sistema de lógica difusa, la entrada normalmente es un valor concreto que para ser procesado por el sistema difuso requiere llevarse a una representación interpretable por el proceso de inferencia. El funcionamiento de un sistema de inferencia difusa comprende los siguientes pasos:

- Fusificación de las entradas: En este paso se toman los valores concretos de las entradas y se establece su respectivo nivel de pertenencia.
- Aplicación de los operadores difusos: Con los valores calculados, para cada regla se realizan las respectivas operaciones difusas entre las diferentes entradas.
- Implicación: En este paso se aplica la respectiva relación difusa entre el antecedente y consecuente para cada regla.
- Agregación: El proceso de implicación para cada regla proporciona un resultado parcial, por lo cual, para considerar todas las reglas se realiza el proceso de agregación.
- Defusificación: En este proceso se toma la función de pertenencia resultado de la agregación y se obtiene un valor puntual de salida.

5.4. Sistemas tipo Mamdani

Este tipo de sistemas presentan un buen grado de interpretabilidad ya que permiten relacionar la entrada y salida mediante conjuntos difusos los cuales se asocian a conceptos lingüísticos para cada universo de discurso [192]. En la figura 5.1 se muestra la configuración básica de un sistema tipo Mamdani.

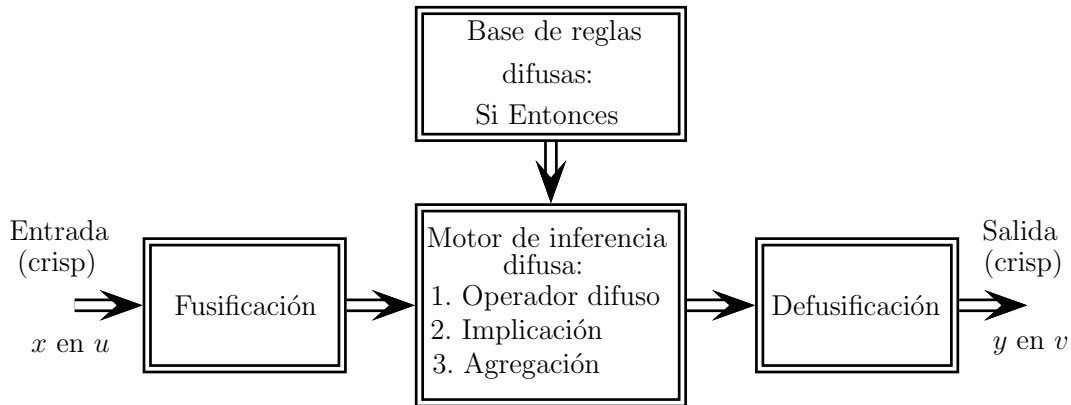


Figura 5.1: Sistemas tipo Mamdani.

En un sistema de inferencia difusa tipo Mamdani se observan las siguientes partes:

- Fusificación.
- Motor de inferencia difusa.
- Defusificación.

5.4.1. Base de reglas difusas

Mediante la base de reglas el sistema difuso almacena el conocimiento lingüístico, el cual constituye el comportamiento del sistema a modelar. Con esta base de reglas, para un determinado valor de entrada se calcula la respectiva salida [192].

La base de reglas por lo general es de la forma *Si...Entonces*, presentando dos partes, el antecedente y el consecuente de la forma:

Si la entrada es x ***Entonces*** la salida es y

Es de señalar que en los sistemas difusos tipo Mamdani tanto en el antecedente como en el consecuente las reglas se componen por expresiones lingüísticas representadas mediante funciones de pertenencia [192].

5.4.2. Defusificación

Del proceso de agregación se obtiene un conjunto difuso, por lo cual, para establecer un valor puntual de la salida se realiza el proceso de defusificación. Para esto existen varios métodos como el centroide y promedio de centros, entre otros. En la figura 5.2 se puede apreciar el funcionamiento de un sistema de inferencia difusa tipo Mamdani.

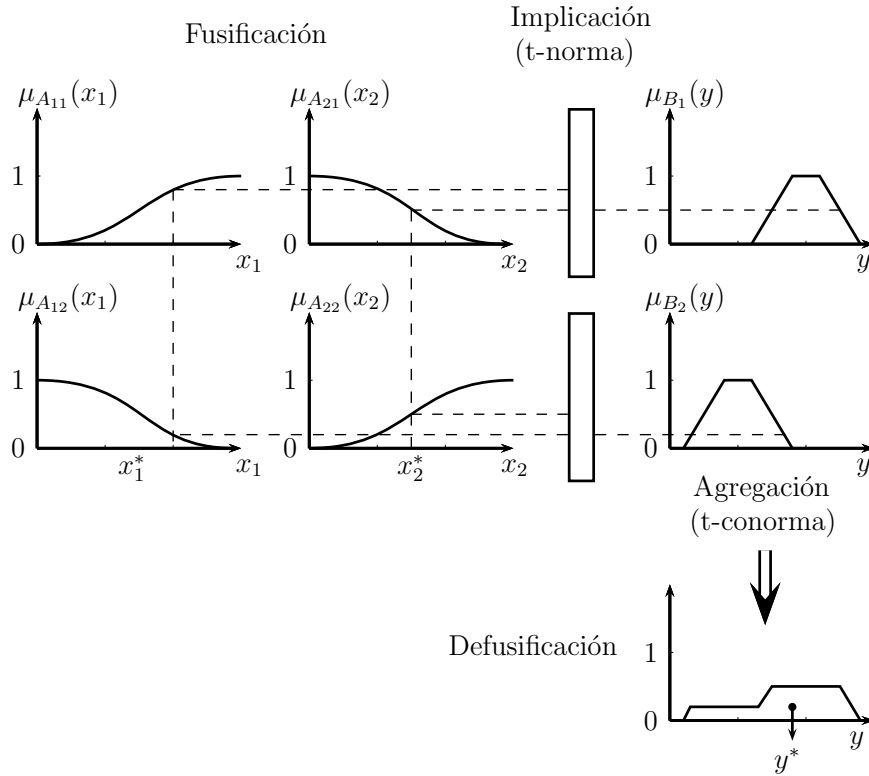


Figura 5.2: Funcionamiento de un sistema de inferencia difusa tipo Mamdani.

En la figura 5.2 se pueden apreciar los siguientes pasos:

1. Un valor ingresa al sistema difuso y es convertido en el respectivo nivel de pertenencia para cada regla.
2. Para cada regla se aplican los respectivos operadores difusos con los valores anteriormente calculados.
3. Empleando el anterior resultado y con la respectiva función de pertenencia de la salida asociada a la respectiva regla, se aplica el operador difuso de implicación.
4. Mediante un operador difuso se procede a realizar la agregación todas las implicaciones.
5. Se obtiene el valor puntual de salida mediante el proceso de defusificación.

5.5. Sistemas tipo Takagi-Sugeno

Este tipo de sistemas presenta un grado de interpretabilidad lingüística menor que los sistemas Mamdani ya que para la salida, en lugar de conjuntos difusos asociados a conceptos lingüísticos, se realiza la ponderación de una función que depende de las entradas, sin embargo, esta misma característica permite su representación como una red neuronal y por lo tanto realizar el ajuste de sus parámetros mediante un esquema supervisado [192].

Los sistemas tipo Takagi-Sugeno son también denominados TS [195]. El esquema general del proceso de inferencia de un sistema tipo Sugeno se puede apreciar en la figura 5.3.

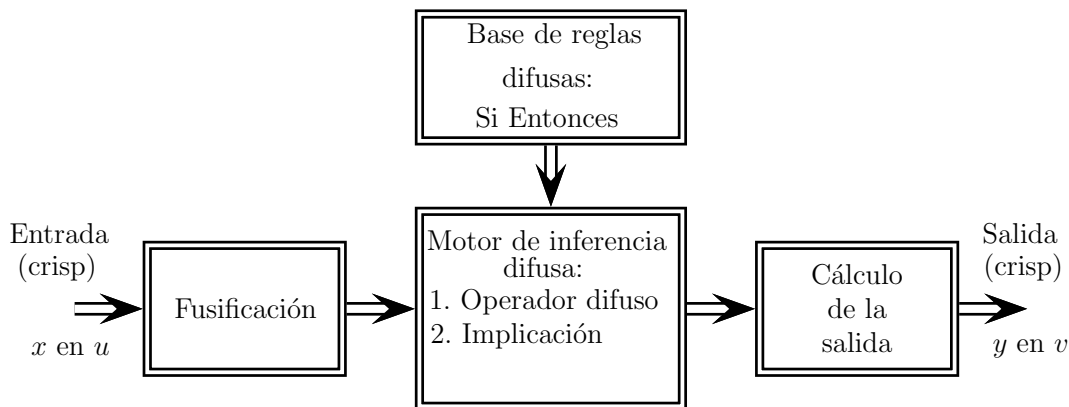


Figura 5.3: Sistema difuso tipo Sugeno.

En los sistemas difusos Mamdani y Sugeno la parte de fusificación y de relaciones difusas del antecedente son similares, la principal diferencia se aprecia en la agregación y el cálculo de la salida.

5.5.1. Base de reglas difusas

En la base de reglas de un sistema tipo Sugeno el consecuente se encuentra asociado con una función que depende de las variables de entrada de la forma:

$$\textit{Si} \text{ la entrada es } x \textit{ Entonces la salida } y = f(x)$$

En los sistemas difusos tipo Sugeno cada regla proporciona un valor concreto, por lo cual se suele considerar que el proceso asociado a la defusificación no existe, sin embargo, es necesario establecer un valor concreto considerando el resultado de todas las reglas.

5.5.2. Cálculo de la salida

Con el fin de establecer la salida del sistema difuso se realiza una suma ponderando cada función con el respectivo valor obtenido de la relación difusa de la respectiva regla.

De esta forma la salida se puede calcular como:

$$y = \frac{w_1 y_1 + w_2 y_2 + \dots + w_n y_n}{w_1 + w_2 + \dots + w_n} \quad (5.1)$$

esto se puede escribir de la forma:

$$y = \bar{w}_1 y_1 + \bar{w}_2 y_2 + \dots + \bar{w}_n y_n \quad (5.2)$$

donde:

$$\bar{w}_i = \frac{w_i}{w_1 + w_2 + \dots + w_n} \quad (5.3)$$

Cada regla tiene asociada una función que depende de las variables de entrada de la forma:

$$\begin{aligned} y_1 &= f_1(x_1, x_2, \dots, x_m) \\ y_2 &= f_2(x_1, x_2, \dots, x_m) \\ &\vdots \\ y_n &= f_n(x_1, x_2, \dots, x_m) \end{aligned}$$

Considerando dos entradas y dos reglas para el proceso de inferencia, en la figura 5.4 se muestra gráficamente el funcionamiento de un sistema difuso Takagi-Sugeno.

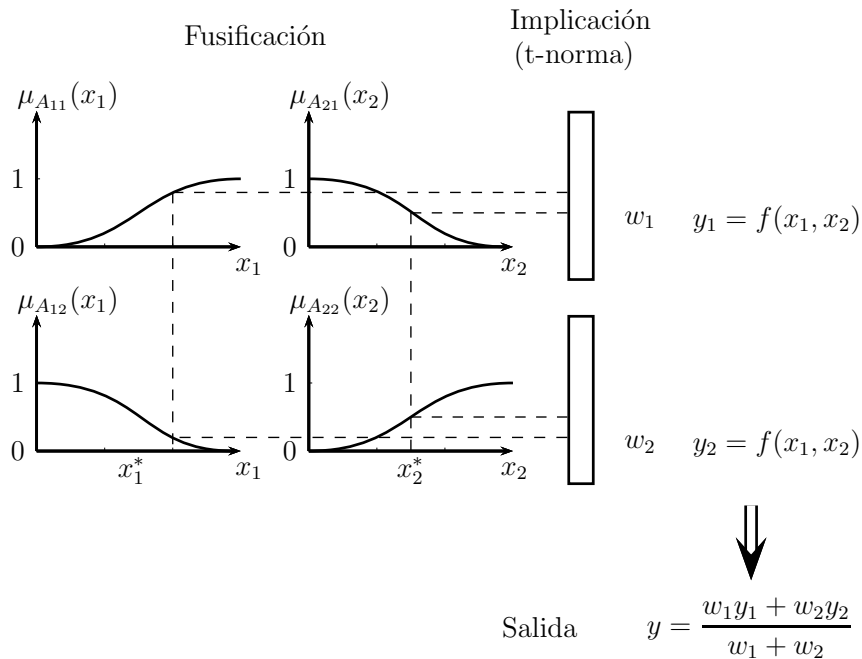


Figura 5.4: Funcionamiento de un sistema de inferencia difusa tipo Sugeno.

Como es de apreciar, en la figura 5.4 se presentan los siguientes pasos:

1. Se ingresa un valor puntual al sistema difuso el cual es convertido en un nivel de pertenencia para cada regla.

2. Para cada regla se aplican los respectivos operadores difusos teniendo como resultado w_1 y w_2 .
3. La salida se calcula mediante la suma ponderada $y = \bar{w}_1 y_1 + \bar{w}_2 y_2$.

5.5.3. Sistemas difusos Takagi-Sugeno no normalizados

Para este tipo de sistemas no se efectúa la normalización (división) de los resultados obtenidos de cada regla, por lo cual, la salida del sistema se calcula como:

$$y = w_1 y_1 + w_2 y_2 + \cdots + w_n y_n \quad (5.4)$$

Escrito de forma compacta se tiene:

$$y = \sum_{i=1}^n w_i y_i \quad (5.5)$$

Capítulo 6

Sistemas de control mediante redes neuronales

6.1. Introducción

Este capítulo muestra los fundamentos relacionados con redes neuronales, revisando el concepto desde la perspectiva biológica; posteriormente se describe un modelo computacional y su generalización, como también los principios involucrados en el proceso de entrenamiento. Posteriormente se revisan las principales estrategias de control que emplean redes neuronales en su estructura.

6.2. Redes neuronales

Las redes neuronales artificiales (*Artificial Neuronal Networks* ANN) son modelos computacionales basados en el comportamiento del cerebro humano. Según [8, 9, 196], teniendo presente que la naturaleza resuelve problemas de supervivencia por medio de procesamiento autoorganizado que emerge de la interacción de numerosos procesadores elementales, las redes neuronales artificiales se fundamentan en un sistema compuesto por múltiples neuronas ampliamente interconectadas formando capas.

Los conceptos que se buscan emular con las redes neuronales artificiales corresponden al paralelismo de cálculo, memoria distribuida y adaptabilidad [8, 9]. La principal característica de una ANN consiste en la habilidad para aprender relaciones funcionales complejas utilizando datos de entrenamiento de entrada y salida observados de un sistema. De esta forma, las redes neuronales pueden ser empleadas como modelos de caja negra para sistemas no lineales, multivariados, estáticos y dinámicos [190].

6.2.1. Modelo biológico de una neurona

Según investigaciones, se estima que el sistema nervioso presenta aproximadamente cien mil millones de neuronas; aunque pueden variar de forma, una posible representación corresponde a la mostrada en la figura 6.1, donde se puede apreciar un cuerpo o “soma”, ramificaciones correspondientes a “dendritas” y una fibra tubular denominada “axón”, la cual se ramifica en su extremo [8, 9].

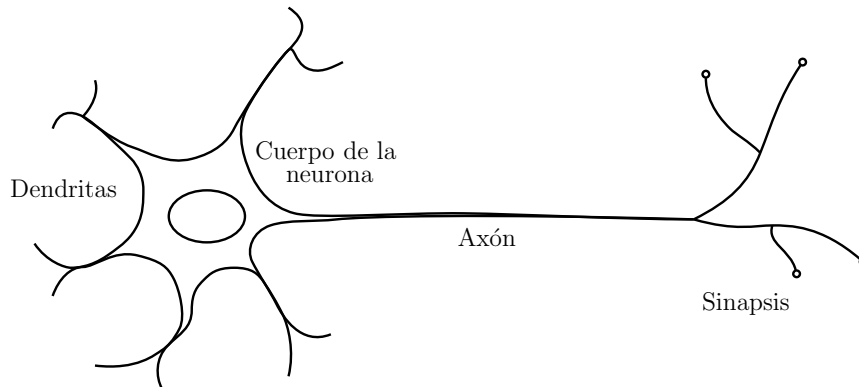


Figura 6.1: Estructura de una neurona biológica típica.

De la figura 6.1 se aprecia un camino de entrada para la información (dendritas), un elemento de cómputo (soma) y un camino de salida (axón), la comunicación entre neuronas se realiza mediante el proceso de sinapsis [8, 9, 197]. Las ramificaciones finales del axón presentan conexiones sinápticas que permiten la comunicación con otras neuronas. La conexión sináptica no está dada por una cantidad fija, esta puede ser modulada en una escala temporal y el aprendizaje está dado por el proceso como se forman las conexiones sinápticas [8, 9].

6.2.2. Perceptrón

El perceptrón corresponde al modelo artificial básico de una neurona el cual se puede apreciar en la figura 6.2. Cada neurona posee un número determinado de entradas multiplicadas por un correspondiente peso, estos productos más un valor constante denominado *bias*, son sumados teniendo así una señal total. Esta señal es procesada mediante una función de activación de tal forma que se obtiene la salida de la neurona [196, 197, 198]. Las funciones de activación comúnmente empleadas son: lineal, escalón, lineal a tramos, sigmoideal y gaussiana, entre otras.

Considerando la representación del perceptrón mostrado en la figura 6.2, en este modelo la salida se puede calcular como:

$$y = f \left(b_1 + \sum_{i=1}^N x_i w_{1i} \right) \quad (6.1)$$

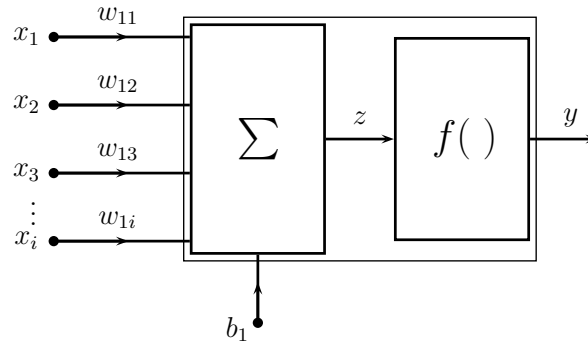


Figura 6.2: Modelo de un perceptrón.

6.2.3. Redes neuronales

Con el fin de ampliar la capacidad de procesamiento, se realiza la interconexión de neuronas, las cuales son usualmente organizadas en capas. Las capas que están entre la entrada y la salida son conocidas como capas ocultas, dependiendo del número de capas se puede tener una red neuronal mono-capa o multi-capa. Considerando el flujo de información en la red, también se presenta la siguiente clasificación:

- Redes unidireccionales: denominadas *feedforward*, donde la información fluye en un único sentido, desde las neuronas de entrada hacia las de salida.
- Redes recurrentes: llamadas *feedback*, o realimentadas, para las cuales la información puede regresar a neuronas de entrada [190].

6.2.4. Entrenamiento y ejecución

Para emplear una red neuronal en una aplicación, por lo general, primero se entrena y luego se ejecuta [8, 9]. El aprendizaje consiste en determinar un conjunto de pesos sinápticos que permitan obtener un valor de salida deseado ante una determinada entrada. Este proceso suele ser iterativo, actualizando los pesos hasta alcanzar el desempeño deseado de la red neuronal [8, 9]. Por lo general, los dos tipos básicos de aprendizaje son el supervisado y el no supervisado o autoorganizado. En el aprendizaje supervisado la red es entrenada con valores de entrada y salida y el ajuste de los parámetros de la red se realiza considerando el error de la salida. En el aprendizaje no supervisado, a la red solamente se le suministran valores de entrada, de tal forma que el ajuste de los pesos se lleva a cabo solamente con los valores de entrada y la salida actual de la red [8, 9, 190]. Desde este punto de vista, las redes neuronales se puede clasificar según su algoritmo de entrenamiento como supervisado, no supervisado, reforzado o híbrido.

En el caso particular de la red neuronal del tipo perceptrón multicapa, se utiliza el algoritmo de propagación hacia atrás (*Back Propagation* BP), denominado de esta forma porque el error entre el valor real y el obtenido por la red es propagado hacia atrás [8, 9, 196, 197, 198]. En este algoritmo se establece una función de error, la cual se deriva en función de todos los parámetros libres de la red empleando para esto la regla de la

cadena. Con el valor de esta derivada se hace el respectivo ajuste de los parámetros considerando la regla de máximo descenso. Para realizar el entrenamiento se emplea el conjunto de datos disponible de tal forma que luego de utilizar todos los datos se ha realizado una época de entrenamiento (*epoch*), convencionalmente se requieren de varias épocas para lograr un buen entrenamiento de la red neuronal [190].

6.3. Sistemas de control mediante redes neuronales

En esta sección se describen algunas de las topologías más empleadas para el control de procesos que emplean en su estructura redes neuronales. Desde un punto de vista general se pueden tener esquemas de control neuronal directos e indirectos, presentando las siguientes características:

- Control directo: El entrenamiento del controlador (red neuronal) se realiza considerando la diferencia entre la señal de control idónea y la señal de control dada por la red neuronal. Considerando que esta señal de control ideal no se conoce, se suelen emplear algoritmos de entrenamiento por refuerzo, los cuales permiten manejar las características cualitativas de la señal de error [199].
- Control indirecto: En este caso es necesario establecer un modelo neuronal de la planta para entrenar el controlador. Dicho modelo se puede emplear para establecer un modelo inverso de la planta, o para el entrenamiento del controlador realizando una propagación del error hacia atrás [199].

Las redes neuronales se han aplicado con mucho éxito en la identificación y control de sistemas dinámicos. Las capacidades de aproximación universales del perceptrón multicapa hacen que sea una opción popular para el modelado de sistemas no lineales para la implementación de controladores de propósito general [200]. De las arquitecturas más populares para el control se tiene:

- Control predictivo basado en modelo de red neuronal: Este controlador utiliza un modelo de red neuronal para predecir la respuesta de la planta para las futuras señales de control. Mediante un algoritmo de optimización se calculan las señales de control que optimizan el rendimiento futuro de la planta. El modelo de la planta mediante una red neuronal se entrena fuera de línea. Mediante un algoritmo de optimización se realiza durante cada tiempo de muestreo el cálculo de la señal de control óptimo. Por lo anterior, este tipo de controlador requiere una cantidad significativa de procesamiento en línea [201].
- Linealización por realimentación NARMA-L2: En este caso, el controlador corresponde a una reordenación del modelo de la planta identificado con redes neuronales, el cual se realiza fuera de línea. El único cálculo en línea es la evaluación del controlador de la red neuronal [201].
- Control mediante modelo de referencia con red neuronal: Este esquema de control requiere la identificación previa de la planta con una red neuronal. Con el modelo

de la planta fuera de línea se entrena el controlador empleando el algoritmo de propagación hacia atrás del error dinámico *Dynamic Back Propagation* (DBP) [202].

Normalmente, cuando se utilizan redes neuronales para control, los pasos a seguir son los de identificación y control. En la etapa de identificación del sistema, se desarrolla un modelo de red neuronal de la planta que se requiere controlar. El modelo de planta se utiliza en la etapa de diseño para establecer el controlador. En cada una de las tres arquitecturas de control consideradas, la etapa de identificación del sistema es similar. La etapa de diseño de control, sin embargo, es diferente para cada arquitectura. El esquema general del proceso de identificación se puede apreciar en la figura 6.3.

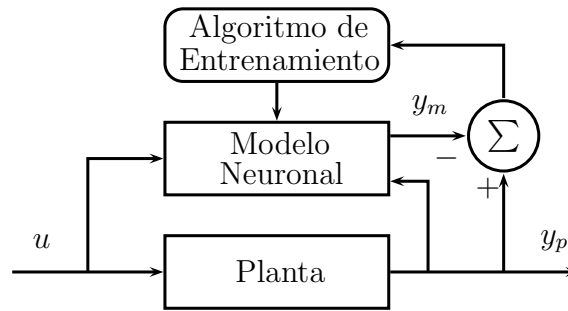


Figura 6.3: Esquema general del proceso de identificación.

Como es de apreciar para el control predictivo, el modelo de la planta se utiliza para predecir el comportamiento futuro del sistema, empleando un algoritmo de optimización para establecer la entrada de control que optimiza el rendimiento futuro. Por su parte, para el control NARMA-L2, el controlador es simplemente un reordenamiento del modelo de la planta. Finalmente para el control con modelo de referencia, el controlador corresponde a una red neuronal entrenada para controlar la planta que ha sido identificada previamente [201].

6.4. Control predictivo

En este caso, el método de control predictivo se encuentra basado en la técnica de horizonte de tiempo [203]. Con el modelo de red neuronal se predice la respuesta de la planta para un determinado horizonte de tiempo. Las predicciones son utilizadas por un algoritmo de optimización numérica para determinar la señal de control que minimiza el siguiente criterio de desempeño:

$$J = \sum_{j=N_1}^{N_2} (y_r[n+j] - y_m[n+j])^2 + \rho \sum_{j=N_1}^{N_u} (u'[n+j-1] - u'[n+j-2])^2 \quad (6.2)$$

Donde N_1 , N_2 , y N_u definen los horizontes sobre los cuales se evalúan el error de seguimiento y los incrementos de control. La variable u' corresponde a la señal de control

provisional, y_r es la respuesta deseada y por su parte y_m es la respuesta del modelo de red. El valor de ρ pondera los valores de la acción de control.

El esquema general de un sistema de control predictivo se puede apreciar en la figura 6.4. El controlador está compuesto por el modelo de la planta (red neuronal) y el bloque de optimización. Este último determina los valores de u' que minimizan J , de tal forma que la u óptima sera la señal que se aplicará en la entrada de la planta.

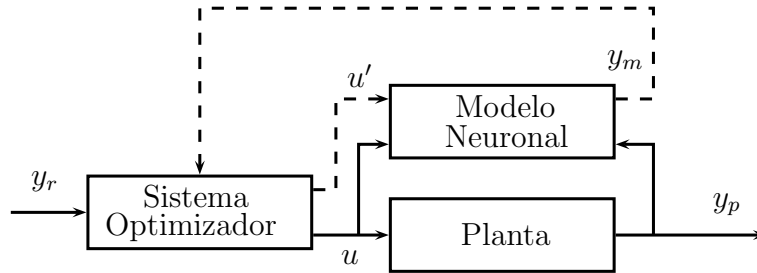


Figura 6.4: Esquema general de un sistema de control predictivo.

6.5. Control NARMA-L2

Este enfoque de control busca linealizar la dinámica de sistemas no lineales mediante la cancelación de estas no linealidades. Considerando lo presentado en [204, 205], el modelo NARMA (*Nonlinear Autoregressive Moving Average*), empleado para la identificación de la planta es:

$$y[n+1] = f(y[n], y[n-1], \dots, y[n-p+1], u[n-1], \dots, u[n-q+1]) + g(y[n], y[n-1], \dots, y[n-p+1], u[n-1], \dots, u[n-q+1])u[n] \quad (6.3)$$

Donde f y g se aproximan empleando redes neuronales tal como se puede apreciar en la figura 6.5.

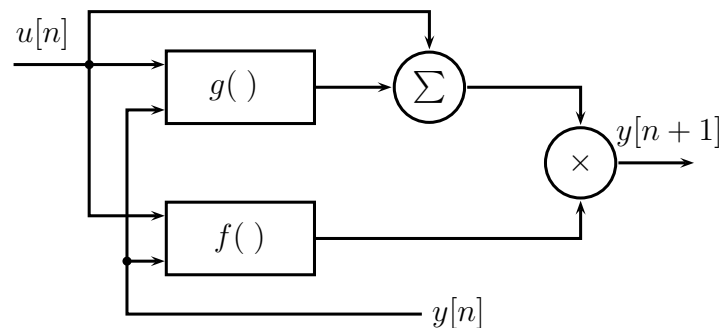


Figura 6.5: Esquema para la identificación empleando el modelo NARMA [204, 205].

Considerando este modelo NARMA-L2 se puede establecer un controlador con la si-

guiente estructura:

$$u[n] = \frac{y_r[n+1] - f(y[n], y[n-1], \dots, y[n-p+1], u[n-1], \dots, u[n-q+1])}{g(y[n], y[n-1], \dots, y[n-p+1], u[n-1], \dots, u[n-q+1])} \quad (6.4)$$

Donde $y_r[n+1]$ es la señal deseada o de referencia a ser seguida. La estructura de este controlador se puede apreciar en la figura 6.6.

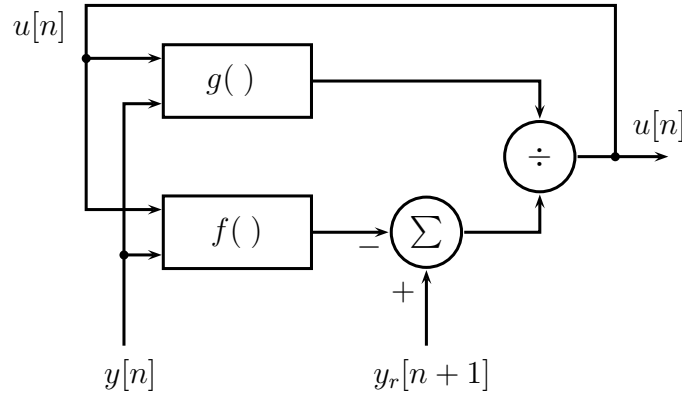


Figura 6.6: Estructura del controlador NARMA-L2.

Como es de apreciar, con este enfoque únicamente se requiere realizar la identificación de la planta puesto que mediante una reorganización de esta identificación se implementa el controlador.

6.6. Controlador neuronal con modelo de referencia

En esta arquitectura se utilizan dos redes neuronales, una para el controlador y otra para el modelo de planta. Primero se efectúa la identificación de la planta y posteriormente se realiza el entrenamiento del controlador para que el sistema siga la referencia deseada. En la figura 6.7 se puede apreciar las dos redes neuronales empleadas [200]. En este esquema, el bloque del modelo de referencia corresponde al comportamiento deseado del sistema en lazo cerrado.

Como es de apreciar, en el modelo de la planta se observa como entrada la señal $u[n]$ y salida $y[n]$, de tal forma que se tiene una estructura:

$$y[n+1] = f_p(y[n], y[n-1], \dots, y[n-p+1], u[n], u[n-1], \dots, u[n-q+1]) \quad (6.5)$$

Por su parte, para el controlador se tienen como entradas la referencia $r[n]$ y la señal medida del proceso $y[n]$, en la salida se presenta la acción de control $u[n]$.

$$u[n+1] = f_c(y[n], \dots, y[n-p+1], r[n], \dots, r[n-m+1], u[n], \dots, u[n-q+1]) \quad (6.6)$$

Donde p corresponde al número de retardos de la salida, q los retardos de la entrada y m los retardos de la referencia. Típicamente, el número de retardos empleados aumenta con el orden de la planta.

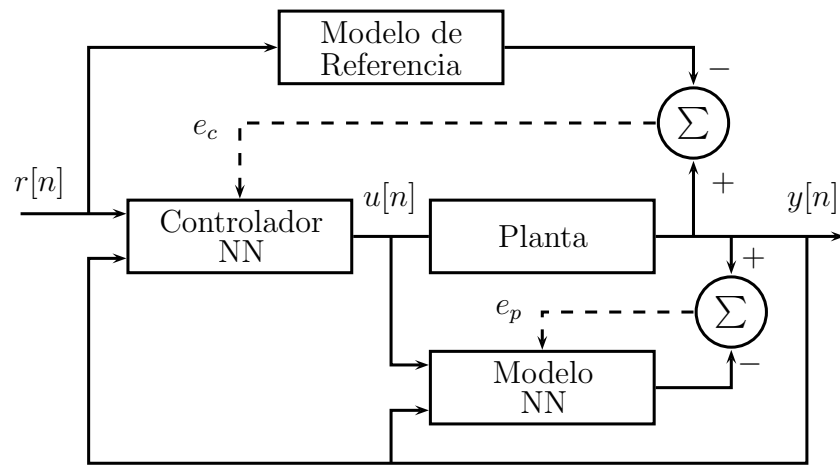


Figura 6.7: Esquema de control mediante red neuronal como modelo de referencia.

Capítulo 7

Identificación supervisada mediante sistemas neuronales

7.1. Introducción

La inteligencia computacional proporciona mecanismos alternativos para la solución de problemas, siendo particularmente útil en aplicaciones donde se requiere la identificación de sistemas no lineales complejos. En este sentido, las redes neuronales son una buena alternativa para la identificación de sistemas dinámicos no lineales.

En este capítulo se presentan conceptos relacionados con la identificación de sistemas dinámicos mediante sistemas neuronales. También se revisan los pasos involucrados en el proceso de identificación. Finalmente se revisa el esquema general de algoritmo de identificación utilizando redes neuro-difusas.

7.2. Identificación de sistemas

El modelado es un aspecto crucial al momento de establecer la solución de un problema, al respecto se tienen los siguientes enfoques generales:

- Modelo de caja blanca: En este caso es posible representar el sistema bajo estudio mediante un conjunto de ecuaciones bien constituidas. Cuando el sistema es complejo y poco conocido se pueden presentar dificultades para establecer este conjunto de ecuaciones.
- Modelo de caja negra: Para este modelo se utiliza una estructura general para la representación del sistema. La dificultad que se tiene consiste en establecer la estructura más adecuada que describe el sistema.
- Modelo de caja gris: Con este enfoque se combinan las ventajas de los modelos de caja negra y blanca, empleando una estructura general basada en un conocimiento previo del sistema considerado.

Los anteriores enfoques tienen varias aplicaciones, la orientación de caja blanca se emplea para el análisis del sistema de forma analítica. El modelo de caja negra se utiliza en aplicaciones computacionales mediante datos tomados de este. Por su parte, el modelo de caja gris permite representar cierta información relevante del sistema considerando un conocimiento previo [206].

Desde un punto de vista general, el proceso de identificación consiste en estimar los parámetros del modelo. Al disponerse de datos asociados al comportamiento del sistema se puede emplear fácilmente un modelo de caja negra sin necesidad de un conocimiento específico del sistema. De acuerdo con [207], la identificación busca establecer un modelo matemático empleando datos medidos. En la figura 7.1 se puede apreciar el esquema general empleado para la identificación de sistemas.

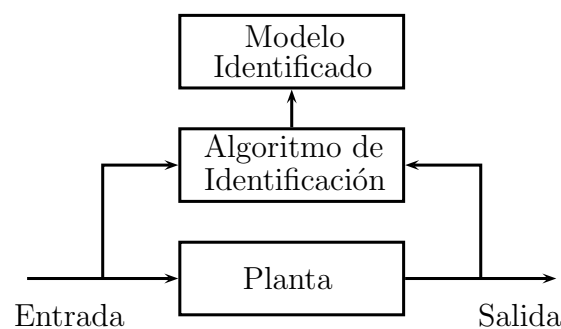


Figura 7.1: Proceso general de identificación.

Considerando lo presentado en [6], los pasos asociados al proceso de identificación son:

1. Adquisición de las parejas de datos de entrada-salida.
2. Adecuación de los datos según las características del modelo realizando procesos de filtrado, escalado y normalización entre otros.
3. Selección de la estructura del modelo, lo cual se puede realizar de forma experimental o basada en un conocimiento previo del sistema.
4. Estimación de los parámetros del sistema, para lo cual se pueden emplear algoritmos de ajuste.
5. Validación del modelo para aceptarlo o rechazarlo.

En el caso que el modelo no cumpla con la validación, es necesario observar si los datos de entrada y salida proporcionan suficiente información sobre el sistema. También sin la estructura escogida logra una buena descripción del modelo [192].

Por su parte, según los requerimientos de la aplicación, la identificación puede ser *on-line* o de otra forma, *off-line*. La identificación *off-line* se utiliza cuando no se requiere el ajuste continuo del modelo. En este caso, se considera que los parámetros obtenidos no se alteran con el paso del tiempo. Por su parte, en la identificación *on-line* los parámetros se actualizan de forma continua con los nuevos datos de entrada-salida obtenidos durante la evolución del proceso [207].

7.3. Identificación mediante redes neuronales

La identificación de sistemas dinámicos consiste en construir modelos matemáticos empleando mediciones reales de las entradas y salidas del sistema [6, 208]. Para la identificación del sistema, se debe seleccionar la estructura del modelo el cual se encuentra compuesto por un determinado grupo de parámetros. Adicionalmente se debe establecer un algoritmo que permita identificar los parámetros de este modelo [6].

Para sistemas lineales, la estimación de la estructura se puede realizar directamente de las ecuaciones que lo describen, por otra parte, para sistemas no lineales esta labor no es tan directa, por lo cual, las redes neuronales resultan ser una buena alternativa como estructura que pueda representar las no linealidades del sistema [6].

Para la identificación de sistemas dinámicos mediante redes neuronales se tiene dos esquemas tradicionales denominados serie-paralelo y paralelo [209]. Una representación de estos esquemas se puede apreciar en la figura 7.2.

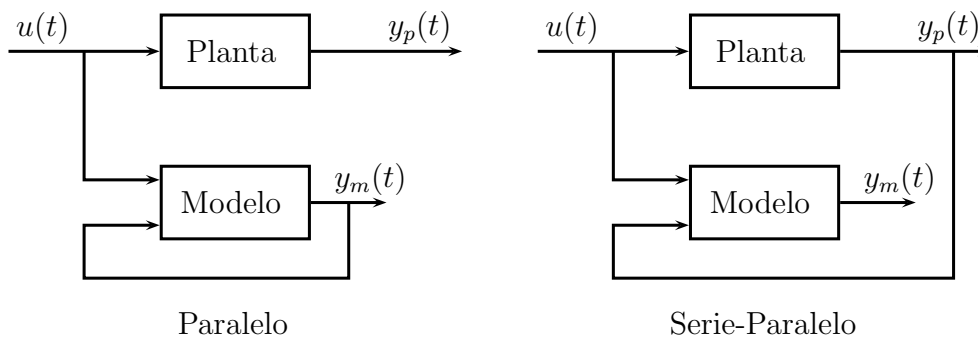


Figura 7.2: Esquema de identificación paralelo y serie-paralelo.

En el esquema de identificación paralela se utilizan las entradas de la planta y se realimenta la salida de la red a las entradas de esta, en este enfoque se suele emplear como algoritmo de entrenamiento el *Dynamic Back Propagation* (DBP). Por su parte en el esquema de identificación serie-paralelo se utilizan las entradas y salidas de la planta. Con este enfoque, se puede utilizar como método de entrenamiento el algoritmo tradicional *Back Propagation* (BP) [208].

7.4. Identificación empleando sistemas de lógica difusa

En este caso, como estructura del modelo utilizado para realizar la identificación se emplea un sistema de lógica difusa, lo que permite incorporar información preliminar de la estructura del sistema en forma lingüística mediante reglas. Considerando lo presentado en [32], principalmente se tienen dos enfoques para la integración del conocimiento empleando sistemas de lógica difusa:

1. Un conocimiento preliminar del sistema, por lo general basado en la experiencia,

se expresa en forma de reglas *Si...Entonces*, permitiendo tener una estructura del modelo. Mediante algoritmos de entrenamiento (optimización) se establecen los mejores valores de los parámetros del modelo de tal forma que la respuesta del modelo se acerca a los datos medidos [6, 207].

2. La estructura del sistema difuso se obtiene a partir de los datos medidos, de esta forma, la estructura y valores de los parámetros del sistema difuso puede proporcionar una interpretación del comportamiento del sistema [6, 207].

Para la selección de la estructura del sistema en el proceso de identificación empleando sistemas difusos, se deben tener presentes los siguientes aspectos:

1. Variables de entrada y salida.
2. Estructura de las reglas.
3. Número y tipo de funciones de pertenencia para cada variable.
4. Tipo del mecanismo de inferencia, operadores de conexión y método de defuzificación.

Según [32], la identificación mediante lógica difusa involucra el diseño del sistema empleando un conocimiento previo y el ajuste de sus parámetros mediante datos medidos. El éxito de este enfoque depende de la información preliminar que se disponga del sistema.

Otro enfoque consiste en utilizar un esquema general y sistemático que permita particionar los universos de discurso, y de esta forma relacionarlos mediante un conjunto de reglas. De esta manera no se requiere un conocimiento preliminar, sin embargo, puede llegar a ser ineficiente ya que el número de reglas se incrementa en la medida que se realicen más particiones de los universos de discurso [192].

7.5. Identificación mediante sistemas neuro-difusos

De acuerdo con [5, 6, 8, 9] los sistemas difusos en relación a las redes neuronales presentan una mayor capacidad para representar el conocimiento estructurado o basado en la experiencia. Por su parte, las redes neuronales son sobresalientes por sus capacidades de aprendizaje y optimización. Los sistemas neuro-difusos aprovechan la integración entre estas técnicas.

Entre las características generales de los sistemas neuro-difusos se tiene la integración de conocimiento humano, bio-inspiración, aplicación en problemas reales, implementaciones computacionales y aprendizaje supervisado, entre otras [5]. Los sistemas neuro-difusos consisten en la representación de un sistema difuso como una red neuronal de tal forma que se puede realizar el ajuste de sus parámetros mediante algoritmos de entrenamiento empleados con redes neuronales.

A diferencia de las redes neuronales en un sistema neuro-difuso, los nodos tienen diferentes funciones y no están completamente conectados con otros nodos en las capas

vecinas. Algunos nodos representan los términos lingüísticos de las variables de entrada, otros mediante conexiones, permiten representar reglas difusas [5, 6]. De esta forma, los sistemas neuro-difusos permiten incluir un conocimiento preliminar en la fase de diseño de la estructura del sistema. Los parámetros libres del sistema se ajustan empleando datos de parejas de entrada-salida mediante algoritmos de entrenamiento [10].

7.6. Proceso general en un algoritmo de entrenamiento

Durante el proceso de entrenamiento, de forma iterativa y mediante datos disponibles, se ajustan los parámetros del sistema empleado para la identificación. De acuerdo con [192], [197] y [210] de forma general un algoritmo de entrenamiento tanto para una red neuronal como en un sistema neuro-difuso comprende los siguientes pasos:

1. **Establecer la estructura y los parámetros iniciales:** Considerando la naturaleza del fenómeno a modelar se determina la estructura del sistema que mejor se ajuste. Esta estructura permite además establecer los valores iniciales los cuales pueden ser aleatorios (para redes neuronales) o basados en un conocimiento preliminar (en sistemas neuro-difusos).
2. **Cálculo de la salida del sistema:** Utilizando un par de datos de entrada-salida se calcula el respectivo valor de la salida del sistema utilizado para la identificación.
3. **Actualización de los parámetros del sistema:** Mediante el respectivo algoritmo de entrenamiento (por lo general *Back Propagation*), y con el cálculo de la salida del sistema se realiza la actualización de los parámetros de este sistema.
4. **Regresar al paso 2:** Se repite este proceso hasta que se logre una mejora de la función objetivo establecida en el algoritmo de entrenamiento.
5. **Regresar al paso 2:** Se utiliza otra pareja de entrada salida y repite el proceso de entrenamiento actualizando los parámetros del sistema con estos nuevos datos.

Parte III

Desarrollo

Capítulo 8

Sistemas de inferencia difusa basados en relaciones booleanas

8.1. Introducción

En este capítulo se presenta la formulación de los sistemas de inferencia difusa basados en relaciones booleanas, con este enfoque se busca tener un modelo basado en el diseño de sistemas mediante álgebra booleana.

Desde el punto de vista de automatismos diseñados con álgebra booleana se tiene una acción sobre uno o varios actuadores, esta acción suele ser todo o nada generando transiciones bruscas en el proceso. Con esta propuesta se busca que estas transiciones no sean bruscas, para lo cual se convierten los conjuntos booleanos en difusos bajo ciertos criterios de diseño, principalmente utilizando álgebra de Kleene.

Con este enfoque, se parte diseñando el sistema empleando conjuntos booleanos los cuales se extienden a difusos empleando álgebra de tres elementos. Mediante reglas de inferencia se obtienen relaciones entre los diferentes conjuntos difusos de cada universo de discurso y una acción parcial denominada salida virtual; de esta forma la acción total corresponde a la suma de las salidas virtuales.

En la primera parte se muestra el esquema general de los sistemas de lógica difusa basados en relaciones booleanas, luego se describe la metodología de diseño utilizando álgebra de Kleene para convertir una tabla de Boole a una de Kleene y así lograr la equivalencia entre las tablas. Además se presentan varios criterios de diseño para convertir conjuntos booleanos a difusos. Posteriormente, se analizan diferentes casos donde es posible lograr la simplificación de términos que implementan el sistema de lógica difusa de tal manera que se logran tener sistemas compactos. Finalmente se muestran las dos arquitecturas empleadas en el sistema de control adaptativo para la implementación del controlador y la identificación de la planta.

8.2. Sistemas de inferencia difusa basados en relaciones booleanas

Los sistemas de inferencia difusa (*Fuzzy Inference System* FIS) tienen gran aplicabilidad en el campo de control debido a su flexibilidad para la implementación de estrategias de control cuando existe imprecisión, ambigüedad o incertidumbre en el modelo de la planta. Con este enfoque, una de las formas de realizar control sobre un sistema es utilizando razonamiento lógico [11, 211]. Por otro lado, el álgebra de Boole es utilizada para implementar sistemas de inferencia bivaluados, siendo así una herramienta útil en el diseño de reglas para la automatización (*Boolean Automation Systems* BAS) [212, 213, 214, 215]. Sin embargo, los sistemas de control basados en BAS presentan un desempeño limitado debido a transiciones abruptas de las acciones de control. Una forma de mejorar el desempeño de estos sistemas consiste en reemplazar los conjuntos booleanos por conjuntos difusos.

Cuando se busca realizar la transición de un diseño booleano a uno difuso se debe tener presente que varias propiedades del álgebra de Boole con conjuntos booleanos no se cumplen para los conjuntos difusos, sin embargo, para lograr la equivalencia y así tener un diseño consistente, se puede emplear álgebra de Kleene, la cual considera un tercer elemento intermedio a los valores booleanos 0 y 1.

El álgebra Boole (\mathcal{B}) es la base para las herramientas que se han desarrollado utilizando conjuntos cuyo valor de pertenencia viene dado por la dupla $\{0, 1\}$; por su parte, el álgebra de Kleene (\mathcal{K}) incluye un tercer elemento u teniendo una estructura de una tupla $\{0, u, 1\}$ [184]. De acuerdo con [184], el álgebra de Kleene permite la implementación del cálculo proposicional difuso.

Los sistemas de inferencia difusa basados en relaciones booleanas (*Fuzzy Inference System - Based on Boolean Relations* FIS-BBR), buscan plantear un método de implementación para los sistemas de lógica difusa, tomando como referencia el diseño de automatismos, ya que estos sistemas son muy empleados en control de procesos por su facilidad de implementación [212, 213]. Esta técnica de diseño considera los sensores, los actuadores y las relaciones booleanas involucradas en las estrategias de control de un proceso industrial [41, 216, 217].

En [41] se presenta el esquema general para un FIS-BBR considerando la conversión de funciones de pertenencia booleanas a difusas; sin embargo, hay varios aspectos por cubrir en el proceso de diseño. Con el fin de establecer un proceso de diseño, se aprovecha que diferentes álgebras de valores de verdad pueden proporcionar la misma lógica proposicional y que la lógica proposicional difusa es equivalente a la lógica proposicional de Kleene de tres valores [187]. De esta manera, una tabla de Boole se puede ampliar a una de Kleene, teniendo presente condiciones de regularidad, así como varios principios de diseño. Usando una tabla y álgebra de Kleene, es posible establecer relaciones para el proceso de inferencia junto con un proceso de simplificación, lo que permite tener una equivalencia con las expresiones obtenidas mediante una tabla y álgebra de Boole.

8.2.1. Elementos del sistema de inferencia difusa basada en relaciones booleanas

Los sistemas FIS-BBR consideran la metodología para el diseño de sistemas de inferencia difusa basados en tablas de verdad. Los componentes de un sistema de inferencia difusa basado en relaciones booleanas se pueden observar en la figura 8.1 donde se aprecia la importancia que tienen el cálculo de la salida virtual en este proceso.

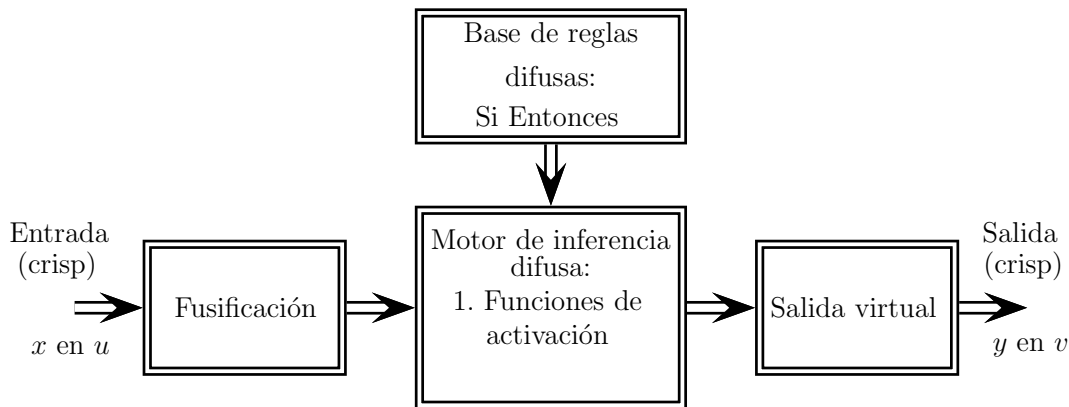


Figura 8.1: Elementos de un sistema de inferencia difusa basado en relaciones booleanas.

Los FIS-BBR son una alternativa para implementar sistemas de control, puesto que se fundamentan en los principios de diseño de automatismos (como el aplicado a los circuitos digitales) extendiéndose a sistemas difusos utilizando para esto álgebra de Kleene. A partir de las reglas de inferencia, las relaciones booleanas que aparecen en una tabla de verdad se pueden extender a tres elementos de Kleene teniendo en cuenta las transiciones entre las acciones de control.

8.2.2. Salida virtual

En los sistemas de control el actuador es el elemento utilizado para aplicar la acción de control [11]. En el diseño de automatismos, es posible emplear varios actuadores para la acción de control (varias baterías, válvulas, interruptores...). Sin embargo, cuando solo hay un elemento de acción, este se puede segmentar en varios actuadores denominados “virtuales”, que al mismo tiempo tienen una salida virtual asociada. Desde este punto de vista, en un sistema de lógica los actuadores virtuales pueden considerarse como valores de acción o salida asociados al universo del discurso en el consecuente. De acuerdo con la aplicación, el valor de los actuadores virtuales puede ser positivo o negativo, y también la acción total puede ser mayor que las acciones virtuales parciales [41].

De esta manera la acción o resultado del sistema de inferencia corresponde a la suma de las respectivas salidas virtuales:

$$y = y_1 + y_2 + \cdots + y_m + \cdots + y_M$$

$$y = \sum_{m=1}^M y_m$$

Donde $y_m = Y_m v_m$, siendo Y_m la función de activación, y v_m el valor del respectivo actuador virtual. El término para salida virtual se concibe bajo una correspondencia real en aplicaciones de control.

A manera de ejemplo, considerando la acción de una válvula de $6\text{cm}^3/\text{s}$ para llenar un tanque, esta se puede representar como la contribución de tres válvulas diferentes dadas por los actuadores virtuales:

- Flujo pequeño: $v_1 = 1\text{cm}^3/\text{s}$.
- Flujo medio: $v_2 = 2\text{cm}^3/\text{s}$.
- Flujo grande $v_3 = 3\text{cm}^3/\text{s}$.

La figura 8.2 muestra el esquema de llenado de un tanque con tres válvulas virtuales.

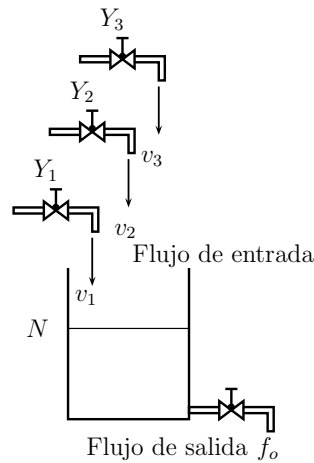


Figura 8.2: Sistema de un tanque con tres válvulas “virtuales”.

Usando estas válvulas virtuales, en la tabla 8.1 se muestran las posibles acciones de llenado, donde ON corresponde a la activación de la válvula.

Actuador	Conf. 1	Conf. 2	Conf. 3	Conf. 4	Conf. 5	Conf. 6	Conf. 7	Conf. 8
v_1	OFF	ON	OFF	OFF	ON	ON	OFF	ON
v_2	OFF	OFF	ON	OFF	ON	OFF	ON	ON
v_3	OFF	OFF	OFF	ON	OFF	ON	ON	ON
Flujo	$0\text{cm}^3/\text{s}$	$1\text{cm}^3/\text{s}$	$2\text{cm}^3/\text{s}$	$3\text{cm}^3/\text{s}$	$3\text{cm}^3/\text{s}$	$4\text{cm}^3/\text{s}$	$5\text{cm}^3/\text{s}$	$6\text{cm}^3/\text{s}$

Tabla 8.1: Diferentes acciones para actuadores virtuales.

Considerando lo anterior, es posible definir las “acciones booleanas” como las posibles acciones producidas usando conjuntos booleanos.

8.3. Proceso de inferencia difusa en un FIS-BBR

Teniendo en cuenta que los sistemas de inferencia difusa basados en relaciones booleanas usan herramientas de diseño y codificación de sistemas booleanos, la estructura del proceso de inferencia viene dada por la tabla de verdad asociada con el diseño booleano.

Para la implementación de un FIS-BBR se considera que es posible tener una equivalencia para la tabla de Boole y Kleene obteniendo así las mismas ecuaciones de inferencia.

En la operación de un FIS-BBR se consideran acciones que contribuyen a una acción total sobre un sistema. Las acciones se realizan ponderando un actuador v_m (actuador virtual) usando una función Y_m que opera sobre él (función de activación), que es una función difusa con valores entre cero y uno. De esta manera, la salida en un FIS-BBR se calcula como la suma de los resultados parciales $y_m = v_m Y_m$ (salidas virtuales), como se muestra en la ecuación (8.3).

La tabla 8.2 muestra una posible codificación para las entradas y acciones de control de un sistema. Las filas del lado izquierdo representan las reglas que proporcionan una salida de activación. Las columnas del lado derecho corresponden a las funciones de activación. De esta forma se identifican dos partes en la tabla 8.2, una correspondiente a una codificación para los conjuntos de cada entrada; la otra corresponde a las salidas de activación asociadas a las salidas virtuales.

Entradas						Funciones de activación					
A_1	A_2	\dots	A_j	\dots	A_P	Y_1	Y_2	\dots	Y_m	\dots	Y_M
$a_{1,1}$	$a_{1,2}$	\dots	$a_{1,j}$	\dots	$a_{1,P}$	$f_{1,1}$	$f_{1,2}$	\dots	$f_{1,m}$	\dots	$f_{1,M}$
$a_{2,1}$	$a_{2,2}$	\dots	$a_{2,j}$	\dots	$a_{2,P}$	$f_{2,1}$	$f_{2,2}$	\dots	$f_{2,m}$	\dots	$f_{2,M}$
\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
$a_{k,1}$	$a_{k,2}$	\dots	$a_{k,j}$	\dots	$a_{k,P}$	$f_{k,1}$	$f_{k,2}$	\dots	$f_{k,m}$	\dots	$f_{k,M}$
\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
$a_{Q,1}$	$a_{Q,2}$	\dots	$a_{Q,j}$	\dots	$a_{Q,P}$	$f_{Q,1}$	$f_{Q,2}$	\dots	$f_{Q,m}$	\dots	$f_{Q,M}$

Tabla 8.2: Tabla de verdad con reglas de control.

Para el cálculo de una función de activación Y_m en la forma normal disyuntiva, se tiene:

$$Y_m = \bigvee_{k=1}^Q \bigwedge_{j=1}^P \hat{A}(a_{k,j}, f_{k,m}) \quad (8.1)$$

La función \hat{A} para el álgebra de Boole puede ser A_j o \bar{A}_j , en el caso de Kleene se tienen las posibilidades de A_j , \bar{A}_j o $A_j \wedge \bar{A}_j$. La expresión de \hat{A} tanto para el álgebra de Boole como de Kleene dependen de los valores de $a_{k,j}$ y $f_{k,m}$ [187], en la tabla 8.3 se pueden observar estos valores.

$a_{k,j}$	$f_{k,m}$	\hat{A}
0	0	0
u	0	0
1	0	0
0	u	\bar{A}_j
u	u	$A_j \wedge \bar{A}_j$
1	u	A_j
0	1	\bar{A}_j
u	1	1
1	1	A_j

Tabla 8.3: Valores para la función \hat{A} .

De acuerdo con [11], sobre el uso de t-normas y t-conormas (s-normas) como operadores en la implementación de sistemas de lógica difusa, para la salida de activación Y_m se tiene la ecuación (8.2), donde \oplus indica la operación difusa de t-conorma (máximo, suma algebraica, etc.); y \odot representa la operación difusa de t-norma (mínimo, producto, etc.). También es de señalar que $\mu_{\hat{A}}$ es la expresión asociada de \hat{A} en términos de funciones de pertenencia.

Adicionalmente, sobre la posibilidad de emplear t-normas y t-conormas en el proceso de inferencia, se presentan diferentes aspectos a considerar tal como se muestra en [187] y [218]. Particularmente, para el proceso de diseño se debe tener presente el cumplimiento de las propiedades algebraicas utilizadas, por ejemplo, para la simplificación de términos la propiedad distributiva [219].

$$Y_m = \bigoplus_{k=1}^Q \bigodot_{j=1}^P \mu_{\hat{A}} \quad (8.2)$$

Teniendo en cuenta que cada columna Y_m corresponde a una acción, la salida global del sistema se determina como:

$$y = \sum_{m=1}^M Y_m v_m \quad (8.3)$$

Donde v_m corresponde al m -ésimo actuador virtual, entonces, la m -ésima salida virtual es:

$$y_m = Y_m v_m \quad (8.4)$$

Considerando la tabla booleana 8.4, la figura 8.3 muestra un ejemplo gráfico para el proceso de inferencia difusa basado en relaciones booleanas. Como primer paso para el cálculo de la salida (sin simplificar), la t-norma se aplica entre los conjuntos a lo largo de la fila donde la salida de activación se encuentra activa (un valor de 1). El segundo paso consiste en calcular cada función de activación, para esto, se aplica la t-conorma a lo largo de cada columna entre los resultados obtenidos del primer paso (asemejando a

un proceso de agregación). Posteriormente, con las funciones de activación, se ponderan los respectivos actuadores virtuales obteniendo las salidas virtuales. Finalmente, al sumar esos productos se obtiene la acción total de salida.

A_1	A_2	Y_1	Y_2
1	1	1	1
0	1	0	0
1	0	0	0
0	0	0	1

Tabla 8.4: Ejemplo de la codificación booleana asociada a la figura 8.3.

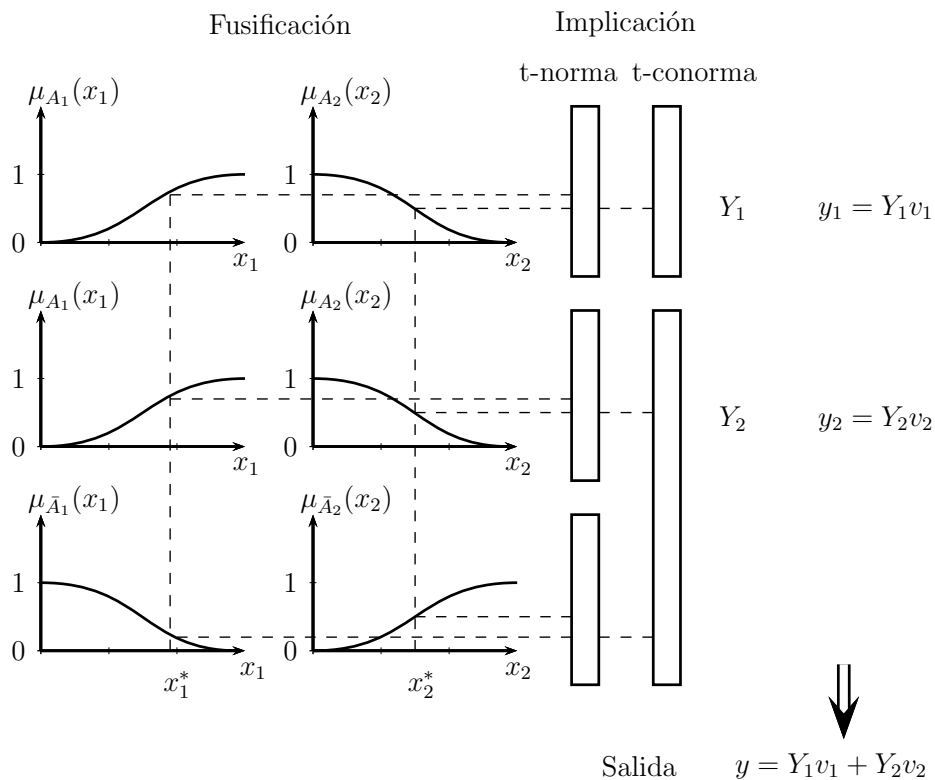


Figura 8.3: Operación de un sistema de inferencia difusa basado en relaciones booleanas.

8.4. Metodología de diseño

En primer lugar, la metodología de diseño considera la codificación de las acciones de control mediante conjuntos booleanos en una tabla de verdad. En segundo lugar, se tiene la transformación de estos conjuntos booleanos a difusos buscando tener transiciones monótonas continuas entre acciones booleanas. De esta manera se extiende la tabla de verdad de Boole a una de Kleene adicionando el elemento intermedio u para lo cual se requiere cumplir con la condición de regularidad. Bajo este enfoque, para convertir los conjuntos booleanos a difusos y convertir la tabla de verdad de Boole a una de Kleene se tienen los siguientes aspectos:

- Transiciones monótonas entre acciones booleanas.
- Cumplimiento de condiciones de regularidad.

Las acciones booleanas se pueden definir como las diferentes acciones obtenidas por las funciones de activación y los actuadores virtuales cuando se tienen conjuntos booleanos, de esta forma, las acciones booleanas están dadas por las salidas virtuales $y_m = v_m Y_m$. Las transiciones continuas y monótonas entre acciones booleanas buscan mejorar la transición (deseada) desde una acción que produce y_m hasta otra acción producida por y_n , para esto, se reemplazan los conjuntos booleanos por difusos. En la figura 8.4(a) se puede observar el ejemplo para una función $y(x)$ compuesta por acciones booleanas; en la figura 8.4(b) se aprecia la transición monótona entre acciones booleanas; y en la figura 8.4(a) una transición que no es monótona.

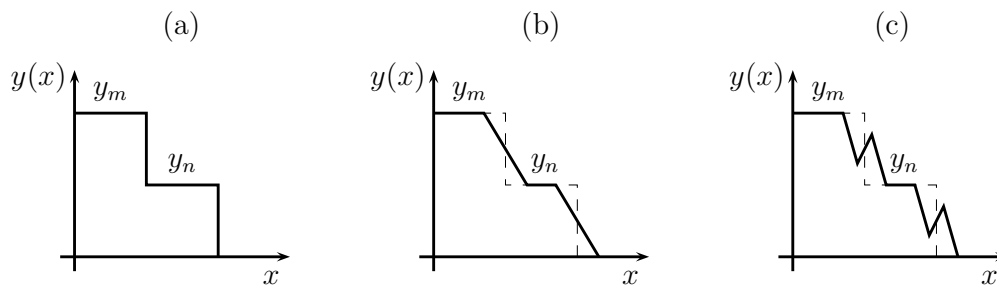


Figura 8.4: Ejemplos de transiciones entre acciones booleanas.

Para la metodología de diseño, en primer lugar, se definen las respectivas acciones que deben realizarse, luego los conjuntos booleanos se convierten en difusos utilizando criterios de diseño teniendo en cuenta la tabla ampliada con el tercer elemento u (tabla de Kleene). En segundo lugar, se determina la fórmula y se simplifican las funciones de activación, si es necesario. Finalmente, se realiza el proceso de inferencia utilizando conjuntos difusos. La figura 8.5 muestra la metodología de diseño.

Cuando se lleva a cabo la extensión a la tabla de Kleene, cada salida de activación se codifica como un valor $\{0, u, 1\}$, teniendo presente que esta tabla debe cumplir la condición de regularidad.

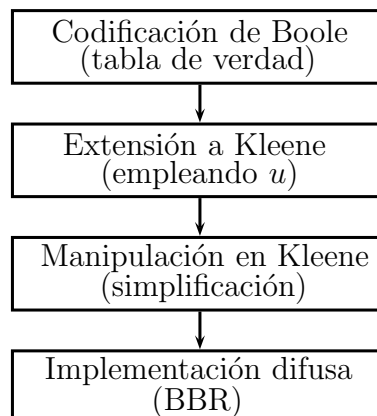


Figura 8.5: Metodología de diseño.

8.4.1. Condición de regularidad

De acuerdo con [220] y [221], la condición de regularidad permite manejar la ambigüedad “*las funciones regulares tratan la ambigüedad desde un punto de vista de lógica ternaria*”. Esta condición establece que si la ambigüedad en la entrada aumenta, entonces la ambigüedad en la salida también aumenta [220].

Considerando $E = \{0, u, 1\}$ y una función de lógica ternaria f de n -variables, la cual es un mapeo de E^n a E el cual para $X \in E^n$ asigna $f(X) \in E$. De esta forma, considerando la figura 8.6 una relación de ambigüedad \preceq corresponde a:

$$0 \preceq u, \quad 1 \preceq u, \quad i \preceq i, \quad (8.5)$$

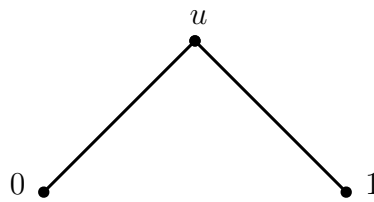


Figura 8.6: Orden parcial \preceq de ambigüedad.

Esto significa que 0 y 1 son menos ambiguos que u , de forma general, si $i \preceq j$ entonces i es menos ambiguo que j . Además, para elementos distintos $X = (x_1, \dots, x_n)$ y $Y = (y_1, \dots, y_n)$ de E^n , entonces X es menos ambiguo que Y si $x_i \preceq y_i$ para cada $i = 1, 2, \dots, n$, es decir $X \preceq Y$. Considerando que 0 y 1 son estados definidos y u un estado no definible como 0 o 1, entonces, el orden parcial \preceq permite describir la ambigüedad [221].

De acuerdo con [220] y [221], una función f es regular si es monótonamente ambigua cumpliendo con:

$$f(A) \preceq f(B) \text{ cuando } A \preceq B \quad (A, B \in E^n). \quad (8.6)$$

Esta condición implica que si A es igual o menos ambiguo que B , entonces, $f(A)$ es igual o menos ambiguo que $f(B)$. Por ejemplo, con (x_1, x_2) de la figura 8.7; para las combinaciones $(1, 1)$ y $(u, 1)$, se tiene que $(1, 1) \preceq (u, 1)$; por lo tanto, $f(1, 1) \preceq f(u, 1)$, de esta manera, si $f(1, 1) = 1$ implica que $f(u, 1)$ puede ser 1 o u para que $f(x_1, x_2)$ sea una función regular.

$x_2 \backslash x_1$	0	u	1
0	$f(0, 0)$	$f(0, u)$	$f(0, 1)$
u	$f(u, 0)$	$f(u, u)$	$f(u, 1)$
1	$f(1, 0)$	$f(1, u)$	$f(1, 1)$

Figura 8.7: Ejemplo de una tabla de tres elementos.

8.4.2. Extensión de una tabla de Boole a una de Kleene

Para realizar la extensión de una tabla booleana a una kleeneana, en primer lugar, es necesario manejar la ambigüedad cumpliendo la condición 8.6. Adicionalmente se debe tener presente la mejor configuración para lograr transiciones monótonas entre acciones booleanas. La tabla de Kleene se obtiene agregando casos intermedios en la tabla booleana usando la variable u en el antecedente, y $\{0, u, 1\}$ en el consecuente de acuerdo con las anteriores consideraciones.

En la figura 8.8 se puede ver un ejemplo para la extensión de una tabla bivalente a una tabla trivalente cumpliendo las condiciones de regularidad. En este ejemplo para la condición (8.6) son posibles dos opciones; sin embargo, la última configuración se utiliza para lograr transiciones monótonas entre acciones booleanas.

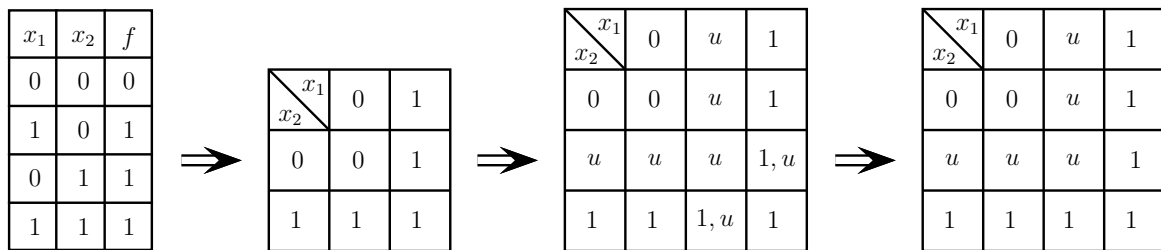


Figura 8.8: Extensión de tablas bivalentes a trivalentes logrando condiciones de regularidad.

8.4.3. Consideraciones para la selección de conjuntos difusos

Como el sistema propuesto se basa en el diseño de automatismos, los conjuntos booleanos se establecen teniendo en cuenta los posibles sensores que dividen los universos del discurso de entrada en diferentes acciones booleanas de control.

Los conjuntos difusos se crean a partir de conjuntos booleanos previamente establecidos, considerando la equivalencia entre la tabla de Boole y la de Kleene, buscando tener transiciones continuas y monótonas entre las respectivas acciones de control booleanas. De esta forma se proponen los siguientes criterios.

Criterio de equivalencia con la tabla de Boole

Con este enfoque, se busca que los conjuntos difusos escogidos permitan reconstruir los conjuntos booleanos originales. Este criterio de conversión se basa en el α -corte (ver [191]), donde considerando un valor $\alpha \in [0, 1]$, el α -corte de un conjunto difuso A corresponde a los elementos de X con un valor de pertenencia μ_A igual o mayor que α cuando:

$$A_\alpha = \{x \in X | \mu_A(x) \geq \alpha\} \quad (8.7)$$

Por lo tanto, el criterio propuesto se basa en un nivel α (dado por el diseñador) para obtener un conjunto booleano B partiendo de una función de pertenencia $\mu(x)$ de la

forma (8.8), donde el conjunto booleano $B(x)$ debe coincidir con el mismo empleado para la codificación en la tabla booleana.

Un ejemplo de conversión de un conjunto booleano a uno difuso usando un nivel α se puede apreciar en la figura 8.9, obteniendo las funciones de pertenencia $\mu_1(x)$ y $\mu_2(x)$.

$$B(x) = \begin{cases} 1, & \text{si } \mu(x) \geq \alpha; \\ 0, & \text{si } \mu(x) < \alpha. \end{cases} \quad (8.8)$$

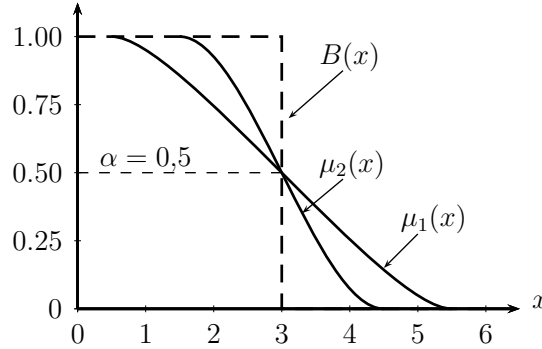


Figura 8.9: Ejemplo de conversión de un conjunto booleano a un conjunto difuso utilizando un nivel α .

Criterio de correspondencia de la tabla de Kleene

En este caso, se busca que los conjuntos difusos permitan llegar a la misma codificación establecida en la tabla de Kleene. Para esto se consideran dos valores $\delta_L, \delta_U \in [0, 1]$, con $\delta_L \leq \delta_U$ (definidos por el diseñador), así, para una función de pertenencia difusa $\mu(x)$ se puede calcular una función de tres elementos $\nu(x)$ mediante la ecuación (8.9). Entonces, el criterio consiste en el cálculo de $\nu(x)$ para mantener la misma codificación empleada en la tabla de Kleene.

$$\nu(x) = \begin{cases} 1, & \text{si } \mu(x) \geq \delta_U; \\ u, & \text{si } \delta_L < \mu(x) < \delta_U; \\ 0, & \text{si } \mu(x) \leq \delta_L. \end{cases} \quad (8.9)$$

Criterio de convexidad

Puesto que uno de los aspectos importantes de los sistemas BBR es la transición monótona entre las acciones booleanas, se propone que los conjuntos difusos obtenidos sean convexos.

De acuerdo con [191], para que un conjunto difuso A sea convexo $\forall x_1, x_2 \in X$ con $\lambda \in [0, 1]$, es necesario cumplir con la condición (8.10).

$$\mu_A(\lambda x_1 + (1 - \lambda)x_2) \geq \text{mín}[\mu_A(x_1), \mu_A(x_2)] \quad (8.10)$$

La figura 8.10 presenta un ejemplo de una función de pertenencia asociada a conjuntos difusos convexos y no convexos.

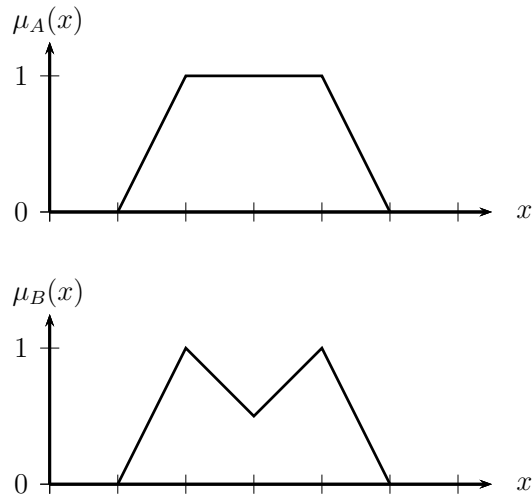


Figura 8.10: Ejemplo de conjuntos difusos convexos y no convexos.

8.5. Aplicación demostrativa para el control de nivel en un tanque

En el control de nivel de sistemas hidráulicos, este es un problema clásico que se ha manejado usando lógica difusa, como se observa en [222, 223, 224]. En este ejemplo, se parte de un diseño booleano donde se segmenta el universo del discurso correspondiente a la altura del líquido. El esquema de control considera tres posibles acciones basadas en el automatismo para llenar el tanque.

8.5.1. Modelo del sistema

El modelo considerado es similar al presentado en [224]. El esquema del sistema hidráulico se muestra en la figura 8.11.

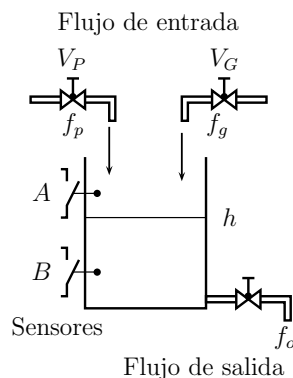


Figura 8.11: Sistema para el llenado de un tanque.

Las variables y parámetros utilizados son:

- $q_e(t)$: Flujo de entrada.

- $h(t)$: Altura del nivel de líquido en el tanque.
- A : Área de la sección transversal del tanque.
- A_0 : Área de sección transversal de la válvula de salida.
- $C_v = A_0\sqrt{2g}$: Constante de la válvula.

El modelo del sistema hidráulico es descrito por la ecuación (8.11).

$$q_e(t) = A \frac{dh(t)}{dt} + A_0 \sqrt{2gh(t)} \quad (8.11)$$

Otra forma de representarlo se tiene con la ecuación (8.12).

$$q_e(t) = A \frac{dh(t)}{dt} + C_v \sqrt{h(t)} \quad (8.12)$$

Para el caso considerado, los parámetros son: altura del tanque: $1m$; altura de referencia para llenar el tanque $80cm$; máximo flujo de salida $0,25L/s$ o $(2,5 \times 10^{-4}m^3/s)$; área transversal del tanque $0,126m^2$, diámetro de tubo de salida $1/4in$ o $(0,635cm)$; área transversal de la válvula de salida $0,317cm^2$ o $(3,17 \times 10^{-5}m^2)$, y la constante para la válvula de salida es $1,4 \times 10^{-4}m^{5/2}/s$.

También se considera un retardo temporal de $t_0 = 0,1s$ (asociado a la válvula), de esta manera, la relación entre la acción de control u y el flujo de entrada es $q_e(t) = u(t - t_0)$.

8.5.2. Diseño booleano

Un primer paso en el diseño consiste en analizar las transiciones dinámicas del sistema. En este sentido, se consideran diferentes niveles en el tanque, sus transiciones, así como las acciones realizadas en las válvulas. Los niveles considerados son:

- Nivel bajo: B .
- Nivel medio: M .
- Nivel alto: A .

Para las acciones de control hay dos válvulas V_P y V_G , que pueden ser reales o virtuales dependiendo de la implementación del hardware. Teniendo en cuenta las regiones y las válvulas de acción, se obtiene la tabla 8.5.

Evento	Estado	Consecuencia
1	Nivel bajo L	V_G y V_P activadas
2	Nivel intermedio M	V_P desactivada
3	Nivel alto H	V_P activada y V_G desactivada

Tabla 8.5: Tabla de eventos considerando el estado y las válvulas.

Las inferencias asociadas a la tabla booleana son:

- **Si** nivel h es bajo **Entonces** el flujo de entrada es grande.
- **Si** nivel h es intermedio **Entonces** el flujo de entrada es medio.
- **Si** nivel h es alto **Entonces** el flujo de entrada es bajo.

Para las válvulas (V_G : válvula grande y V_P válvula pequeña), las reglas de inferencia se pueden definir como:

- **Si** nivel h es bajo **Entonces** V_G y V_P se activan.
- **Si** nivel h es intermedio **Entonces** V_G activa y V_P inactiva.
- **Si** nivel h es alto **Entonces** V_G inactiva y V_P activa.

Para implementar las acciones en las válvulas, se definen dos funciones de activación que permiten la apertura y el cierre de las válvulas de tal manera que la acción total en el tanque es:

$$u = Y_{VG}V_G + Y_{VP}V_P \quad (8.13)$$

Cada válvula V_G y V_P tiene un flujo máximo asociado, que es:

- $V_G = 1,5L/s$
- $V_P = 1,0L/s$

8.5.3. Diseño FIS-BBR

En este caso, los conjuntos booleanos se reemplazan por conjuntos difusos bajo ciertas consideraciones. Del mismo modo, con los conjuntos difusos definidos, es posible la implementación de las relaciones encontradas en el diseño utilizando operaciones específicas como t-norma (mínimo) y t-conorma (máximo).

8.5.4. Conjuntos booleanos

La figura 8.11 considera posibles sensores que se desactivan o activan cuando el líquido es más bajo o más alto que un nivel definido. Las funciones de pertenencia booleanas propuestas para implementar el sistema de control se pueden ver en la figura 8.12

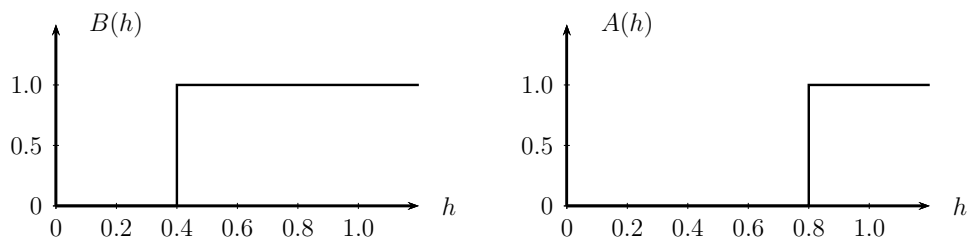


Figura 8.12: Conjuntos booleanos.

Las regiones que se tienen con los anteriores conjuntos son:

- Bajo: $\mu_B(h) = 0$ y $\mu_A(h) = 0$.
- Medio: $\mu_B(h) = 1$ y $\mu_A(h) = 0$.
- Alto: $\mu_B(h) = 1$ y $\mu_A(h) = 1$.

La tabla 8.6 representa la codificación booleana respectiva para los estados y acciones de control.

B	A	Y_{VG}	Y_{VP}
0	0	1	1
1	0	1	0
1	1	0	1
0	1	X	X

Tabla 8.6: Codificación booleana de acciones y estados.

Las ecuaciones para las funciones de activación son:

$$Y_{VG} = (\bar{B} \wedge \bar{A}) \vee (B \wedge \bar{A}) \quad (8.14)$$

$$Y_{VP} = (\bar{B} \wedge \bar{A}) \vee (B \wedge A) \quad (8.15)$$

Al aplicar la síntesis y considerando los casos “no importa” ($A = 0, B = 1$) como se muestra en la tabla 8.7, se tienen las ecuaciones (8.16) y (8.17). Los casos “no importa” se emplean únicamente para simplificar Y_{VP} .

$$Y_{VG} = \bar{A} \quad (8.16)$$

$$Y_{VP} = \bar{B} \vee A \quad (8.17)$$

B	A	Y_{VG}	Y_{VP}
0	0	1	1
1	0	1	0
1	1	0	1
0	1	0	1

Tabla 8.7: Codificación booleana utilizando los valores para X .

Para la extensión de una tabla de Boole a una de Kleene, se utiliza un valor u correspondiente a la transición entre valores 0 y 1 como se muestra en la figura 8.13. Para convertir la tabla de Boole 8.7 a una tabla de Kleene, primero se considera la condición de regularidad obteniendo la tabla 8.8. En la segunda etapa, se establece tabla 8.9 considerando las transiciones monótonas entre acciones booleanas. La tabla 8.10 muestra otra forma para representar la codificación realizada en la tabla 8.9.

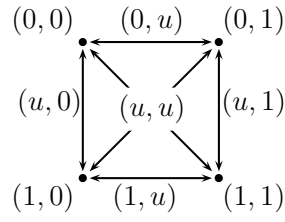


Figura 8.13: Transiciones booleanas usando u .

B	A	Y_{VG}	Y_{VP}
0	0	1	1
u	0	$\{1, u\}$	u
1	0	1	0
1	u	u	u
1	1	0	1
u	1	$\{0, u\}$	$\{1, u\}$
0	1	0	1
0	u	u	$\{1, u\}$
u	u	u	u

Tabla 8.8: Posible codificación de Kleene para estados y acciones.

B	A	Y_{VG}	Y_{VP}
0	0	1	1
u	0	1	u
1	0	1	0
1	u	u	u
1	1	0	1
u	1	0	1
0	1	0	1
0	u	u	1
u	u	u	u

Tabla 8.9: Codificación final de Kleene para estados y acciones.

Y_{VG}		A			Y_{VP}		A		
		0	u	1			0	u	1
B	0	1	u	0	B	0	1	1	1
	u	1	u	0		u	u	u	1
	1	1	u	1		1	0	u	1

Tabla 8.10: Representación de Kleene alternativa para estados y acciones.

Aplicando propiedades del álgebra de Kleene (ver [184]) se tienen las siguientes ecua-

ciones:

$$\begin{aligned}
Y_{VG} &= (\bar{B} \wedge \bar{A}) \vee \bar{A} \vee (B \wedge \bar{A}) \vee (B \wedge \bar{A} \wedge A) \\
&\quad \vee (\bar{B} \wedge \bar{A} \wedge A) \vee (\bar{B} \wedge B \wedge \bar{A} \wedge A) \\
Y_{VG} &= \bar{A} \wedge [\bar{B} \vee 1 \vee B \vee (B \wedge A) \vee (\bar{B} \wedge A) \\
&\quad \vee (\bar{B} \wedge B \wedge A)] \\
&= \bar{A}
\end{aligned} \tag{8.18}$$

Como es de apreciar, lo anterior se obtiene debido a la propiedad de absorción. Del mismo modo, aplicando el álgebra de Kleene para Y_{VP} se tiene:

$$\begin{aligned}
Y_{VP} &= (\bar{B} \wedge \bar{A}) \vee (\bar{B} \wedge B \wedge \bar{A}) \vee (B \wedge \bar{A} \wedge A) \\
&\quad \vee (B \wedge A) \vee A \vee (\bar{B} \wedge A) \vee \bar{B} \\
&\quad \vee (\bar{B} \wedge B \wedge \bar{A} \wedge A) \\
Y_{VP} &= \bar{B} \wedge [\bar{A} \vee (\bar{B} \wedge \bar{A}) \vee A \vee 1] \\
&\quad \vee A \wedge [(B \wedge \bar{A}) \vee B \vee 1 \vee (\bar{B} \wedge B \wedge \bar{A})] \\
Y_{VP} &= \bar{B} \vee A
\end{aligned} \tag{8.19}$$

Como es de notar, con este enfoque se obtienen las mismas expresiones que en el caso booleano cuando se aplica la simplificación. Aunque la propiedad utilizada para efectuar la simplificación con la tabla de Boole (tercer excluido) es diferente de la utilizada con la tabla de Kleene (absorción), es posible tener las mismas expresiones considerando las respectivas transiciones intermedias.

8.5.5. Implementación FIS-BBR

De acuerdo con las consideraciones para la selección de conjuntos difusos de la sección 8.4.3, se toman los valores $\delta_U = 1$ y $\delta_L = 0$, y un nivel de 0,5 para el α -corte de tal forma que se mantenga la codificación empleada en las tablas de Boole y Kleene.

En la figura 8.14 se proponen conjuntos difusos, los cuales se pueden asociar a sensores que miden el nivel de líquido presente en el tanque considerando un comportamiento lineal saturado.

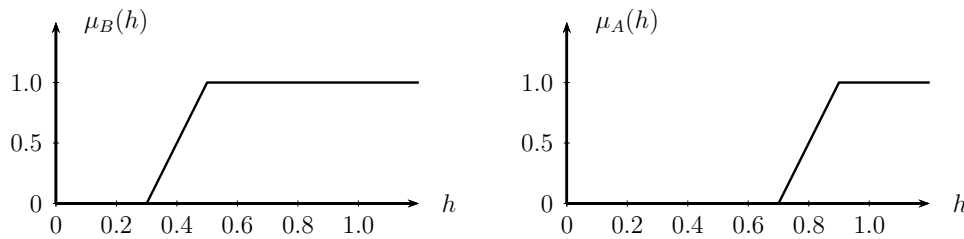


Figura 8.14: Conjuntos difusos propuestos.

Las ecuaciones simplificadas para la implementación del sistema son:

$$Y_{VG} = \mu_{\bar{A}} \quad (8.20)$$

$$Y_{VP} = \mu_{\bar{B}} \oplus \mu_A \quad (8.21)$$

Finalmente, tomando la ecuación (8.13) y usando “mín” para la t-norma, “máx” para la t-conorma y el complemento estándar, entonces, la acción de control corresponde a:

$$u = V_G(1 - \mu_A) + V_P \text{ máx}[(1 - \mu_B), \mu_A] \quad (8.22)$$

Para mostrar los procesos involucrados en la operación del controlador booleano y el BBR, las figuras 8.15 y 8.16 muestran los respectivos diagramas de bloques. En la figura 8.15 se presentan los conjuntos empleados para implementar el controlador booleano, mientras que la figura 8.16 muestra las funciones de pertenencia utilizadas para el controlador BBR.

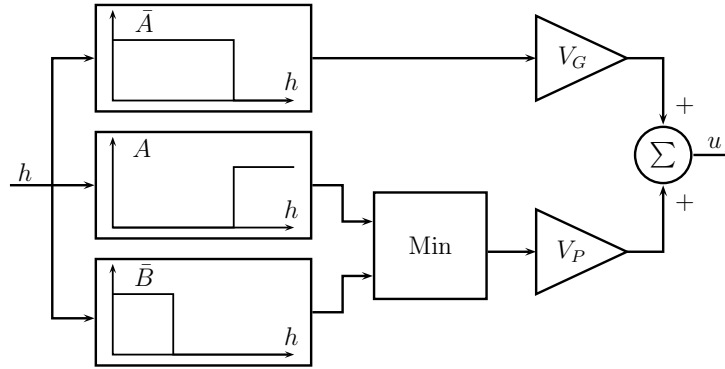


Figura 8.15: Diagrama de bloques asociado al controlador booleano.

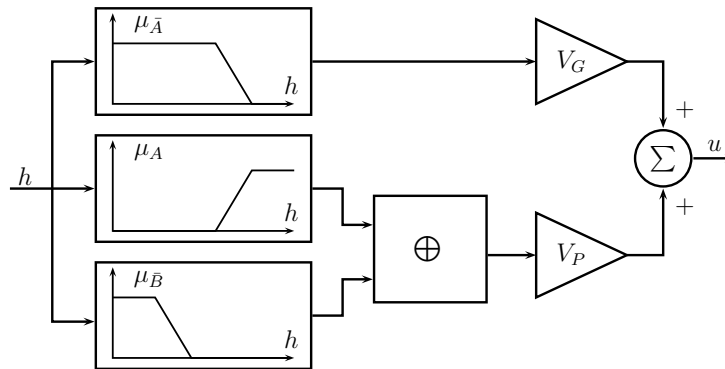


Figura 8.16: Diagrama de bloques para el controlador BBR.

8.5.6. Resultados

Los resultados corresponden, en primer lugar, a la superficie de control (curva en este caso) de la implementación del controlador booleano y BBR. Estas curvas se pueden

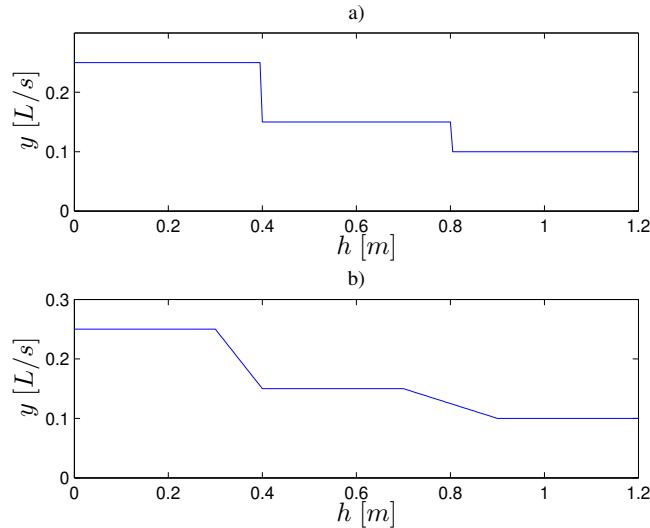


Figura 8.17: Curva de transferencia para controlador: a) Booleana, b) BBR.

observar en la figura 8.17. Otro resultado es la respuesta dinámica del sistema para los dos controladores; esto se puede ver en la figura 8.18.

Del mismo modo, la figura 8.17 presenta la curva característica de transferencia del controlador BBR, que muestra un comportamiento monótono entre las acciones booleanas de acuerdo con el diseño.

En la figura 8.18 se observa el comportamiento de conmutación típico asociado al controlador booleano, por su parte, el controlador BBR muestra un comportamiento más suave. Un detalle de estas características se muestra en la figura 8.19, donde se aprecia un comportamiento oscilatorio para el controlador booleano, mientras que el controlador BBR presenta un comportamiento continuo. En sistemas de control donde el actuador es sensible a cambios repentinos, este es un factor de importancia. De acuerdo con [225], el comportamiento oscilatorio se asocia con un ciclo límite, lo cual es común en los sistemas de control ON-OFF.

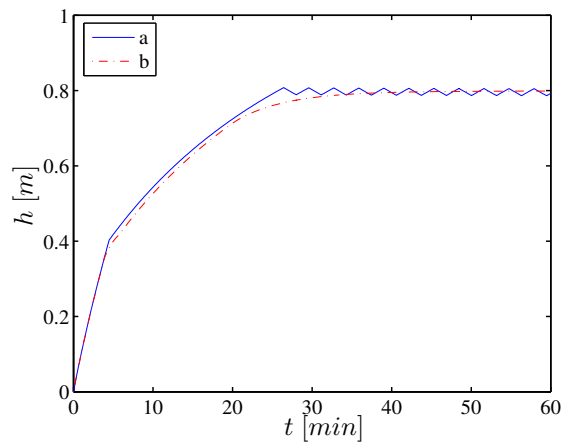


Figura 8.18: Respuestas del sistema para una referencia de 80cm: a) Booleano, b) BBR.

Adicionalmente, la figura 8.20 muestra las acciones de control para los diseños booleano y difuso. El controlador difuso exhibe transiciones continuas en comparación con el booleano. La implementación de un controlador booleano utiliza sensores y actuadores ON-OFF; sin embargo, al realizar la implementación difusa, se requiere que estos sensores y actuadores tengan un comportamiento continuo de acuerdo con el diseño.

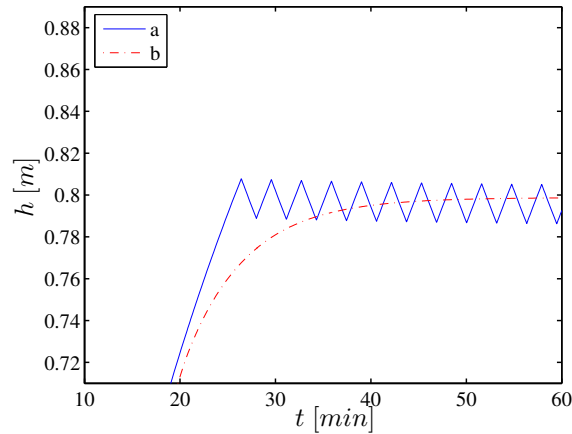


Figura 8.19: Detalle de las respuestas: a) Booleano, b) BBR.

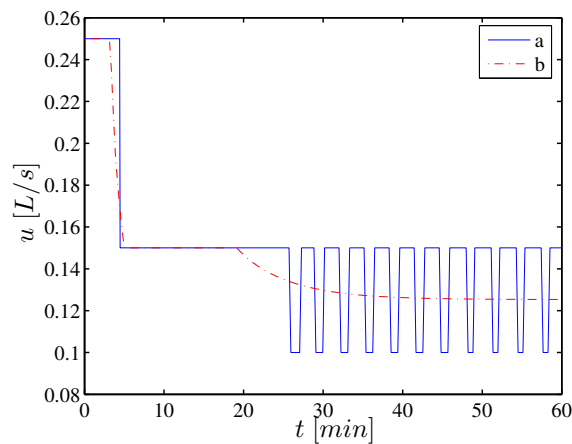


Figura 8.20: Acciones de control: a) Booleano, b) BBR.

Los indicadores de rendimiento del sistema están contenidos en la tabla 8.11, siendo $\%e_{ss}$ el error de estado estable, T_s tiempo de establecimiento y finalmente la característica de oscilación. El valor T_s se calcula cuando la señal está dentro de un rango con 5% de la referencia. La referencia considerada para las simulaciones es 80cm. En la tabla 8.11 se observa que el menor error de estado estable es para el sistema BBR; el T_s más bajo se logra usando el controlador booleano y no se presenta oscilación para el sistema BBR.

Característica \ Control	Booleano	BBR
$\%e_{ss}$	0,82%	0,25%
T_s	25,5min	31,5min
Oscilación	Si	No

Tabla 8.11: Indicadores del desempeño sistema.

8.6. Simplificación de términos en forma normal disyuntiva utilizando álgebra de Kleene

Un aspecto de importancia en la metodología propuesta para el diseño e implementación de los sistemas de inferencia difusa basados en relaciones booleanas, consiste en la manipulación que se puede realizar con el álgebra de Kleene, para esto se debe tener presente los operadores de disyunción, conjunción y complemento en el álgebra \mathcal{K} , los cuales se pueden apreciar en la tabla 8.12. Adicionalmente, es necesario considerar el mecanismo para establecer la Forma Normal Disyuntiva (FND) utilizada en la implementación de los FIS-BBR.

\vee	0	u	1	\wedge	0	u	1	$\bar{}$	
0	0	u	1	0	0	0	0	0	1
u	u	u	1	u	0	u	u	u	u
1	1	1	1	1	0	u	1	1	0

Tabla 8.12: Tabla de verdad para la disyunción, conjunción y complemento en álgebra \mathcal{K} .

Un elemento x se dice \vee -irreducible (*join-irreducible*) si es distinto de 0 y no puede ser escrito como la disyunción de elementos estrictamente menores, y se dice \wedge -irreducible (*meet-irreducible*) si es distinto de 1 y no puede ser escrito como la conjunción de elementos estrictamente mayores [187].

La forma normal disyuntiva para un elemento a es una representación del tipo $a_0 \vee a_1 \vee \dots \vee a_l$, donde cada a_i es incomparable con cada a_j , ($0 \leq i, j \leq l$) si $i \neq j$ y cada a_i es \vee -irreducible.

Se ha demostrado que las formas normales disyuntiva y conjuntiva existen y son únicas para el caso de las álgebras \mathcal{K} y \mathcal{B} [185, 186, 187].

8.6.1. Forma normal disyuntiva obtenida a partir de tablas de Kleene

Las siguientes tres reglas se utilizan para obtener la forma normal disyuntiva \mathcal{K} de una fórmula f a partir de una tabla [187].

- Para las filas que tienen valor 1 en la columna de f : Se forma la conjunción de las variables que tienen valor 1 con los complementos de las variables que tienen valor 0 y la constante 1 para las variables que tienen valor u .
- Para las filas que tienen valor u en la columna de f : Se forma la conjunción de las variables que tienen valor 1 con los complementos de las variables que tienen valor 0 y la conjunción, tanto de la variable como de su complemento, para las que tienen valor u .
- Se construye la disyunción de las conjunciones obtenidas con las dos reglas anteriores. En caso de no existir conjunciones se obtendrá una disyunción vacía ya que representa la forma normal disyuntiva para 0.

Un ejemplo de la FND se puede apreciar en la tabla 8.13 donde se tienen las variables A_1, A_2 las cuales pueden ser $\{0, u, 1\}$ hasta completar toda la tabla de verdad de Kleene. Al lado derecho de la tabla se presentan los respectivos términos de las variables para construir la forma normal disyuntiva de f .

A_1	A_2	f	FND
0	0	0	-
0	u	u	$\bar{A}_1 \wedge (A_2 \wedge \bar{A}_2)$
0	1	1	$\bar{A}_1 \wedge A_2$
u	0	u	$(A_1 \wedge \bar{A}_1) \wedge \bar{A}_2$
u	u	u	$(A_1 \wedge \bar{A}_1) \wedge (A_2 \wedge \bar{A}_2)$
u	1	1	$1 \wedge A_2$
1	0	1	$A_1 \wedge \bar{A}_2$
1	u	1	$A_1 \wedge 1$
1	1	1	$A_1 \wedge A_2$

Tabla 8.13: Ejemplo de términos de la forma normal disyuntiva.

La ecuación asociada a la FND de la tabla 8.13 es:

$$f = [\bar{A}_1 \wedge A_2 \wedge \bar{A}_2] \vee [\bar{A}_1 \wedge A_2] \vee [A_1 \wedge \bar{A}_1 \wedge \bar{A}_2] \vee [A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2] \vee A_2 \vee [A_1 \wedge \bar{A}_2] \vee A_1 \vee [A_1 \wedge A_2] \quad (8.23)$$

Para expresiones obtenidas en formas normales aplicando la propiedad distributiva junto con la propiedad de absorción en \mathcal{K} , se puede reducir el número de literales y operadores. Como ejemplo, la tabla 8.14 muestra una posible codificación para la función Y .

La respectiva ecuación para Y de la tabla 8.14 es:

$$Y_1 = (A_1 \wedge A_2) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2) \vee (A_1 \wedge \bar{A}_1 \wedge A_2) \vee (A_1 \wedge A_2 \wedge \bar{A}_2) \quad (8.24)$$

Usando la propiedad distributiva:

$$Y_1 = (A_1 \wedge A_2) \wedge [1 \vee (\bar{A}_1 \wedge \bar{A}_2) \vee \bar{A}_1 \vee \bar{A}_2] \quad (8.25)$$

A_1	A_2	Y
1	1	1
u	u	u
0	0	0
0	u	0
0	1	0
u	1	u
1	0	0
1	u	u
u	0	0

Tabla 8.14: Ejemplo de codificación para una tabla de Kleene.

Entonces la ecuación simplificada para Y es:

$$Y_1 = A_1 \wedge A_2 \quad (8.26)$$

8.7. Sistemas difusos compactos basados en relaciones booleanas

Los sistemas difusos compactos basados en relaciones booleanas se presentan cuando en el consecuente la función de activación tiene asociada una sola regla de inferencia. En primer lugar, de esta forma se elimina la operación de disyunción. Por otra parte, aplicando las respectivas simplificaciones mediante el álgebra de Kleene se puede tener una expresión compacta equivalente a la simplificación realizada con álgebra de Boole.

En las tablas 8.15, 8.16 y 8.17 se muestran ejemplos de estas posibles codificaciones para el caso booleano, donde las casillas marcadas con X indican que la variable Booleana puede ser 1 o 0.

A_1	\dots	A_m	\dots	A_{n-1}	A_n	Y_U
X	\dots	1	\dots	X	X	1
X	\dots	0	\dots	X	X	0

Tabla 8.15: Codificación para la activación con un conjunto en la entrada.

A_1	\dots	A_m	A_p	\dots	A_{n-1}	A_n	Y_P
X	\dots	1	1	\dots	X	X	1
X	\dots	0	X	\dots	X	X	0
X	\dots	X	0	\dots	X	X	0

Tabla 8.16: Codificación para la activación parcial con dos conjuntos de la entrada.

A_1	A_2	\dots	A_{n-1}	A_n	Y_T
1	1	\dots	1	1	1
0	X	\dots	X	X	0
X	0	\dots	X	X	0
X	X	\dots	0	X	0
X	X	\dots	X	0	0

Tabla 8.17: Codificación para activación con todos los conjuntos en la entrada.

En la tabla 8.15, para la función Y_U se tiene la acción cuando A_1 es “activa”, es decir, que sólo depende de un conjunto de la entrada. Para Y_P de la tabla 8.16 se busca su activación con la acción de A_1 y A_2 , dependiendo parcialmente de los conjuntos de la entrada. Por su parte, en la tabla 8.17 para Y_T se tiene una acción cuando se activan simultáneamente A_1, A_2, \dots, A_n , por lo cual, depende de todos los conjuntos de la entrada. Considerando lo anterior estas funciones se llaman:

- Y_U : Función de activación con dependencia unitaria.
- Y_P : Función de activación con dependencia parcial.
- Y_T : Función de activación con dependencia total.

Con la finalidad de establecer las ecuaciones generales para los casos en consideración, se revisa el proceso para obtener estas expresiones. En primer lugar, se tiene la descripción de estas funciones en la forma normal disyuntiva y las simplificaciones que se pueden realizar. Posteriormente, se observa la equivalencia que se puede tener de las simplificaciones con las tablas de Boole y Kleene. Finalmente, se propone la generalización de esta simplificación.

8.7.1. Generación de términos en forma normal disyuntiva

Cuando en una tabla de verdad booleana la función de salida es 1 se presentan todos los posibles términos para la forma normal disyuntiva. Como un primer caso, considerando una función booleana tal que $f = 1$, entonces:

$$f_1 = 1 = (A_1 \vee \bar{A}_1) \quad (8.27)$$

De tal forma que se tiene una variable, y dos términos para f . De la misma manera, para dos variables se tienen cuatro términos:

$$f_2 = (A_1 \vee \bar{A}_1) \wedge (A_2 \vee \bar{A}_2) \quad (8.28)$$

$$f_2 = (A_1 \wedge A_2) \vee (A_1 \wedge \bar{A}_2) \vee (\bar{A}_1 \wedge A_2) \vee (\bar{A}_1 \wedge \bar{A}_2) \quad (8.29)$$

Por su parte, para tres variables se presentan ocho términos de la siguiente forma:

$$f_3 = (A_1 \vee \bar{A}_1) \wedge (A_2 \vee \bar{A}_2) \wedge (A_3 \vee \bar{A}_3) \quad (8.30)$$

$$\begin{aligned}
f_3 = & (A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge \bar{A}_2 \wedge A_3) \vee \\
& (\bar{A}_1 \wedge A_2 \wedge A_3) \vee (\bar{A}_1 \vee \bar{A}_2 \wedge A_3) \vee \\
& (A_1 \wedge A_2 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee \\
& (\bar{A}_1 \wedge A_2 \wedge \bar{A}_3) \vee (\bar{A}_1 \vee \bar{A}_2 \wedge \bar{A}_3)
\end{aligned} \tag{8.31}$$

Generalizando para n variables se tendrían 2^n términos.

$$f_n = (A_1 \vee \bar{A}_1) \wedge \cdots \wedge (A_{n-1} \vee \bar{A}_{n-1}) \wedge (A_n \vee \bar{A}_n) \tag{8.32}$$

Como es de apreciar, este es el principio utilizado para la simplificación de términos utilizando álgebra de Boole.

8.7.2. Expresiones con álgebra de Boole para FIS-BBR compactos

En primer lugar, considerando el proceso para tener todos los términos de la forma normal disyuntiva, y en segundo lugar empleando álgebra de Boole de la tabla 8.15 la expresión para Y_U de manera general correspondería a:

$$Y_U = (A_1 \vee \bar{A}_1) \wedge (A_2 \vee \bar{A}_2) \wedge \cdots \wedge A_m \wedge \cdots \wedge (A_{n-1} \vee \bar{A}_{n-1}) \wedge (A_n \vee \bar{A}_n) \tag{8.33}$$

$$Y_U = A_m \tag{8.34}$$

De la misma forma, tomando la tabla 8.16 la expresión para Y_P es:

$$Y_P = (A_1 \vee \bar{A}_1) \wedge (A_2 \vee \bar{A}_2) \wedge \cdots \wedge A_m \wedge A_p \wedge \cdots \wedge (A_{n-1} \vee \bar{A}_{n-1}) \wedge (A_n \vee \bar{A}_n) \tag{8.35}$$

$$Y_P = A_m \wedge A_p \tag{8.36}$$

Finamente, con la tabla 8.17 la expresión para Y_T corresponde a:

$$Y_T = A_1 \wedge A_2 \wedge \cdots \wedge A_{n-1} \wedge A_n \tag{8.37}$$

8.8. Extensión a la tabla de Kleene

Para realizar la extensión de una tabla booleana a una kleenena para los sistemas compactos se debe tener presente que los casos marcados con X pueden ser 1, 0 o u . Para cumplir con las condiciones de regularidad, y buscando transiciones monótonas entre acciones booleanas se tienen presente las siguientes consideraciones:

- Los casos que se adicionan donde no hay transición para $Y = 1$ o cuando $Y = 0$ deben mantener esta salida.
- Los casos adicionales donde existe una transición entre $Y = 1$ y $Y = 0$ (o al contrario) a la salida se asigna u .

8.8.1. Función de activación con dependencia unitaria

En el caso de una función de activación con dependencia unitaria, la tabla 8.18 muestra la extensión de la tabla de Boole a Kleene donde X puede ser $\{1, u, 0\}$.

A_1	\dots	A_m	\dots	A_{n-1}	A_n	Y_U
X	\dots	1	\dots	X	X	1
X	\dots	u	\dots	X	X	u
X	\dots	0	\dots	X	X	0

Tabla 8.18: Extensión a Kleene de la codificación para la activación con un conjunto en la entrada.

En esta extensión, al adicionar la variable u para tener una tabla regular y mantener una transición monótona, entonces Y_U es 1 para cualquier caso que $A_m = 1$, corresponderá a u cuando $A_m = u$ y 0 para $A_m = 0$. Como es de apreciar, los casos donde existe transición para Y_U entre 0 y 1 se les asigna u .

Considerando las reglas para establecer la forma normal disyuntiva (ver sección 8.6.1) se aprecia que A_m es común para los términos que se obtienen cuando Y_U es 1 o u , por lo cual la expresión general se puede escribir como:

$$Y_U = (A_m \wedge F_B) \vee (A_m \wedge F_E) \vee A_m = A_m \quad (8.38)$$

Donde:

- F_B : corresponde a la disyunción de todas las conjunciones obtenidas en la tabla de verdad, para las cuales f_U es 1 eliminando de estas la variable A_m .
- F_E : corresponde a la disyunción de todas las conjunciones obtenidas en la tabla de verdad, para las cuales f_U es u eliminando de estas la variable A_m .

Como es de apreciar, el aspecto clave en este resultado ocurre cuando al calcular la forma normal disyuntiva la variable A_m aparece en todos los términos, por lo cual se puede realizar un proceso de absorción. Para todas las conjunciones cuando f_U es 1 siempre se presenta A_m y cuando f_U es u siempre se tiene la conjunción $A_m \wedge \bar{A}_m$.

Como un primer ejemplo para mostrar lo anterior se considera la tabla de verdad booleana 8.19, donde f_U solamente depende de A_1 .

A_1	A_2	f_U
0	0	0
0	1	0
1	0	1
1	1	1

Tabla 8.19: Relaciones de Boole para dos variables.

La respectiva simplificación mediante álgebra de Boole es:

$$f_U = (A_1 \wedge A_2) \vee (A_1 \wedge \bar{A}_2) = A_1 \wedge (A_2 \vee \bar{A}_2) = A_1 \quad (8.39)$$

Al emplear álgebra de Kleene, el anterior procedimiento no se cumple ya que $A_2 \vee \bar{A}_2 \leq 1$, sin embargo, al incluir la variable u empleando la configuración de la tabla 8.20 se logra que esta sea regular, y también se tiene una equivalencia con la fórmula obtenida con álgebra de Boole.

A_1	A_2	f_U
0	0	0
0	u	0
0	1	0
1	1	1
1	0	1
1	u	1
u	0	u
u	1	u
u	u	u

Tabla 8.20: Relaciones de Kleene para dos variables.

La tabla 8.21 muestra otra forma para representar la tabla extendida.

\mathcal{B}		A_1	
		0	1
A_2	0	0	1
	1	0	1

\mathcal{K}		A_1		
		0	u	1
A_2	0	0	u	1
	u	0	u	1
	1	0	u	1

Tabla 8.21: Codificación de Kleene alternativa.

Aplicando álgebra de Kleene se tiene:

$$\begin{aligned}
 f_U &= (A_1 \wedge A_2) \vee (A_1 \wedge \bar{A}_2) \vee A_1 \\
 &\quad \vee (A_1 \wedge \bar{A}_1 \wedge \bar{A}_2) \vee (A_1 \wedge \bar{A}_1 \wedge A_2) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2) \\
 &= A_1 \wedge (A_2 \vee \bar{A}_2) \vee A_1 \vee A_1 [(A_1 \wedge \bar{A}_2) \vee (A_1 \wedge A_2) \vee (A_1 \wedge A_2 \wedge \bar{A}_2)] \\
 &= A_1
 \end{aligned} \quad (8.40)$$

Esta expresión se puede escribir como:

$$f_U = A_1 \wedge F_B(A_2, \bar{A}_2) \vee A_1 \wedge F_E(\bar{A}_1, A_2, \bar{A}_2) \vee A_1 = A_1 \quad (8.41)$$

Con la finalidad de continuar mostrando este caso se considera la tabla de verdad 8.22 de tres variables, donde f_U solamente dependería de A_1 .

A_1	A_2	A_3	f_U
0	X	X	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Tabla 8.22: Relaciones de Boole para tres variables.

La respectiva simplificación mediante álgebra de Boole es:

$$\begin{aligned}
f_U &= (A_1 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (A_1 \wedge A_2 \wedge \bar{A}_3) \vee \\
&\quad (A_1 \wedge \bar{A}_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge A_3) \\
&= A_1 \wedge [(\bar{A}_2 \wedge \bar{A}_3) \vee (A_2 \wedge \bar{A}_3) \vee (\bar{A}_2 \wedge A_3) \vee (A_2 \wedge A_3)] \\
&= A_1 \wedge [(\bar{A}_3 \wedge (\bar{A}_2 \vee A_2)) \vee (A_3 \wedge (\bar{A}_2 \vee A_2))] \\
&= A_1 \wedge (\bar{A}_3 \vee A_3) \wedge (\bar{A}_2 \vee A_2) \\
&= A_1
\end{aligned} \tag{8.42}$$

Considerando el tercer elemento u para construir la tabla de Kleene se tiene la configuración mostrada en la tabla 8.23.

A_1	A_2	A_3	f_U	A_1	A_2	A_3	f_U	A_1	A_2	A_3	f_U
1	0	0	1	u	0	0	u	0	0	0	0
1	0	1	1	u	0	1	u	0	0	1	0
1	1	0	1	u	1	0	u	0	1	0	0
1	1	1	1	u	1	1	u	0	1	1	0
1	0	u	1	u	0	u	u	0	0	u	0
1	1	u	1	u	1	u	u	0	1	u	0
1	u	0	1	u	u	0	u	0	u	0	0
1	u	1	1	u	u	1	u	0	u	1	0
1	u	u	1	u	u	u	u	0	u	u	0

Tabla 8.23: Relaciones de Kleene para tres variables.

De forma alternativa la relación entre las variables se puede mostrar mediante el siguiente conjunto de tablas:

A_3	A_1			A_3	A_1			A_3	A_1		
	0	u	1		0	u	1		0	u	1
0	0	0	1	u	0	0	1	1	0	0	1
A_2	0	0	1	A_2	0	0	1	A_2	0	0	1
	u	0	1		u	0	1		u	0	1
	1	0	1		1	0	1		1	0	1

Tabla 8.24: Relaciones de tres variables para f_U .

Aplicando álgebra de Kleene para f_U se tiene:

$$\begin{aligned}
f_U &= (A_1 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge \bar{A}_3) \vee (A_1 \wedge A_2 \wedge A_3) \vee \\
&\quad (A_1 \wedge \bar{A}_2) \vee (A_1 \wedge A_2) \vee (A_1 \wedge \bar{A}_3) \vee (A_1 \wedge A_3) \vee A_1 \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_1 \wedge \bar{A}_2 \wedge A_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge \bar{A}_1 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge A_3 \wedge \bar{A}_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \\
f_U &= A_1 \wedge [(\bar{A}_2 \wedge \bar{A}_3) \vee (\bar{A}_2 \wedge A_3) \vee (A_2 \wedge \bar{A}_3) \vee (A_2 \wedge A_3) \vee \bar{A}_2 \vee A_2 \vee \bar{A}_3 \vee A_3] \vee A_1 \vee \\
&\quad A_1 \wedge [(\bar{A}_1 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge \bar{A}_2 \wedge A_3) \vee (\bar{A}_1 \wedge A_2 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge A_2 \wedge A_3) \vee \\
&\quad (\bar{A}_1 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge A_2 \wedge A_3 \wedge \bar{A}_3) \vee \\
&\quad (\bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3) \vee (\bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3)] \quad (8.43)
\end{aligned}$$

Para este caso, la expresión se puede escribir como:

$$f_U = (A_1 \wedge F_B) \vee (A_1 \wedge F_E) \vee A_1 = A_1 \quad (8.44)$$

8.8.2. Función de activación con dependencia parcial

Para el caso de una función de activación con dependencia parcial, la tabla 8.25 muestra la extensión de la tabla de Boole a Kleene donde X puede ser $\{1, u, 0\}$ y $X_{u,1}$ los elementos $\{u, 1\}$.

A_1	\dots	A_m	A_p	\dots	A_{n-1}	A_n	Y_P
X	\dots	1	1	\dots	X	X	1
X	\dots	u	$X_{u,1}$	\dots	X	X	u
X	\dots	$X_{u,1}$	u	\dots	X	X	u
X	\dots	0	X	\dots	X	X	0
X	\dots	X	0	\dots	X	X	0

Tabla 8.25: Extensión a Kleene de la codificación para la activación con dos conjuntos en la entrada.

En esta tabla, si cualquiera de las variables A_m y A_p son 0 entonces $Y_P = 0$; por su parte si A_m y A_p son 1 entonces $Y_P = 1$ y finalmente en el caso que cualquiera de las variables A_m y A_p sean u o 1 entonces $Y_P = u$. Como es de apreciar, en los casos donde existe transición para Y_P entre 0 y 1 se les asigna u .

Teniendo presente las reglas para establecer la forma normal disyuntiva (sección 8.6.1), es de observar que el término $A_m \wedge A_p$ es común para las expresiones que se tienen cuando Y_P es 1 o u , por lo tanto, la ecuación para Y_P se puede escribir como:

$$Y_P = (A_m \wedge A_p \wedge F_B) \vee (A_m \wedge A_p \wedge F_E) \vee (A_m \wedge A_p) = A_m \wedge A_p \quad (8.45)$$

Donde:

- F_B : corresponde a la disyunción de todas las conjunciones obtenidas en la tabla de verdad para las cuales Y_P es 1 eliminando de estas la conjunción $A_m \wedge A_p$.
- F_E : corresponde a la disyunción de todas las conjunciones obtenidas en la tabla de verdad para las cuales Y_P es u eliminando de estas la conjunción $A_m \wedge A_p$.

El aspecto de importancia en este caso es que el término $A_m \wedge A_p$ es común a todos los términos que se generan en la ecuación para Y_P , lo cual permite aplicar la propiedad de absorción.

Con el fin de mostrar este caso, se considera la tabla de verdad de tres variables 8.26, donde f_P depende de A_1 y A_2 . En esta tabla X puede ser $\{0, 1\}$ hasta obtener todos los posibles casos de la tabla de verdad.

A_1	A_2	A_3	f_P
0	X	X	0
X	0	X	0
1	1	X	1

Tabla 8.26: Relaciones de Boole para tres variables.

La respectiva simplificación booleana es:

$$f_P = A_1 \wedge A_2 \quad (8.46)$$

Considerando el tercer elemento u para construir la tabla de Kleene se tiene la configuración mostrada en la tabla 8.27.

A_1	A_2	A_3	f_U	A_1	A_2	A_3	f_U	A_1	A_2	A_3	f_U
1	0	0	0	u	0	0	0	0	0	0	0
1	0	1	0	u	0	1	0	0	0	1	0
1	1	0	1	u	1	0	u	0	1	0	0
1	1	1	1	u	1	1	u	0	1	1	0
1	0	u	0	u	0	u	0	0	0	u	0
1	1	u	1	u	1	u	u	0	1	u	0
1	u	0	u	u	u	0	u	0	u	0	0
1	u	1	u	u	u	1	u	0	u	1	0
1	u	u	u	u	u	u	u	0	u	u	0

Tabla 8.27: Relaciones de Kleene para tres variables.

De forma alternativa, la relación entre las variables se puede mostrar mediante el siguiente conjunto de tablas:

A_3				A_1				A_3				A_1				A_3				A_1						
0				0	u	1				u				0	u	1				1				0	u	1
A_2		0	0	0	0	A_2		0	0	0	0	A_2		0	0	0	0	A_2		0	0	0	0			
		u	0	u	u			u	0	u	u			u	0	u	u			u	0	u	u			
		1	0	u	1			1	0	u	1			1	0	u	1			1	0	u	1			

Tabla 8.28: Relaciones para tres variables para f_U .

Aplicando álgebra de Kleene para f_P se tiene:

$$\begin{aligned}
f_P &= (A_1 \wedge A_2 \wedge \bar{A}_3) \vee (A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge A_2) \vee \\
&\quad (A_1 \wedge A_2 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (A_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge A_3 \wedge \bar{A}_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \\
f_P &= (A_1 \wedge A_2) \wedge [\bar{A}_3 \vee A_3] \vee (A_1 \wedge A_2) \\
&\quad (A_1 \wedge A_2) \wedge [(\bar{A}_2 \wedge \bar{A}_3) \vee (\bar{A}_2 \wedge A_3) \vee (\bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge \bar{A}_3) \vee \\
&\quad (\bar{A}_1 \wedge A_3) \vee (\bar{A}_1 \wedge A_3 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge \bar{A}_2 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge \bar{A}_2 \wedge A_3) \vee \\
&\quad (\bar{A}_1 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3)] \tag{8.47}
\end{aligned}$$

En este caso la expresión se puede escribir como:

$$f_P = (A_1 \wedge A_2 \wedge F_B) \vee (A_1 \wedge A_2 \wedge F_E) \vee (A_1 \wedge A_2) \tag{8.48}$$

por lo cual:

$$f_P = A_1 \wedge A_2 \tag{8.49}$$

8.8.3. Función de activación con dependencia total

Para este caso se realiza la activación mediante todos los conjuntos de entrada. La tabla 8.25 muestra la extensión de la tabla de Boole a Kleene donde X puede ser $\{1, u, 0\}$ y $X_{u,1}$ los elementos $\{u, 1\}$.

En esta tabla, si cualquiera de las variables A_m es 0, entonces, $Y_T = 0$, por su parte, si todas las variables A_m son 1, entonces, $Y_T = 1$ y finalmente, en el caso que cualquiera de las variables A_m sea u y las otras u o 1, entonces, $Y_T = u$. Como es de apreciar, para los casos donde existe transición de Y_T entre 0 y 1 se les asigna u .

De nuevo, considerando las reglas para establecer la forma normal disyuntiva (sección 8.6.1) es de observar que $A_1 \wedge A_m \wedge \dots \wedge A_n$ es común para las expresiones que se establecen cuando Y_P es 1 o u , por lo tanto, de forma general Y_T se puede escribir como:

$$Y_T = (A_1 \wedge A_m \wedge \dots \wedge A_n) \vee (A_1 \wedge A_m \wedge \dots \wedge A_n \wedge F_E) \tag{8.50}$$

$$Y_T = A_1 \wedge A_m \wedge \dots \wedge A_n \tag{8.51}$$

A_1	A_2	\dots	A_{n-1}	A_n	Y_T
1	1	\dots	1	1	1
u	$X_{u,1}$	\dots	$X_{u,1}$	$X_{u,1}$	u
$X_{u,1}$	u	\dots	$X_{u,1}$	$X_{u,1}$	u
$X_{u,1}$	$X_{u,1}$	\dots	u	$X_{u,1}$	u
$X_{u,1}$	$X_{u,1}$	\dots	$X_{u,1}$	u	u
0	X	\dots	X	X	0
X	0	\dots	X	X	0
X	X	\dots	0	X	0
X	X	\dots	X	0	0

Tabla 8.29: Extensión a Kleene de la codificación para la activación con todos los conjuntos en la entrada.

Donde F_E corresponde a la disyunción de todas las conjunciones obtenidas en la tabla de verdad para las cuales Y_T es u eliminando de estas la conjunción $A_1 \wedge A_m \wedge \dots \wedge A_n$.

Es importante notar que $A_1 \wedge A_m \wedge \dots \wedge A_n$ se presenta en los términos calculados para Y_T cuando es u . De esta forma al realizarse el proceso de absorción se obtiene la expresión para Y_T .

Para mostrar el caso de la función de activación Y_T se considera la tabla de verdad 8.30, donde X puede ser $\{0, 1\}$ hasta completar todos los casos.

A_1	A_2	A_3	f_T
0	X	X	0
X	0	X	0
X	X	0	0
1	1	1	1

Tabla 8.30: Relaciones de Boole para tres variables.

La respectiva expresión empleando álgebra booleana es:

$$f_T = A_1 \wedge A_2 \wedge A_3 \quad (8.52)$$

Como es de apreciar en la tabla 8.30, para cualquier valor de X que puede ser 0 o 1 en la columna de f_T se tiene asignado un 0.

Considerando el tercer elemento u para construir la tabla de Kleene se tiene la configuración mostrada en la tabla 8.31.

A_1	A_2	A_3	f_T	A_1	A_2	A_3	f_T	A_1	A_2	A_3	f_T
1	0	0	0	u	0	0	0	0	0	0	0
1	0	1	0	u	0	1	0	0	0	1	0
1	1	0	0	u	1	0	0	0	1	0	0
1	1	1	1	u	1	1	u	0	1	1	0
1	0	u	0	u	0	u	0	0	0	u	0
1	1	u	u	u	1	u	u	0	1	u	0
1	u	0	0	u	u	0	0	0	u	0	0
1	u	1	u	u	u	1	u	0	u	1	0
1	u	u	u	u	u	u	u	0	u	u	0

Tabla 8.31: Relaciones de Kleene para tres variables.

De forma alternativa, la relación entre las variables se puede mostrar mediante el siguiente conjunto de tablas:

A_3		A_1			A_3		A_1			A_3		A_1			
0		0	u	1	u		0	u	1	1		0	u	1	
A_2	0	0	0	0	A_2	0	0	0	0	A_2	0	0	0	0	
	u	0	0	0		u	0	u	u		u	0	u	u	u
	1	0	0	0		1	0	u	u		1	0	u	1	1

Tabla 8.32: Relaciones para tres variables para f_T .

Aplicando álgebra de Kleene para f_T se obtiene:

$$\begin{aligned}
f_T &= (A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge A_3 \wedge \bar{A}_3) \vee (A_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3) \vee \\
&\quad (A_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge A_3 \wedge \bar{A}_3) \vee (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3) \vee \\
&\quad (A_1 \wedge \bar{A}_1 \wedge A_2 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_3) \\
f_T &= (A_1 \wedge A_2 \wedge A_3) \vee [(A_1 \wedge A_2 \wedge A_3) \wedge [\bar{A}_3 \vee \bar{A}_2 \vee (\bar{A}_2 \wedge \bar{A}_3) \vee \\
&\quad \bar{A}_1 \vee (\bar{A}_1 \wedge \bar{A}_3) \vee (\bar{A}_1 \wedge \bar{A}_2) \vee (\bar{A}_1 \wedge \bar{A}_2 \wedge \bar{A}_3)]]
\end{aligned} \tag{8.53}$$

En este caso la expresión se puede escribir como:

$$f_T = (A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge A_3 \wedge F_E) \tag{8.54}$$

por lo cual

$$f_T = A_1 \wedge A_2 \wedge A_3 \tag{8.55}$$

8.9. Arquitecturas de sistemas difusos compactos basados en relaciones booleanas

Para el desarrollo de los sistemas difusos compactos basados en relaciones booleanas se consideran las siguientes configuraciones de las funciones de activación:

- Arquitectura I: Empleando funciones de activación con dependencia total, se realiza la activación de cada actuador virtual mediante todos los conjuntos de entrada. Luego se organizan las salidas virtuales para establecer la estructura del sistema compacto.
- Arquitectura II: Utilizando funciones de activación con dependencia unitaria, se efectúa la activación de cada actuador virtual mediante un sólo conjunto de entrada. Posteriormente se organizan las salidas virtuales para determinar la estructura del sistema compacto.

8.9.1. Sistema compacto con arquitectura I

Con el fin de establecer un sistema con la arquitectura I se propone la codificación mostrada en la tabla 8.33.

$A_{1m}(x_1)$	$A_{2m}(x_2)$	\dots	$A_{pm}(x_p)$	\dots	$A_{Nm}(x_N)$	Y_m
1	1	\dots	1	\dots	1	1
0	X	\dots	X	\dots	X	0
X	0	\dots	X	\dots	X	0
X	X	\dots	0	\dots	X	0
X	X	\dots	X	\dots	0	0

Tabla 8.33: Codificación para sistemas compactos con arquitectura I.

De esta forma:

$$Y_m = A_{1m}(x_1) \wedge A_{2m}(x_2) \wedge \dots \wedge A_{pm}(x_p) \wedge \dots \wedge A_{Nm}(x_N) \quad (8.56)$$

Para la implementación empleando conjuntos difusos, la conjunción se realiza con la t-norma del producto y cada conjunto A_{pm} tiene asociada una función de pertenencia μ_{pm} , con $p = 1, 2, \dots, N$.

Tomando $m = 1$, es decir una salida virtual, este proceso de inferencia se puede apreciar en la figura 8.21; donde v_1 corresponde al respectivo actuador virtual.

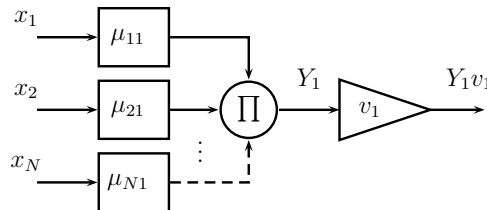


Figura 8.21: Salida virtual parcial del proceso de inferencia.

Utilizando M actuadores virtuales con sus respectivas funciones de activación se tiene el esquema mostrado en la figura 8.22. En este caso para complementar el diseño se utiliza

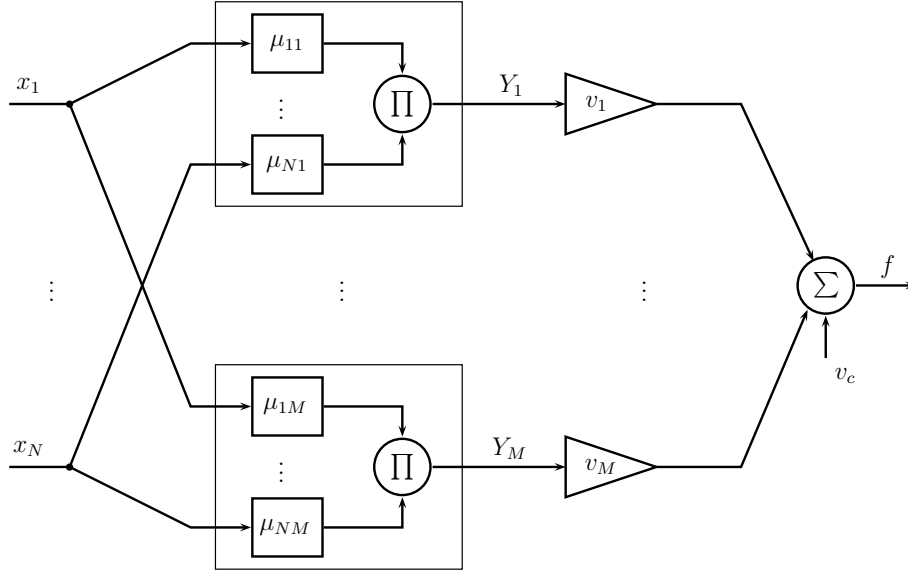


Figura 8.22: Representación de un sistema compacto con arquitectura I.

un actuador virtual constante v_c el cual correspondería a una codificación con una función de activación Y_c siempre activa, es decir, con valores de 1 en su columna.

De esta manera la expresión para el proceso de inferencia de este sistema es:

$$f = v_c + \sum_{m=1}^M v_m \left(\prod_{p=1}^N \mu_{pm}(x_p) \right) \quad (8.57)$$

Este esquema puede ser utilizado en la identificación de sistemas ya que presenta una estructura similar a la empleada convencionalmente en redes neuronales de base radial al utilizarse para μ_{pm} funciones de pertenencia gaussianas [226]. Tal como se aprecia en la figura 8.22, al tener este sistema difuso un esquema compacto, puede ser interpretado como una red neuronal, permitiendo la adaptabilidad de los actuadores virtuales y de los parámetros de las funciones de pertenencia.

8.9.2. Sistema compacto con arquitectura II

En este caso, para tener un sistema difuso donde las funciones de activación dependen de un solo conjunto de la entrada, se propone la codificación mostrada en la tabla 8.34.

$A_{11}(x_1)$	$A_{12}(x_1)$	\dots	$A_{1j}(x_1)$	\dots	$A_{1N}(x_1)$	Y_{11}
1	X	\dots	X	\dots	X	1
0	X	\dots	X	\dots	X	0

Tabla 8.34: Codificación parcial para un sistema compacto con arquitectura II.

Para este caso se tiene que $Y_{11} = A_{11}(x_1)$, de forma general, tomando $j = 1, 2, \dots, N$ y

también incrementando las salidas de activación que dependen de x_1 se tiene:

$$Y_{1j} = A_{1j}(x_1) \quad (8.58)$$

Teniendo cada conjunto A_{1j} una función de pertenencia asociada μ_{1j} y con actuadores virtuales v_{1j} , se tiene una salida parcial f_1 como la mostrada en la figura 8.23.

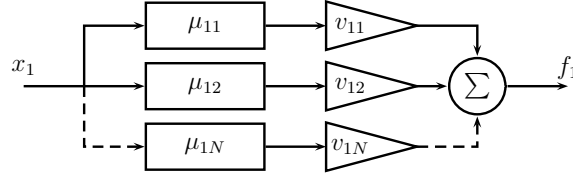


Figura 8.23: Ejemplo de la configuración para una salida parcial.

Generalizando la tabla 8.34 para más variables de entrada $x_1, x_2, \dots, x_i, \dots, x_M$, se puede tener la codificación de la tabla 8.35.

$A_{i1}(x_i)$	$A_{i2}(x_i)$	\dots	$A_{ij}(x_i)$	\dots	$A_{iN}(x_i)$	Y_{i1}	Y_{i2}	\dots	Y_{ij}	\dots	Y_{iN}
1	X	\dots	X	\dots	X	1	0	\dots	0	\dots	0
X	1	\dots	X	\dots	X	0	1	\dots	0	\dots	0
\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
X	X	\dots	1	\dots	X	0	0	\dots	1	\dots	0
\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
X	X	\dots	X	\dots	1	0	0	\dots	0	\dots	1

Tabla 8.35: Codificación general para un sistema compacto con arquitectura II.

De esta forma, cada función de activación Y_{ij} depende directamente de un conjunto A_{ij} teniendo la expresión:

$$Y_{ij} = A_{ij}(x_i) \quad (8.59)$$

Empleando las respectivas funciones de pertenencia μ_{ij} asociadas a los conjuntos A_{ij} se tiene el esquema general mostrado en la figura 8.24.

La salida del proceso de inferencia se puede calcular como:

$$y = \sum_{i=1}^M \sum_{j=1}^N v_{ij} \mu_{ij}(x_i) \quad (8.60)$$

Como es de apreciar en la figura 8.24, el esquema del sistema difuso compacto puede ser interpretado como una red neuronal ya que se tienen relaciones de la entrada y la salida mediante las funciones de pertenencia y los actuadores virtuales. De esta manera, los parámetros de adaptación corresponderían a los actuadores virtuales y los valores de configuración para las funciones de pertenencia. Esta arquitectura puede ser utilizada

en identificación y control por tener una representación directa de no linealidades para sistemas dinámicos.

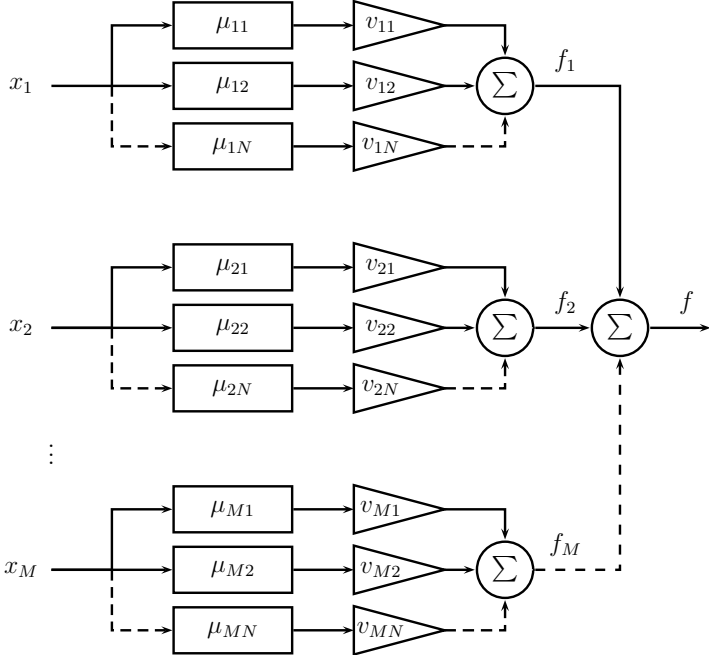


Figura 8.24: Esquema general de un sistema compacto con arquitectura II.

Capítulo 9

Sistema de control neuro-difuso adaptativo basado en relaciones booleanas

9.1. Introducción

En el presente capítulo se describe la arquitectura del sistema de control neuro-difuso adaptativo desarrollado. Tanto para la identificación de la planta como para el controlador se emplea un sistema neuro-difuso basado en relaciones booleanas. En este esquema, de forma iterativa se realiza la identificación de la planta y luego, utilizando este modelo, se efectúa la optimización del controlador. Con este enfoque se busca incorporar información preliminar en la arquitectura del sistema neuro-difuso, de tal forma que se logre la adaptabilidad del sistema de control propuesto.

Para que un sistema de control adaptativo sea viable desde un punto de vista práctico, la estructura del sistema de control y los métodos de entrenamiento deben permitir el ajuste de parámetros en el tiempo requerido.

En la primera parte se describe de forma general el esquema adaptativo, luego se presentan las diferentes configuraciones de los sistemas neuro-difusos dependiendo de las funciones de pertenencia utilizadas. Adicionalmente se muestra un esquema de optimización general para realizar la identificación de la planta y el ajuste del controlador.

9.2. Características del sistema de control adaptativo neuro-difuso

Desde una perspectiva biológica, la adaptabilidad consiste en la capacidad que tienen los organismos para sobrevivir en un determinado ambiente, al lograrlo, prosperan y crean descendencia, pero en caso contrario pueden desaparecer [14]. Según [16] y [17], este principio es aplicable en optimización como también en sistemas de control inteligente

adaptativo. Considerando lo descrito en [20], los sistemas de control adaptativo son adecuados para monitorear el desempeño del sistema de control con parámetros desconocidos y variantes.

Sobre las técnicas de entrenamiento asociadas a los métodos de optimización, los algoritmos evolutivos han mostrado ser una herramienta destacable para acercarse a un valor óptimo global, sin embargo, requieren de varias ejecuciones y un número elevado de evaluaciones de la función objetivo. Por su parte, los métodos basados en el cálculo de gradientes presentan una rápida convergencia, no obstante son susceptibles al punto inicial de búsqueda teniendo convergencia a mínimos locales [26].

De acuerdo con [16], las técnicas de gradiente ofrecen métodos prácticos y efectivos para llevar a cabo la optimización; los métodos de gradiente se pueden utilizar en línea para ajustar todos los parámetros del sistema de control. El enfoque básico consiste en ajustar iterativamente los parámetros para minimizar la función objetivo.

De esta forma, una alternativa para mejorar el desempeño del sistema de control adaptativo empleando algoritmos basados en gradientes, consiste en una adecuada configuración preliminar de los sistemas empleados tanto para la identificación de la planta como para el controlador.

Por su parte, los sistemas difusos tienen la capacidad de modelar procesos no lineales como también adquirir información de un conjunto de datos empleando algoritmos de aprendizaje. A diferencia de las redes neuronales, los basados en lógica difusa utilizan el conocimiento de los expertos directamente o como punto de partida para su optimización [8, 9]. Los sistemas difusos basados en relaciones booleanas permiten tener estructuras compactas, lo cual facilita los cálculos asociados al proceso de inferencia.

Un sistema neuro-difuso basado en relaciones booleanas permite establecer una estructura y configuración preliminar del sistema utilizado para la identificación de la planta y la optimización del controlador. Esto también permite abordar la dificultad que tienen las redes neuronales para establecer esta estructura, como también la inicialización de sus parámetros [11].

Con este enfoque, para el ajuste del modelo de la planta y el controlador sólo se requieren datos de entrenamiento, siendo esta una característica importante en sistemas altamente complejos con incertidumbre y variabilidad [20].

9.3. Arquitectura del sistema de control neuro-difuso adaptativo

El control adaptativo consiste en técnicas que proporcionan un enfoque sistemático para el ajuste automático de la configuración del controlador en tiempo real para lograr o mantener el rendimiento requerido cuando cambian los parámetros del sistema o estos presentan incertidumbre. De esta manera, la técnica de control adaptativo a través de una ley de adaptación puede hacer frente a perturbaciones, incertidumbres en la dinámica del sistema, así como variaciones en las condiciones operativas [227].

Según [227], con respecto a las técnicas de control adaptativo: los métodos directos para el control adaptativo combinan el objetivo de control y la estimación de parámetros en un solo paso, mientras que los métodos indirectos separan la estimación de parámetros y el control en dos pasos. Los métodos directos generalmente se basan en el desarrollo de técnicas para la estimación de parámetros de modo que el error de estimación y el error de trayectoria se lleven a cero simultáneamente. Por su parte, en los métodos indirectos se separa el proceso de estimación y el ajuste del controlador, por lo tanto, este enfoque de control se basa en el principio de equivalencia de certeza (*Certainty Equivalence Principle* CEP) donde el modelo del sistema se ajusta observando su comportamiento en el tiempo, y luego se diseña una política de control tratando el modelo obtenido como verdadero.

Adicionalmente sobre la estructura utilizada, el control adaptativo con modelo de referencia (*Model Reference Adaptive Control* MRAC) es uno de los métodos de diseño de control adaptativo más importantes que proporciona estructuras del controlador y leyes adaptativas para garantizar el seguimiento de salida para un determinado modelo de referencia, y la delimitación de la respuesta en lazo cerrado en presencia de incertidumbres del sistema [228]. Además, para la implementación de un esquema de control adaptativo se debe tener presente que los cambios en las condiciones de funcionamiento pueden requerir un reinicio del procedimiento de adaptación.

La arquitectura considerada utiliza dos sistemas neuro-difusos, uno como controlador y otro para el modelo de la planta. En este esquema, primero se efectúa la identificación de la planta y posteriormente se realiza el entrenamiento del controlador. En la figura 9.1 se pueden apreciar los dos sistemas neuro-difusos empleados.

La identificación de la planta se puede llevar a cabo de dos formas. Una “fuera de línea” obteniendo datos de la planta en lazo abierto, para lo cual se utilizan diferentes señales de entrada que permitan caracterizar el comportamiento de la planta. La segunda opción es “en línea” y corresponde al esquema mostrado en la figura 9.1, donde los datos de entrenamiento se obtienen durante el funcionamiento del sistema de control. En este diagrama, el bloque del modelo de referencia corresponde al comportamiento deseado que se quiere tener del sistema.

En el esquema de la figura 9.1, después de realizar la identificación de la planta, esta se utiliza en el lazo de control para entrenar el controlador utilizando el algoritmo *Dynamic Back Propagation* [229, 230].

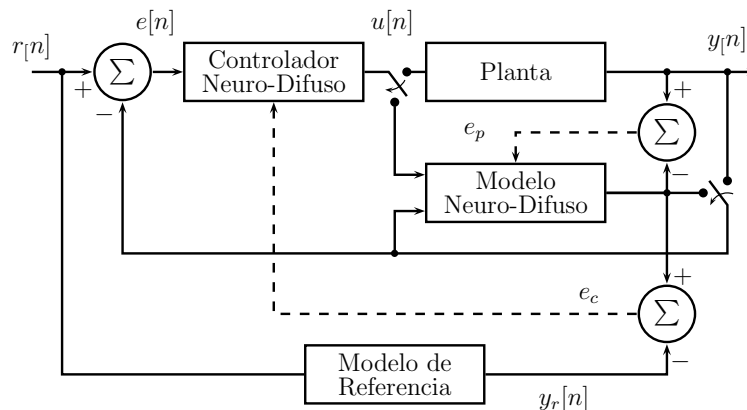


Figura 9.1: Esquema de control mediante sistemas neuro-difusos.

9.3.1. Descripción del proceso de control adaptativo

Para la implementación del sistema de control adaptativo, primero se efectúa la identificación de la planta, después, se realiza el entrenamiento del controlador neuro-difuso. Bajo este esquema se integra el modelo de la planta al lazo de control para el entrenamiento del controlador. El proceso del sistema de control neuro-difuso adaptativo se puede apreciar en la figura 9.2.

En el primer paso se establecen las configuraciones iniciales del controlador y del modelo neuro-difuso de la planta, los cuales se pueden obtener de un conocimiento previo o mediante un entrenamiento “fuera de línea”. En el segundo paso, se toman datos de entrada-salida de la planta durante el funcionamiento del sistema en lazo cerrado. Posteriormente, con los datos tomados se efectúa una nueva identificación de la planta. Luego, utilizando el modelo actualizado de la planta, se hace el entrenamiento del controlador. Como paso siguiente se pone en funcionamiento el controlador optimizado; de esta forma, se busca corregir la variación presente en el sistema. Si no existe algún requerimiento de parada, este proceso se repite desde el paso 2 para el siguiente intervalo de tiempo.

Considerando la limitada cantidad de datos producida durante el funcionamiento del controlador luego que se ha optimizado, el proceso de identificación de la planta y entrenamiento del controlador ocurren de forma iterativa. Es de apreciar la importancia de establecer un punto inicial de búsqueda para la identificación de la planta y optimización del controlador, lo cual se logra con los sistemas neuro-difusos diseñados en el punto 1 del proceso.

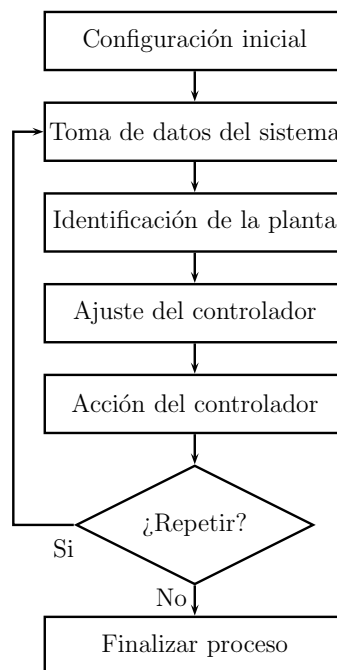


Figura 9.2: Algoritmo del proceso de control adaptativo.

Como es de apreciar, el proceso de control adaptativo está compuesto por los siguientes pasos:

1. **Configuración inicial:** En este paso se determinan las estructuras y configura-

ciones iniciales del controlador y del modelo neuro-difuso de la planta. Como opción se puede realizar un entrenamiento previo del controlador y la identificación de la planta (fuera de línea).

2. **Toma de datos del sistema:** En este proceso se toman datos de entrada-salida de la planta durante el funcionamiento del sistema.
3. **Ajuste de la planta:** Con los datos tomados se efectúa un nuevo entrenamiento del sistema neuro-difuso de la planta, ajustando sus parámetros.
4. **Entrenamiento del controlador:** Con el modelo ajustado de la planta se realiza el entrenamiento del controlador.
5. **Acción del controlador adaptado:** Durante este proceso se pone en funcionamiento el controlador optimizado, de esta forma se busca corregir la variación que se presente en el sistema.
6. **Repetir proceso:** Se repite desde el paso 2 para el siguiente intervalo de tiempo. De esta forma se realiza iterativamente la identificación de la planta y entrenamiento del controlador.

9.4. Arquitectura general para la identificación de la planta

En esta sección se presenta la arquitectura del sistema propuesto para la identificación de la planta. En primer lugar, se describe la estructura básica de un sistema de tiempo continuo, y posteriormente se presenta la arquitectura del sistema en tiempo discreto usado en la identificación.

Para un sistema dinámico el tiempo puede ser una variable continua o discreta. Cuando el sistema es de tiempo continuo la variable t está definida en \mathbb{R} (reales) y el sistema puede ser descrito mediante un conjunto de ecuaciones diferenciales de la forma:

$$\begin{aligned}
 \frac{dx_1}{dt} &= f_1(x_1, x_2, \dots, x_p) \\
 \frac{dx_2}{dt} &= f_2(x_1, x_2, \dots, x_p) \\
 &\vdots \\
 \frac{dx_p}{dt} &= f_p(x_1, x_2, \dots, x_p)
 \end{aligned}
 \tag{9.1}$$

En el caso de considerar un sistema de tiempo discreto la variable independiente n está definida en \mathbb{N} (naturales) y el sistema se puede representar mediante ecuaciones en diferencias de la forma:

$$\begin{aligned}
 x_1[n+1] &= f_1(x_1[n], x_2[n], \dots, x_p[n]) \\
 x_2[n+1] &= f_2(x_1[n], x_2[n], \dots, x_p[n]) \\
 &\vdots \\
 x_p[n+1] &= f_p(x_1[n], x_2[n], \dots, x_p[n])
 \end{aligned}
 \tag{9.2}$$

Para la identificación con el sistema neuro-difuso se toman muestras de la entrada y salida de tal forma que la salida puede interpretarse como una función no lineal de estas señales.

De acuerdo con [6], una aproximación para el modelo del sistema consiste en estimar una estructura del sistema neuro-difuso que pueda desempeñar la misma función. En la figura 9.3 se muestra un ejemplo de la arquitectura empleada.

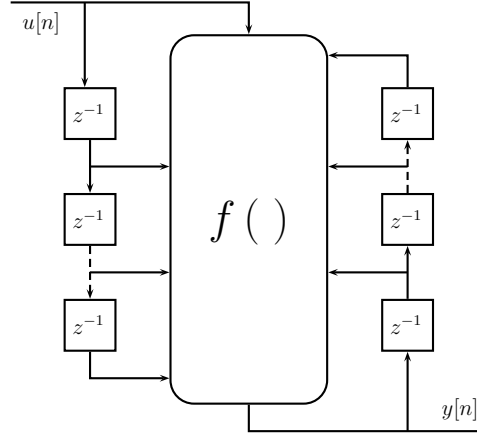


Figura 9.3: Modelo del sistema empleado para la identificación de la planta.

Desde este punto de vista se tiene un conjunto de datos de salida $y[n + 1]$ de la forma:

$$y_s[n + 1] = f_p(y[n], \dots, y[n - N_y], u[n], \dots, u[n - N_u], \mathbf{H}_p) \quad (9.3)$$

donde N_y corresponde al número de muestras tomadas para la salida, N_u el número de muestras de la entrada y \mathbf{H}_p el vector de parámetros del sistema neuro-difuso.

Siendo la variable $h_p \in \mathbf{H}_p$ uno de los parámetros del sistema, entonces, el respectivo ajuste de estos se realiza de la forma:

$$h_p(k + 1) = h_p(k) - \eta \frac{dJ}{dh_p} \quad (9.4)$$

donde J corresponde a la función de ajuste definida como:

$$J = \frac{1}{2} (y[n + 1] - y_s[n + 1])^2 \quad (9.5)$$

y la derivada de J en función de los parámetros de ajuste de la planta es:

$$\frac{dJ}{dh_p} = \frac{dJ}{dy_s[n + 1]} \frac{dy_s[n + 1]}{dh_p} \quad (9.6)$$

9.5. Arquitectura del controlador

Una arquitectura general para el controlador neuro-difuso se puede apreciar en la figura 9.4, donde se tienen como entradas la referencia r y la realimentación de y .

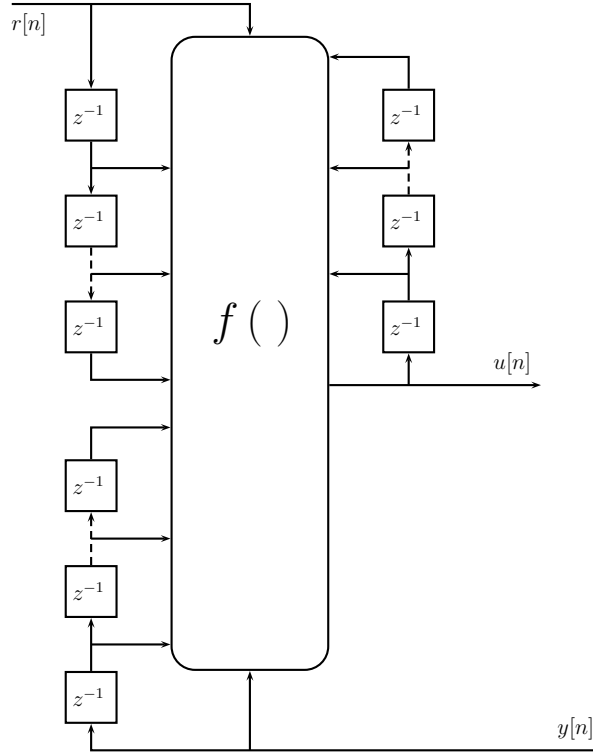


Figura 9.4: Modelo general del controlador.

Bajo este enfoque la acción de control corresponde a:

$$u[n] = f_c(y[n], \dots, y[n - N_y], r[n], \dots, r[n - N_r], u[n - 1], \dots, u[n - N_u], \mathbf{H}_c) \quad (9.7)$$

de la misma forma la salida empleada para el entrenamiento del controlador es:

$$y_s[n + 1] = f_p(y[n], \dots, y[n - N_y], u[n], \dots, u[n - N_u], \mathbf{H}_p) \quad (9.8)$$

donde N_y corresponde al número de muestras tomadas para la salida, N_u el número de muestras de la entrada y \mathbf{H}_p el vector de parámetros del sistema neuro-difuso.

Por otra parte, cuando se considera como entrada al controlador neuro-difuso la variable del error $e[n]$, el sistema se puede representar como se muestra en la figura 9.5.

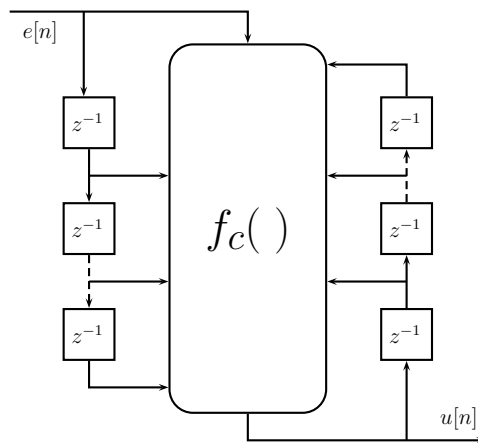


Figura 9.5: Arquitectura del controlador utilizando la señal de error.

De esta manera, se pueden tener arquitecturas similares para la planta y el controlador de tal forma que se pueden alternar en un diseño experimental. Con esta estructura la ecuación del controlador es:

$$u[n] = f_c(e[n], e[n-1], e[n-2], \dots, e[n-N_e], u[n-1], u[n-2], \dots, u[n-N_u], \mathbf{H}_c) \quad (9.9)$$

así, la salida empleada para el entrenamiento del controlador es:

$$y_s[n+1] = f_p(y[n], \dots, y[n-N_y], u[n], \dots, u[n-N_u], \mathbf{H}_p) \quad (9.10)$$

Considerando que $h_c \in \mathbf{H}_c$ es uno de los parámetros del controlador, entonces, la adaptación de estos se hace de la forma:

$$h_c(k+1) = h_c(k) - \eta \frac{dJ}{dh_c} \quad (9.11)$$

donde J es la función de ajuste definida como:

$$J = \frac{1}{2} (y_r[n+1] - y_s[n+1])^2 \quad (9.12)$$

En esta ecuación y_r es la respuesta del modelo de referencia y y_s la salida simulada de la planta. Para el entrenamiento del controlador la derivada de J en función de los parámetros de ajuste es:

$$\frac{dJ}{dh_c} = \frac{dJ}{dy_s[n+1]} \frac{dy_s[n+1]}{dh_c} \quad (9.13)$$

9.6. Sistemas difusos basados en relaciones booleanas empleados para identificación y control

Para la implementación del sistema neuro-difuso se busca incorporar información de un experto lo cual ayuda a la posterior sintonía del sistema. Para esto se consideran dos arquitecturas de sistemas neuro-difusos compactos:

- Arquitectura I: Similar al esquema empleado en redes neuronales de base radial.
- Arquitectura II: Basada en la estructura de un sistema dinámico de tiempo discreto.

9.6.1. Arquitectura I

Esta arquitectura resulta similar al esquema tradicional empleado para el entrenamiento de redes neuronales de base radial y la implementación de sistemas neuro-difusos [226].

La estructura del sistema neuro-difuso en consideración es dada por:

$$f = v_c + \sum_{l=1}^M v_l \left[\prod_{i=1}^N \mu_{il}(x_i) \right] \quad (9.14)$$

La representación de la red neuro-difusa asociada se puede apreciar en la figura 9.6. La primera capa corresponde a la productoria de las funciones de pertenencia, es decir, el cálculo de Y_i . En la segunda se efectúa la sumatoria de las salidas virtuales correspondiente a los actuadores virtuales multiplicados por la función de activación Y_i . Finalmente, en la tercer capa se determina la salida de la inferencia f .

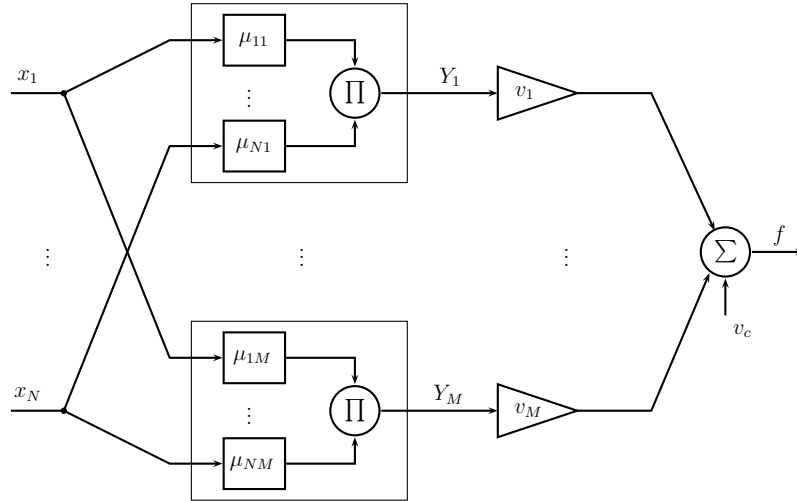


Figura 9.6: Representación de la red neuro-difusa.

Considerando un retardo de entrada y dos para la realimentación de la salida, la figura 9.7 muestra una configuración de este sistema difuso para identificación o control.

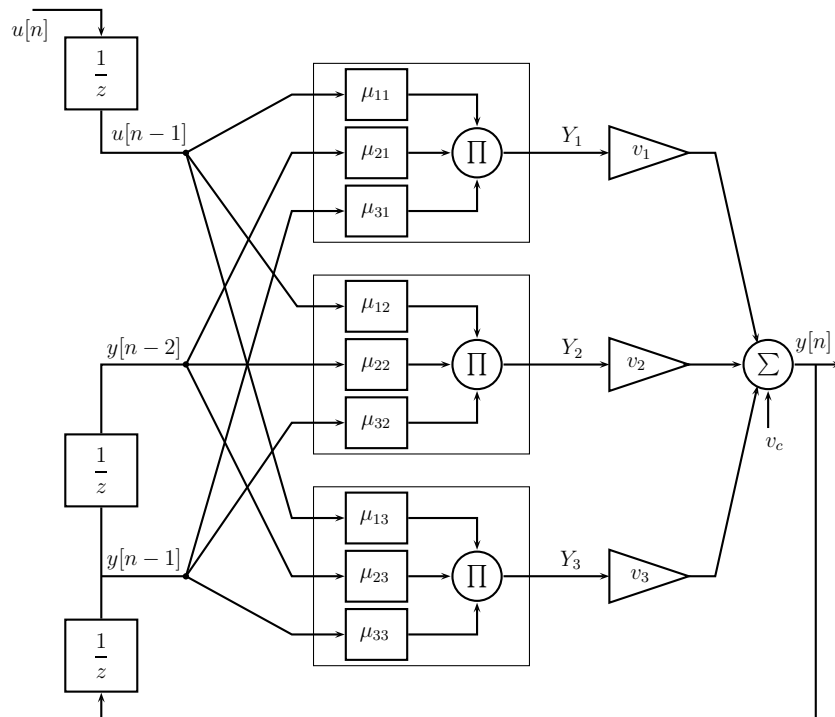


Figura 9.7: Configuración de la red neuro-difusa para identificación o control.

Las reglas que implementan la estructura de la figura 9.7 se pueden representar como

se muestra en la tabla booleana 9.1. En general, para $l = 1, 2, 3$ estas reglas se describen como:

- Si $u[n-1]$ es μ_{1l} y $y[n-2]$ es $\mu_{2,l}$ y $y[n-1]$ es $\mu_{3,l}$ Entonces la función de activación es Y_l .

μ_{11}	μ_{21}	μ_{31}	μ_{12}	μ_{22}	μ_{32}	μ_{13}	μ_{23}	$\mu_{3,3}$	Y_1	Y_2	Y_3	Y_c
1	1	1	X	X	X	X	X	X	1	0	0	1
X	X	X	1	1	1	X	X	X	0	1	0	1
X	X	X	X	X	X	1	1	1	0	0	1	1

Tabla 9.1: Reglas asociadas con la estructura de la figura 9.7.

De esta forma, la salida se calcula mediante la ecuación (9.15), donde v_c tiene asociada una función de activación siempre activa $Y_c = 1$.

$$f = v_1Y_1 + v_2Y_2 + v_3Y_3 + v_4Y_4 + v_c \quad (9.15)$$

9.6.2. Arquitectura II

En este caso, se observa la relación directa que tiene un FIS-BBR compacto para representar un sistema dinámico de tiempo discreto no lineal. Esta equivalencia se observa al modificar las relaciones lineales que se tienen para cada uno de los retardos tanto de la entrada como de las realimentaciones de la salida. De este modo, se modifican las relaciones lineales utilizando conjuntos difusos para representar relaciones no lineales.

Para mostrar la estructura del sistema neuro-difuso, en primer lugar se considera un sistema (controlador) lineal en tiempo discreto donde su función de transferencia es:

$$C(z) = \frac{U(z)}{E(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_{N_e}z^{-N_e}}{1 + a_1z^{-1} + a_2z^{-2} + \dots + b_{N_u}z^{-N_u}} \quad (9.16)$$

La ecuación en diferencias de este sistema corresponde a:

$$u[n] = b_0e[n] + b_1e[n-1] + b_2e[n-2] + \dots + b_je[n-j] + \dots + b_{N_e}[n-N_e] - a_1u[n-1] - a_2u[n-2] - \dots - a_iu[n-i] - \dots - a_{N_u}u[n-N_u] \quad (9.17)$$

donde los respectivos coeficientes a_i , b_j son constantes, para el controlador difuso estas constantes se reemplazan por relaciones no lineales de tal forma que:

$$u[n] = f_{e,0}(e[n]) + f_{e,1}(e[n-1]) + f_{e,2}(e[n-2]) + \dots + f_{e,N_e}(e[n-N_e]) - f_{u,1}(u[n-1]) - f_{u,2}(u[n-2]) - \dots - f_{u,N_u}(u[n-N_u]) \quad (9.18)$$

Con el fin de considerar las no linealidades, incluida la saturación para las funciones $f_{e,j}$ y $f_{u,i}$ del sistema dinámico de tiempo discreto, se puede tener una función como la mostrada en la figura 9.8. Aunque existen diferentes formas, para lograr esto se considera

su planteamiento mediante un proceso de inferencia difusa empleando funciones de pertenencia como las mostradas en la figura 9.9; así, es posible utilizar la arquitectura II de un FIS-BBR compacto para la implementación de un sistema neuro-difuso.

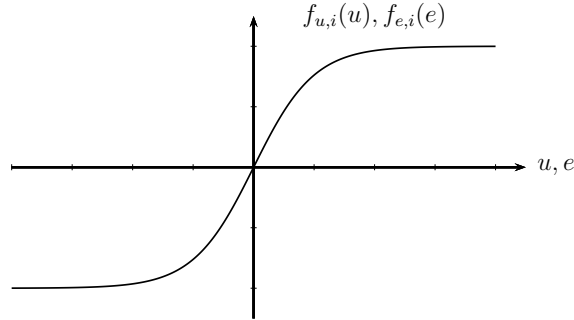


Figura 9.8: Función no lineal para implementar un sistema dinámico de tiempo discreto.

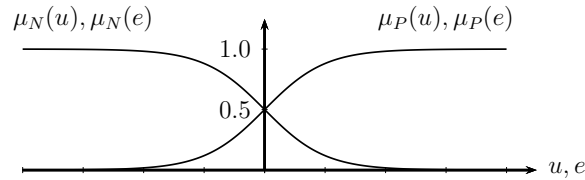


Figura 9.9: Funciones de pertenencia difusas para la implementación de la función no lineal de la figura 9.8.

Considerando los conjuntos difusos de la figura 9.9 y la estructura general del sistema dada por la ecuación 9.18, a manera de ejemplo se tiene el esquema de la figura 9.10, donde se muestra un sistema difuso (propuesto).

La salida del sistema se puede calcular como:

$$u[n] = \sum_{i=1}^5 \sum_{j=1}^2 v_{ij} \mu_{ij}(x_i) \quad (9.19)$$

donde $x_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2], \}$. Para cada entrada x_i se puede definir una función f_i de la forma:

$$f_i = \sum_{j=1}^2 v_{ij} \mu_{ij}(x_i) \quad (9.20)$$

por lo tanto:

$$f_i = v_{i1} \mu_{i1}(x_i) + v_{i2} \mu_{i2}(x_i) \quad (9.21)$$

En la figura 9.10 se puede observar un ejemplo de la configuración de cada f_i para obtener de esta forma un sistema neuro-difuso basado en relaciones booleanas.

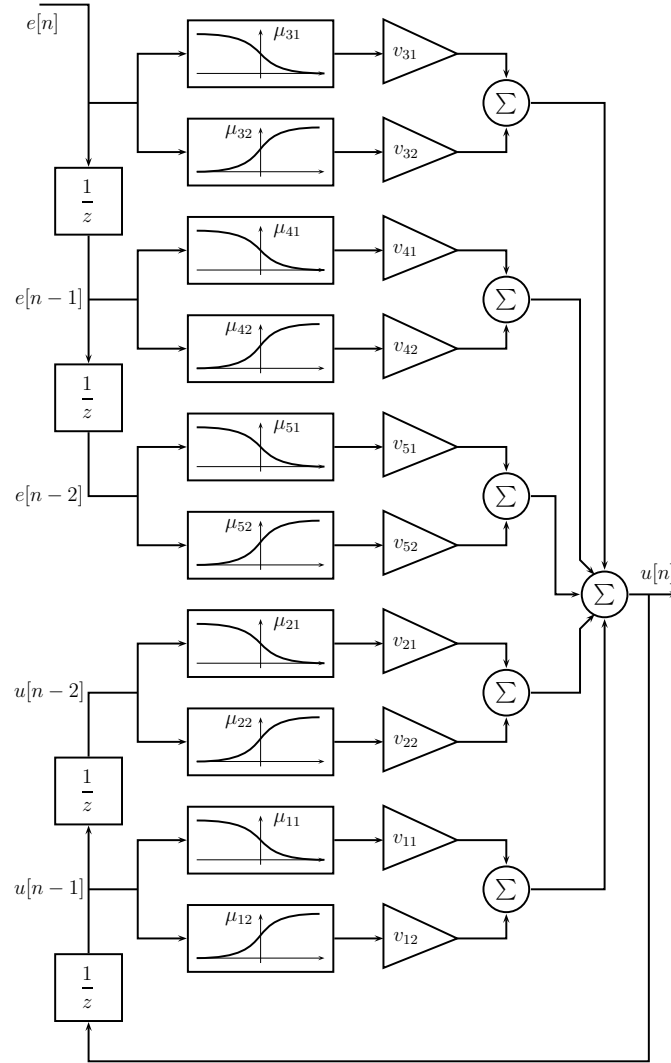


Figura 9.10: Esquema del sistema de neuro-difuso compacto.

Las reglas utilizadas para implementar la estructura de la figura 9.10 se pueden representar como se muestra en la tabla booleana 9.2. En general, estas reglas se describen como:

- Si x_i es μ_{i1} Entonces la función de activación es Y_{i1} .
- Si x_i es μ_{i2} Entonces la función de activación es Y_{i2} .

De este modo, la salida del sistema se calcula como:

$$\begin{aligned}
 u = & v_{11}Y_{11} + v_{12}Y_{12} + v_{21}Y_{21} + v_{22}Y_{22} + v_{31}Y_{31} + v_{32}Y_{32} \\
 & + v_{41}Y_{41} + v_{42}Y_{42} + v_{51}Y_{51} + v_{52}Y_{52} \quad (9.22)
 \end{aligned}$$

μ_{11}	μ_{12}	μ_{21}	μ_{22}	μ_{31}	μ_{32}	μ_{41}	μ_{42}	μ_{51}	μ_{52}	Y_{11}	Y_{12}	Y_{21}	Y_{22}	Y_{31}	Y_{32}	Y_{41}	Y_{42}	Y_{51}	Y_{52}
1	X	X	X	X	X	X	X	X	X	1	0	0	0	0	0	0	0	0	0
X	1	X	X	X	X	X	X	X	X	0	1	0	0	0	0	0	0	0	0
X	X	1	X	X	X	X	X	X	X	0	0	1	0	0	0	0	0	0	0
X	X	X	1	X	X	X	X	X	X	0	0	0	1	0	0	0	0	0	0
X	X	X	X	1	X	X	X	X	X	0	0	0	0	1	0	0	0	0	0
X	X	X	X	X	1	X	X	X	X	0	0	0	0	0	1	0	0	0	0
X	X	X	X	X	X	1	X	X	X	0	0	0	0	0	0	1	0	0	0
X	X	X	X	X	X	X	1	X	X	0	0	0	0	0	0	0	1	0	0
X	X	X	X	X	X	X	X	1	X	0	0	0	0	0	0	0	0	1	0
X	X	X	X	X	X	X	X	X	1	0	0	0	0	0	0	0	0	0	1

Tabla 9.2: Reglas asociadas a la estructura de la figura 9.10.

Considerando $x_i \in \{e[n], e[n-1], \dots, e[n-p], \dots, u[n-1], \dots, u[n-q], \dots, u[n-k], \dots\}$, de forma general, para representar las configuraciones del sistema difuso con diferentes funciones de pertenencia se propone el esquema de la figura 9.11.

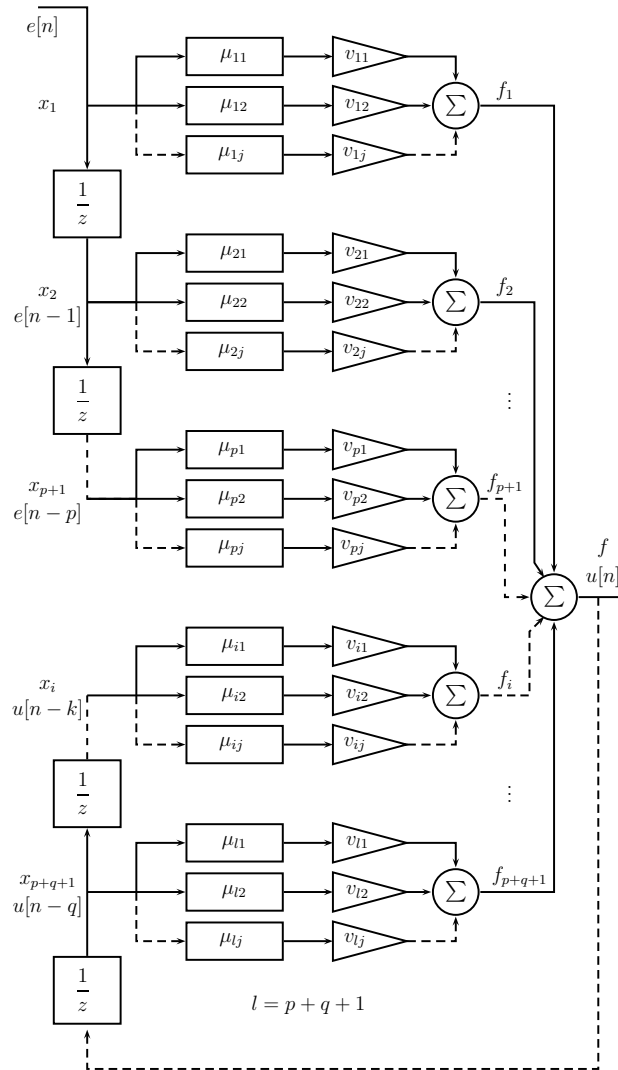


Figura 9.11: Esquema general del sistema neuro-difuso.

La salida del sistema de la figura 9.11 se puede calcular como:

$$f = \sum_{i=1}^M \sum_{j=1}^N v_{ij} \mu_{ij}(x_i) \quad (9.23)$$

Como se observa en la figura 9.11, el esquema del sistema difuso compacto se puede interpretar como una red neuro-difusa ya que las relaciones de entrada y salida se obtienen a través de funciones de pertenencia y actuadores virtuales. El valor de pertenencia para cada entrada se obtiene en la primera capa. Luego, este valor se multiplica por el respectivo actuador virtual v_{ij} en la segunda capa. En la tercera capa, se calculan las funciones f_i asociadas a cada entrada x_i . Finalmente, en esta misma capa, realizando la suma de todas las f_i se determina la salida f .

De esta manera, los parámetros de adaptación corresponden a los actuadores virtuales y los valores de configuración para las funciones de pertenencia. Esta arquitectura se puede utilizar en identificación y control ya que permite la representación de relaciones no lineales entre la entrada y la salida que pueden asociarse con un sistema dinámico.

9.7. Configuraciones de los sistemas neuro-difusos basados en relaciones booleanas

Para la implementación del sistema de control neuro-difuso adaptativo se tienen diferentes configuraciones dependiendo de la arquitectura y las funciones de pertenencia a utilizar, las cuales se pueden apreciar en la tabla 9.3.

Configuración Sistema BBR	Planta			Controlador		
	Arquitectura	Gaussianas	Sigmoidales	Arquitectura	Gaussianas	Sigmoidales
A2P1SA2C1S	II	0	1	II	0	1
A2P2SA2C2S	II	0	2	II	0	2
A2P1G2SA2C1G2S	II	1	2	II	1	2
A2P4SA2C4S	II	0	4	II	0	4
A2P2GA2C2S	II	2	0	II	0	2
A2P2SA2C2G	II	0	2	II	2	0
A1P4GA2C2G	I	4	0	II	2	0
A1P4GA2C2S	I	4	0	II	0	2
A1P4SA2C2G	I	0	4	II	2	0
A1P4SA2C2S	I	0	4	II	0	2
A2P2GA1C4G	II	2	0	I	4	0
A2P2GA1C4S	II	2	0	I	0	4
A2P2SA1C4G	II	0	2	I	4	0
A2P2SA1C4S	II	0	2	I	0	4
A1P4GA1C4G	I	4	0	I	4	0
A1P4GA1C4S	I	4	0	I	0	4
A1P4SA1C4G	I	0	4	I	4	0
A1P4SA1C4S	I	0	4	I	0	4

Tabla 9.3: Configuraciones de las arquitecturas de los sistemas difusos.

9.7.1. Configuración para la arquitectura II con un conjunto difuso sigmoial

En este caso, se utiliza un conjunto difuso sigmoial para la arquitectura II. En la figura 9.12 se tiene un conjunto difuso sigmoial μ_P para modelar los valores positivos del universo de discurso, mientras que con el complemento de este $\mu_{\bar{P}}$ se representan los valores negativos para el error $e[n]$ y la acción de control $u[n]$.

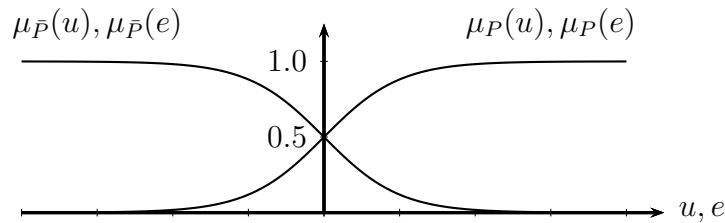


Figura 9.12: Funciones de pertenencia μ_P y $\mu_{\bar{P}}$.

9.7.2. Configuración para la arquitectura II con dos conjuntos difusos sigmoiales

Para este sistema, en la arquitectura II se utilizan dos conjuntos difusos sigmoiales. En la figura 9.13 se muestra un conjunto difuso sigmoial μ_P para modelar los valores positivos del universo de discurso, mientras que μ_N se emplea para representar los valores negativos del error $e[n]$ y la acción de control $u[n]$.

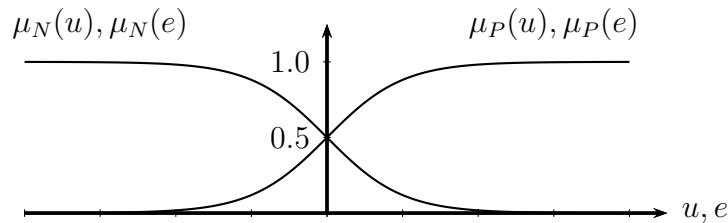


Figura 9.13: Funciones de pertenencia μ_N y μ_P .

Considerando lo anterior la expresión para la función de pertenencia $\mu_{ij}(x_i)$ es:

$$\mu_{ij}(x_i) = \frac{1}{1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})}} = (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} \quad (9.24)$$

9.7.3. Configuración para la arquitectura II con dos conjuntos difusos sigmoiales y uno gaussiano

En este caso, se utilizan las funciones de pertenencia mostradas en la figura 9.14. Con esta configuración μ_P se emplea para modelar los valores positivos del universo de discurso, mientras que μ_N para valores negativos y μ_Z los valores cercanos a cero.

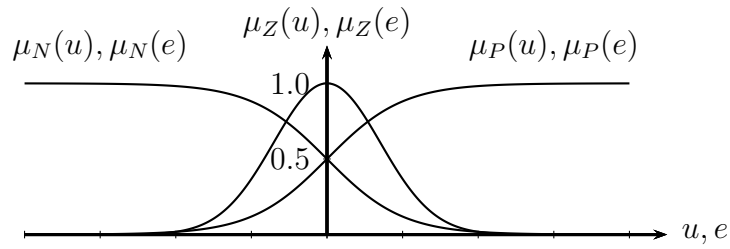


Figura 9.14: Funciones de pertenencia μ_N , μ_Z y μ_P .

9.7.4. Configuración para la arquitectura II con cuatro conjuntos difusos sigmoidales

Para esta configuración se emplean las funciones de pertenencia sigmoidales de la figura 9.15. De esta forma μ_{MP} y μ_P se utilizan para modelar los valores positivos del universo de discurso, mientras que μ_{MN} y μ_N se usan para los valores negativos.

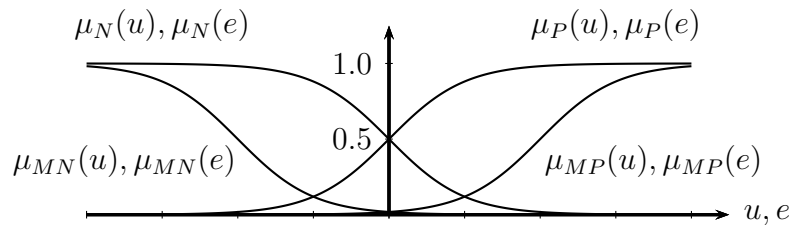


Figura 9.15: Funciones de pertenencia μ_{MN} , μ_N , μ_P , y μ_{MP} .

9.7.5. Configuración para la arquitectura II con dos conjuntos difusos gaussianos

Para este sistema, en la arquitectura II se emplean las funciones gaussianas mostradas en la figura 9.16. Esta configuración se propone considerando acciones sectorizadas dependiendo el universo de discurso.

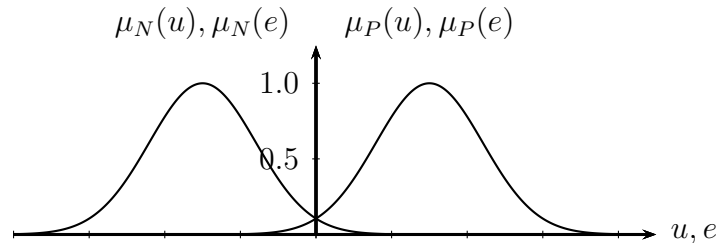


Figura 9.16: Funciones de pertenencia gaussianas μ_N y μ_P .

9.7.6. Configuración para la arquitectura I con cuatro conjuntos difusos gaussianos

En este caso, para la arquitectura I se utilizan las funciones gaussianas de la figura 9.17. Las funciones de pertenencia se reparten uniformemente en los universos de discurso. Esta configuración se enfoca principalmente en la identificación de la planta.

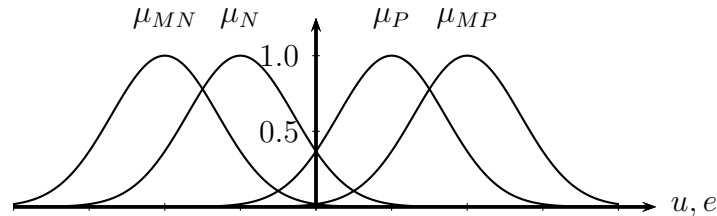


Figura 9.17: Funciones de pertenencia gaussianas.

9.7.7. Configuración para la arquitectura I con cuatro conjuntos difusos sigmoidales

En esta configuración, para la arquitectura I se emplean las funciones sigmoidales de la figura 9.18. Esta configuración se propone como una alternativa de la anterior empleando funciones sigmoidales.

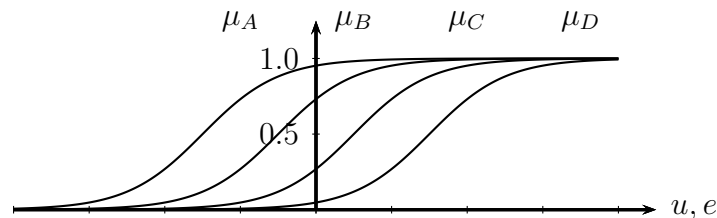


Figura 9.18: Funciones de pertenencia sigmoidales μ_A , μ_B , μ_C y μ_D .

9.8. Proceso general para la optimización de sistemas difusos

Desde una perspectiva general, el proceso de optimización consiste en determinar los mejores valores de los parámetros del sistema neuro-difuso con el fin de lograr la respuesta deseada. Para esto se considera el esquema mostrado en la figura 9.19; primero se calcula la respuesta del sistema con los respectivos parámetros a optimizar, luego, con esta respuesta se implementa la función objetivo utilizada en el algoritmo de optimización para el ajuste de los parámetros del sistema.

Con este enfoque se puede utilizar cualquier algoritmo de optimización, sin embargo, para la implementación del sistema de control neuro-difuso se emplean algoritmos con una rápida convergencia a mínimos locales.

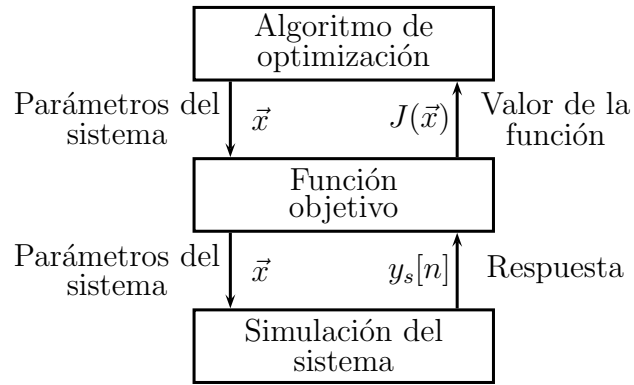


Figura 9.19: Proceso general para la optimización de un sistema neuro-difuso.

En este proceso, la función objetivo considerada es:

$$J(\vec{x}) = \frac{1}{N_T} \sum_{n=1}^{N_T} (y_r[n] - y_s[n, \vec{x}])^2 \quad (9.25)$$

donde N_T es el número de datos tomados, \vec{x} es el grupo de parámetros a optimizar, y_s la respuesta del sistema y y_r la salida deseada del sistema. Este esquema puede ser empleado tanto para identificación como para control.

Capítulo 10

Control del flujo de potencia en un sistema de generación

10.1. Introducción

La energía renovable ha mostrado ser una fuente alternativa prometedora, como también de complemento a la generación eléctrica convencional. Así, los sistemas de generación distribuida resultan ser de gran importancia para el uso racional de la energía y la integración de sistemas de energía renovable [231].

Con la demanda y el aumento de los costos asociados, nuevas alternativas energéticas son cada vez más importantes en la generación económica e ininterrumpida de energía eléctrica. La generación distribuida (*Distributed Generation* DG) se ha convertido en un método atractivo de proporcionar electricidad a los consumidores. Con este enfoque, los costos de instalación de los generadores y los de producción de electricidad pueden ser más económicos. Adicionalmente, la eficiencia eléctrica también se puede mejorar si los servicios públicos utilizan la cogeneración [231].

Convencionalmente, los sistemas de energía eléctrica están compuestos por grandes sistemas interconectados que se caracterizan por la generación centralizada de alta tensión y la transmisión a larga distancia. En los últimos años, con el fin de utilizar nuevos recursos y reducir las pérdidas de energía de transmisión, se emplea el enfoque DG. En general, las tecnologías de generación distribuida se pueden clasificar en: generación de micro-turbina, generación eólica, generación fotovoltaica y generación de motores diesel [232].

Los recursos distribuidos de energía tales como las celdas de combustible, microturbinas y sistemas fotovoltaicos ofrecen muchas ventajas para los sistemas de energía [233], [234]. Por ejemplo, se puede mitigar con eficacia la demanda, como también aumentar la confiabilidad contra fallas del sistema de energía y mejorar la calidad de la misma a través de esquemas de control sofisticados. El concepto de una microrred se ha propuesto con el fin de resolver los problemas de interconexión comunes de las DG individuales en diversos sistemas de energía [235]. Una microrred se define como una red independiente de baja o media tensión de distribución que puede funcionar en tres modos diferentes: conectada a la red, en isla (autónoma), y modo de transición [236].

En este capítulo se presenta la aplicación del sistema de control neuro-difuso adaptativo para un sistema de distribución de energía. En una primera instancia, se revisan algunos aspectos relacionados con la distribución de energía, posteriormente se describe el funcionamiento del esquema de control neuro-difuso adaptativo y se presenta la configuración del sistema de distribución considerado. Luego se muestra la implementación y resultados considerando diferentes configuraciones del sistema neuro-difuso como también los casos para el control de uno y tres generadores.

10.2. Cálculo del flujo de carga

En la ingeniería eléctrica, el estudio del flujo de potencia, también conocido como estudio de flujo de carga, es una herramienta importante que implica el análisis numérico aplicado a un sistema de energía [237]. Los estudios de flujo de carga se utilizan para asegurar que la transferencia de energía eléctrica de los generadores a los consumidores a través del sistema de red es estable, fiable y económica. Los cálculos de flujo de carga permiten establecer los flujos de potencia y tensión en un sistema de energía según la capacidad de regulación de los generadores, condensadores y transformadores [238]. La eficiencia de los algoritmos que permiten establecer el flujo de potencia es de suma importancia, ya que para un estudio de optimización se requiere de numerosas estimaciones de este flujo. El método de barrido de atrás/adelante (*Backward/Forward Sweep* BFS) corresponde a la técnica más empleada para establecer los flujos de potencia con redes de topología radial.

El cálculo del flujo de energía es una herramienta básica e importante en el análisis de sistemas de energía y su convergencia es de gran atención [239, 240]. Los métodos ampliamente utilizados para las redes de distribución son el método de Newton-Raphson (NR) y el método de barrido hacia atrás/adelante (BFS). El método NR tiene una buena tasa de convergencia, pero su ejecución requiere de mucho tiempo [241]. Por su parte, el método BFS es rápido pero presenta dificultades con redes tipo bucle. Algunas investigaciones señalan que el método NR tiene una tasa de convergencia cuadrática mientras que el método BFS presenta una tasa de convergencia lineal [242].

10.3. Operador de la red de distribución

Las redes de distribución de electricidad son una parte crítica en la infraestructura por cuanto permiten llevar la electricidad a los hogares y empresas. El ofrecer un servicio continuo y estable es una función esencial de los operadores de la red de distribución (*Distribution Network Operators* DNOs) para que los hogares y las empresas sean capaces de funcionar.

La incorporación de la generación distribuida (*Distributed Generation* DG) en las redes de distribución produce efectos importantes sobre el funcionamiento tradicional de este sistema. Las redes de distribución existentes están diseñadas para ser pasivas proporcionando el transporte de electricidad con un nivel mínimo de control, seguimiento y supervisión. De la misma forma, tampoco fueron diseñadas para dar cabida a la generación

con voltajes más bajos; por lo tanto, la DG introduce nuevos retos para los DNOs pero también oportunidades que reflejan los beneficios económicos derivados de las redes más activas [243].

Diferentes estudios han evaluado el impacto producido por la integración de más DG; sin embargo, la distribución de estos beneficios a través de las diferentes partes es todavía un trabajo en progreso [243].

10.4. Métodos de regulación

En la actualidad se presenta una tendencia mundial para el uso de fuentes de energía renovables consideradas la solución óptima para la contaminación ambiental, el calentamiento global, y el alto costo de los combustibles fósiles. La naturaleza intermitente de las fuentes de energía renovables como la eólica y la energía solar presenta desviaciones rápidas y significativas de la tensión de la red de distribución [244].

La regulación de voltaje en las redes de distribución se hace más difícil cuando se introducen las unidades de generación distribuida [244], siendo las principales razones:

- Flujo de potencia inversa que se introduce por las unidades de generación distribuida.
- Naturaleza probabilística asociada a las fuentes de energía renovable.

Estrategias tales como *Coordinated On-Load Tap Changer* (OLTC), control de tensión y de compensación de energía reactiva (*Reactive Power Compensation* RPC) se sugieren con frecuencia para la regulación de voltaje en un sistema de distribución con un alto nivel de generación distribuida [245].

Al incorporar generadores distribuidos se tiene en una red de distribución activa. Así la red de distribución no tiene un flujo de potencia unidireccional, como en las redes pasivas, ya que se tiene un flujo de potencia inversa [244].

El flujo de potencia inversa tiene un efecto considerable en la regulación de la tensión a nivel de distribución. Por lo tanto, el problema de la regulación de tensión en la red de distribución se hace más difícil cuando se consideran generadores distribuidos.

10.4.1. Regulador automático de voltaje

Según [246], con el crecimiento y la complejidad de los sistemas de generación de electricidad distribuida, como se tiene con la integración de parques eólicos en los sistemas de energía tradicionales; se plantean considerables preocupaciones sobre la estabilidad de los sistemas de energía y la adecuación de los controles de estabilidad convencionales.

Al integrar generación distribuida en una red de baja tensión (*Low Voltage* LV), los problemas para la regulación de la tensión son cada vez más frecuentes y con un nivel

mayor de complejidad. Debido a la conexión de micro generadores pueden aparecer problemas como el aumento de voltaje, especialmente en escenarios sin carga; lo cual produce la desconexión indeseable de los generadores cuando el voltaje de la red excede los valores establecidos [247].

El voltaje es un parámetro importante para el control de los sistemas de energía eléctrica y los operadores de red de distribución (DNO) tienen la responsabilidad de regular el voltaje suministrado al consumidor dentro de los límites legales. La conexión de la generación distribuida (DG) a la red afecta los perfiles de voltaje e influye en su control en los sistemas de distribución [248].

En los sistemas de energía para áreas rurales, los reguladores automáticos de voltaje (*Automatic Voltage Regulator* AVR) ayudan a reducir la pérdida de energía y mejorar la calidad de los servicios eléctricos, compensando las caídas de voltaje a través de las líneas de distribución [249]. Las redes débiles en estas áreas a menudo tienen problemas de sobretensión cuando se presenta una carga baja y alta generación, lo cual puede producir que se activen las protecciones y se interrumpa el suministro de energía.

Los controles de voltaje en el sistema de distribución se consideran en tres niveles jerárquicos: primarios, secundarios y terciarios. El control primario se realiza mediante un regulador automático de voltaje; el control secundario se efectúa con un cambiador de tomas bajo carga (*On-Load Tap Changer* OLTC), mientras que el control terciario corresponde a una operación corta que se desarrolla para coordinar la acción del dispositivo de control primario y secundario, según la operación segura y criterios económicos basados en la carga y pronóstico de generación [248].

Un regulador automático de voltaje es un dispositivo diseñado para controlar de forma automática, ajustar o mantener un nivel de tensión constante a pesar de las variaciones de carga dentro de su potencia nominal. De acuerdo con [249], la razón para la regulación de voltaje es de tipo financiera para evitar los costos asociados con el daño material y el tiempo de inactividad causada por niveles de tensión peligrosos e indeseados.

10.4.2. Control de conmutadores en carga

Para muchas aplicaciones de transformadores de potencia, una interrupción del suministro durante un cambiador de tomas es inaceptable y el transformador es a menudo provisto de mecanismos de tomas en carga (*On-Load Tap Changer* OLTC), los cuales pueden ser mecánicos, con asistencia electrónica o totalmente electrónicos.

La función principal del cambiador o computador de tomas en carga (OLTC) es regular el voltaje del lado secundario del transformador de distribución de modo que la tensión del bus no supere los límites establecidos [250].

Al incorporar generadores distribuidos se tiene una red de distribución activa puesto que no se presenta un flujo de potencia unidireccional, como en las redes pasivas, ya que se tiene un flujo de potencia inversa [244]. Este flujo de potencia inversa tiene un efecto considerable en la regulación de la tensión a nivel de distribución. La intermitencia y el flujo de potencia inversa puede resultar en una operación excesiva del OLTC. El problema es peor si el OLTC le proporciona potencia a múltiples alimentadores, puesto que algunos

alimentadores pueden sufrir de sobretensión debido a la alta penetración de los generadores distribuidos, mientras que otros pueden sufrir de baja tensión durante la carga elevada. En este escenario, el OLTC convencional, que asume una tendencia descendente de tensión desde la subestación a los terminales de los alimentadores, no puede regular la tensión empleando mediciones locales [244].

10.5. Trabajos realizados para el control de flujo de potencia

La introducción de la generación distribuida (DG) a partir de recursos renovables en una red eléctrica puede tener un profundo impacto en su fiabilidad, calidad de la energía y el funcionamiento [251].

En la actualidad, las medidas para mejorar la estabilidad de la tensión incluyen: la compensación reactiva y la distribución de la potencia reactiva razonable, transformadores equipados con OLTC, compensación de potencia reactiva combinada con transformadores ajustables para regular la tensión y la ampliación de los cables de diámetro, entre otros [252].

Para el problema de estabilidad de la tensión causada por la integración, en particular de la energía eólica, se suelen emplear bancos de baterías o de condensadores. En [253], se indica que adicionar dispositivos de compensación de potencia reactiva estáticos en el bus en la subestación del parque eólico, en combinación con el control de las turbinas de viento (ángulo de la hoja), pueden mejorar la estabilidad de la tensión transitoria del parque eólico.

Según [254], en los últimos años ha habido un creciente interés en alejarse de la generación centralizada de energía hacia los recursos energéticos distribuidos. Los sistemas de generación híbridos de energía solar y eólica presentan varias ventajas para su uso como fuente de energía distribuida, especialmente como fuente de energía en horas pico. La energía solar almacenada en baterías se puede emplear para la compensación de la potencia reactiva y la eliminación de armónicos con las unidades DG vecinas empleando inversores. Para este enfoque, convencionalmente los controladores desarrollados son del tipo Proporcional Integral (PI) y los métodos de control predictivo [254].

Normalmente, los controladores PI convencionales no tienen una compensación adecuada de los inversores para las aplicaciones de redes conectadas con DGs. Los sistemas de control predictivo existentes se basan en el control de inversores de tensión, sin embargo, este método es bastante complicado y algunas estrategias de control predictivo digitales sufren de retardos de control [254].

Sobre otros trabajos realizados, en [255] se presenta una propuesta para la compensación de potencia reactiva en un parque eólico. Se realiza el estudio para el control de tensión constante y el factor de potencia constante para la operación en estado estacionario de la red.

Por su parte en [254], se realiza un análisis de la calidad de la energía y el rendimiento de un inversor conectado a una red inteligente de generación distribuida. La estructura

del controlador diseñado consta de un lazo exterior para el control de potencia, un lazo intermedio para el control de voltaje y un lazo interno de control de corriente. El funcionamiento del controlador se investiga para variar la demanda de energía con cargas lineales y no lineales desde el lado del cliente.

Finalmente, en [256] se presenta una estrategia para el control de potencia reactiva en parques eólicos. En este trabajo se propone un esquema de control donde se combina el control de caída de tensión y el control de factor de potencia constante. El control de caída de tensión se realiza con el voltaje de red promedio durante un ciclo de 30 minutos, de esta forma se permite que el controlador de tensión sea más eficaz en una amplia gama de voltajes.

10.6. Aplicación del sistema de control neuro-difuso adaptativo para un sistema de distribución de energía

Con el fin de realizar el control de tensión en un sistema de distribución se debe contar con los elementos para que un sistema de control neuro-difuso adaptativo sea viable desde un punto de vista práctico, es decir, que la estructura del sistema de control y los métodos de entrenamiento permitan realizar el ajuste de parámetros en el tiempo requerido.

Tal como se comentó en anteriores capítulos para la identificación de la planta y la optimización del controlador, se pueden utilizar algoritmos evolutivos, sin embargo, el número de iteraciones es mayor que para un algoritmo basado en gradientes [26, 50]. Por lo cual, en esta propuesta se busca tener una configuración inicial adecuada que permita la adaptación de los parámetros del sistema de control empleando algoritmos basados en gradientes.

Los sistemas difusos permiten modelar procesos no lineales y adquirir información de un conjunto de datos empleando algoritmos de aprendizaje. A diferencia de los sistemas neuronales, los basados en lógica difusa permiten utilizar fácilmente el conocimiento de los expertos directamente o como punto de partida para su optimización [8, 9]. Para el desarrollo del sistema de control neuro-difuso adaptativo se busca utilizar este concepto.

Por su parte, los sistemas difusos basados en relaciones booleanas muestran un esquema compacto, lo cual facilita los cálculos asociados al proceso de inferencia teniendo estructuras compactas, tanto para la identificación de la planta como para el controlador.

De esta forma se buscan utilizar las características de los sistemas neuro-difusos basados en relaciones booleanas para el control de un sistema de distribución, el cual presenta una alta variabilidad para los valores del consumo de energía.

10.6.1. Proceso de control adaptativo

Para el sistema de control primero se efectúa la identificación de la planta correspondiente al sistema de distribución; con la planta identificada se puede proceder al entrena-

miento del controlador neuro-difuso. Bajo este esquema se integra el modelo de la planta con la sintonía del controlador. El proceso del sistema de control neuro-difuso adaptativo es:

1. **Configuración inicial:** En este paso se lleva a cabo el entrenamiento del controlador y la identificación de la planta con el modelo nominal de la red de distribución. En las figuras 10.1 y 10.2 se puede apreciar una representación de este proceso. De esta forma se tienen las configuraciones iniciales del controlador y del modelo neuro-difuso de la planta.
2. **Toma de datos del sistema:** En este proceso se toman datos de entrada-salida de la planta durante el funcionamiento del sistema. Un ejemplo de este paso se puede apreciar en la figura 10.3.
3. **Ajuste de la planta:** Con los datos tomados se efectúa un nuevo entrenamiento del sistema neuro-difuso de la planta, de tal forma que se adaptan sus parámetros con los datos tomados. Este proceso se observa en la figura 10.4.
4. **Entrenamiento del controlador:** Con el modelo ajustado de la planta se efectúa el entrenamiento del controlador, el esquema de este paso se muestra en la figura 10.5.
5. **Acción del controlador adaptado:** Durante este proceso se pone en funcionamiento el control optimizado en la red de distribución, así se busca corregir la variación que se presentó en el sistema. El respectivo ejemplo se puede apreciar en la figura 10.6.
6. **Repetir proceso:** Se repite desde el paso 2 para el siguiente intervalo de tiempo, de tal forma que se lleva a cabo un proceso iterativo de identificación y entrenamiento del controlador. Un ejemplo de este proceso iterativo se puede apreciar en la figura 10.7.

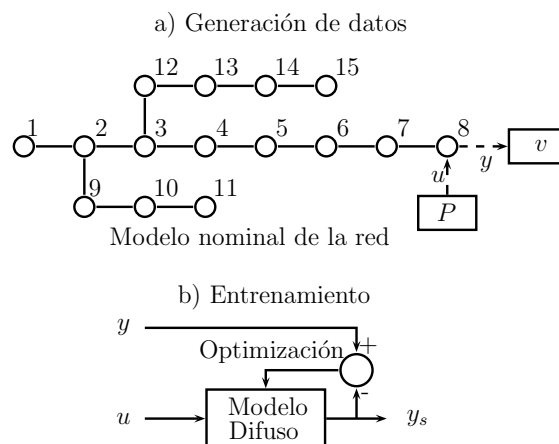


Figura 10.1: Identificación de la planta con el modelo nominal de la red.

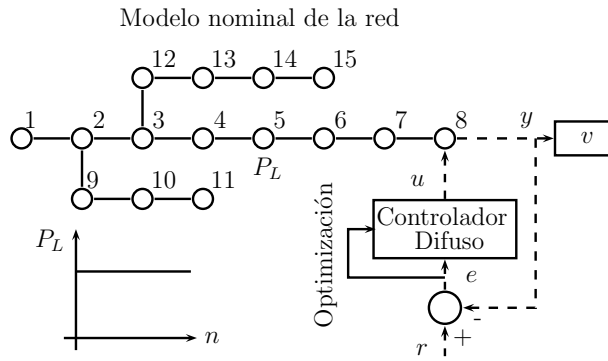


Figura 10.2: Entrenamiento del controlador con el modelo nominal de la red.

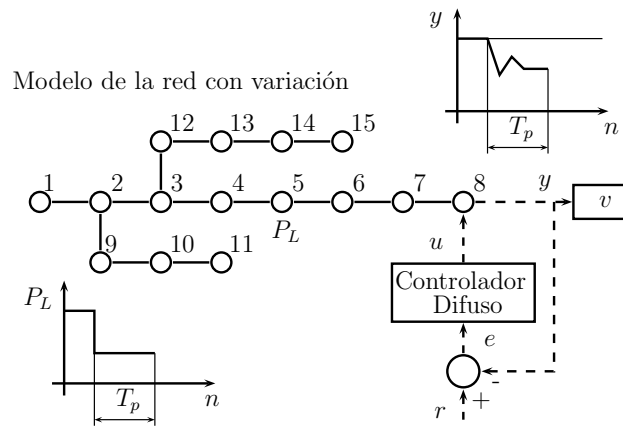


Figura 10.3: Funcionamiento del sistema de control cuando se presenta variación de la carga.

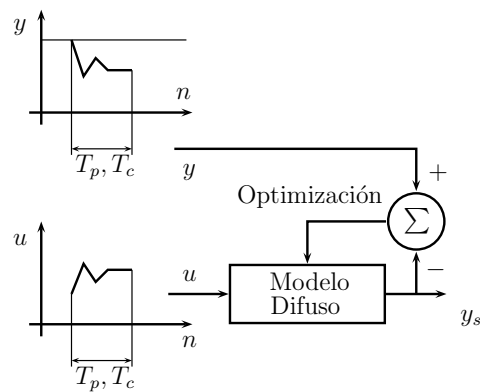


Figura 10.4: Ajuste del modelo de la planta.

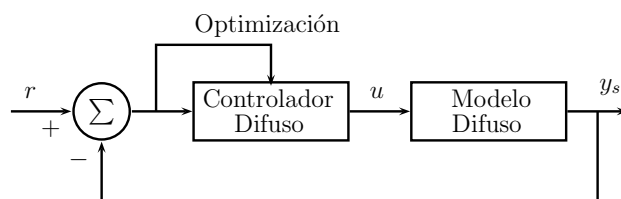


Figura 10.5: Entrenamiento del controlador con el modelo difuso de la planta.

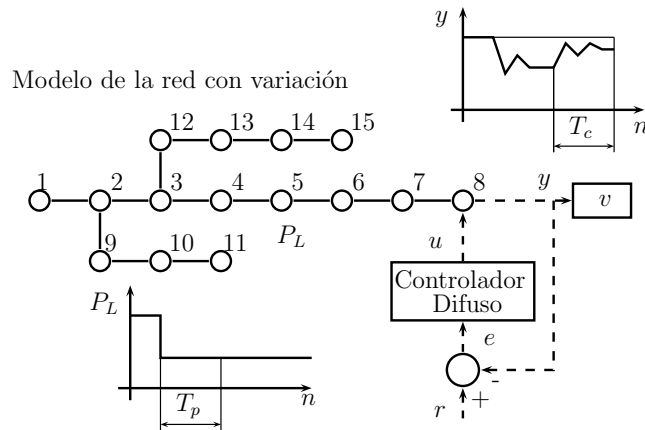


Figura 10.6: Funcionamiento del sistema luego del respectivo ajuste del controlador.

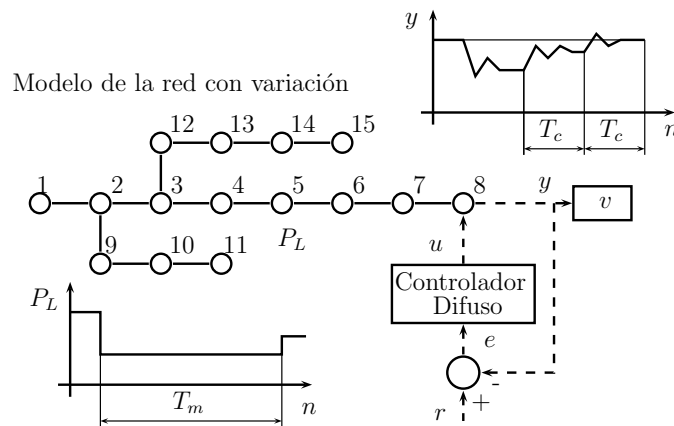


Figura 10.7: Funcionamiento del sistema luego del segundo ajuste del controlador.

Considerando la limitada cantidad de datos producida durante la variación de la carga, como también de la respuesta del sistema luego que se ha optimizado el controlador, el proceso de identificación de la planta y entrenamiento del controlador es iterativo. Es de apreciar la importancia de establecer un punto inicial de búsqueda para la identificación de la planta y optimización del controlador cuando existe variación de la carga, lo cual se logra con los sistemas neuro-difusos determinados en el punto 1 del proceso.

Los tiempos involucrados en un proceso de adaptación y control se pueden apreciar en la figura 10.8, donde T_p corresponde al tiempo para el cual se tiene la operación normal del controlador después de ocurrir un cambio en el sistema, T_c es el tiempo donde opera el sistema con el respectivo ajuste del controlador, T_m corresponde al intervalo de tiempo para el cual se presenta la variación del sistema y finalmente, T_a es el tiempo que se dispone para efectuar el proceso de adaptación compuesto por la identificación de la planta y la optimización del controlador. En este caso, se debe disponer tanto de las herramientas de cálculo como de algoritmos adecuados para lograr el proceso de adaptación en este intervalo de tiempo.

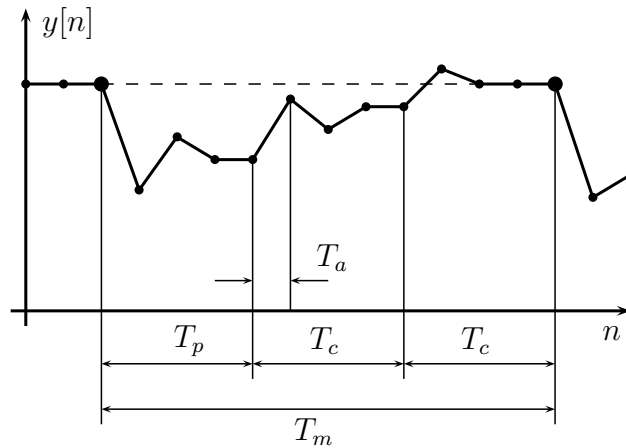


Figura 10.8: Tiempos involucrados en el proceso de adaptación y control.

10.7. Sistema de distribución de energía

El modelo de red radial considerado para el sistema de distribución se puede apreciar en la figura 10.9 donde se presenta la numeración de los nodos.

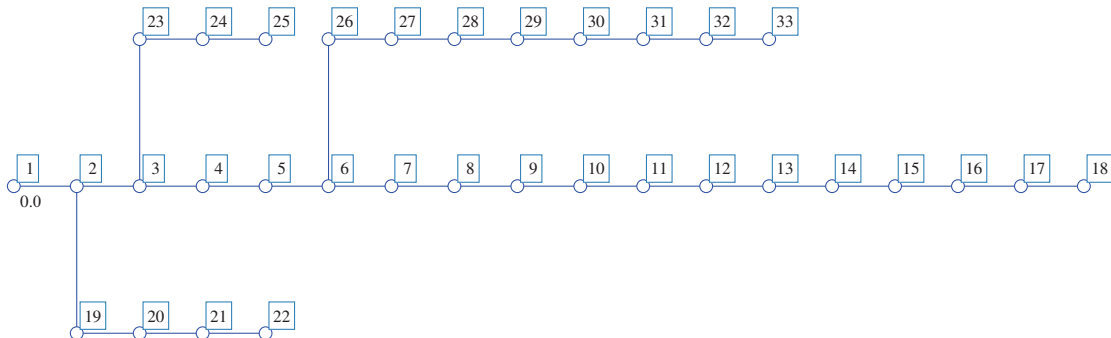


Figura 10.9: Modelo de la red considerada.

El sistema está compuesto por 33 nodos, los valores de las impedancias para las líneas del sistema de distribución se presentan en la tabla 10.1. Por su parte, las potencias de carga para cada nodo se pueden apreciar en la tabla 10.2. Adicionalmente, el nodo 1 se toma como referencia. En [257] se puede apreciar un caso de aplicación de este sistema de distribución.

Esta configuración corresponde al modelo nominal de la red sin considerar los generadores distribuidos y las cargas variables. En el diseño experimental se establece la configuración de estos elementos.

Línea	Bus de entrada	Bus de salida	Resistencia	Reactancia
1	1	2	0.0922	0.0470
2	2	3	0.4930	0.2511
3	3	4	0.3660	0.1864
4	4	5	0.3811	0.1941
4	5	6	0.8190	0.7070
6	6	7	0.1872	0.6188
7	7	8	0.7114	0.2351
8	8	9	1.0300	0.7400
9	9	10	1.0440	0.7400
10	10	11	0.1966	0.0650
11	11	12	0.3744	0.1238
12	12	13	1.4680	1.1550
13	13	14	0.5416	0.7129
14	14	15	0.5910	0.5260
15	15	16	0.7463	0.5450
16	16	17	1.2890	1.7210
17	17	18	0.7320	0.5740
18	2	19	0.1640	0.1565
19	19	20	1.5042	1.3554
20	20	21	0.4095	0.4784
21	21	22	0.7089	0.9373
22	3	23	0.4512	0.3083
23	23	24	0.8980	0.7091
24	24	25	0.8960	0.7011
25	6	26	0.2030	0.1034
26	26	27	0.2842	0.1447
27	27	28	1.0590	0.9337
28	28	29	0.8042	0.7006
29	29	30	0.5075	0.2585
30	30	31	0.9744	0.9630
31	31	32	0.3105	0.3619
32	32	33	0.3410	0.5302
-	-	-	-	-

Tabla 10.1: Impedancias de la red.

Bus	Real	Reactiva
1	0	0
2	100	60
3	90	40
4	120	80
5	60	30
6	60	20
7	200	100
8	200	100
9	60	20
10	60	20
11	45	30
12	60	35
13	60	35
14	120	80
15	60	10
16	60	20
17	60	20
18	90	40
19	90	40
20	90	40
21	90	40
22	90	40
23	90	50
24	420	200
25	420	200
26	60	25
27	60	25
28	60	20
29	120	70
30	200	600
31	150	70
32	210	100
33	60	40

Tabla 10.2: Potencias de carga.

10.7.1. Resultados experimentales

En esta sección se muestran los resultados de la implementación del sistema de control neuro-difuso adaptativo, se observan resultados desde la perspectiva estadística y también la forma de las señales en las simulaciones. Según lo presentado en [17, 20], para este tipo de aplicaciones se observa el resultado de un controlador adaptativo en comparación a su versión no adaptativa.

En esta aplicación, para la identificación de la planta y la optimización del controlador se utiliza el procedimiento descrito en la sección 9.8 utilizando el gradiente descendente como algoritmo de optimización. Para la inicialización de los sistemas neuro-difusos, en primer lugar, se considera su configuración descrita en la sección 9.7. En segundo lugar, se tiene que los valores de las señales de entrada y salida se encuentran normalizados con valores de referencia iguales a la unidad, por lo cual, la inicialización de las funciones de pertenencia está en el rango $[-2, 2]$. Además se lleva a cabo un entrenamiento preliminar (fuera de línea) del controlador y del modelo neuro-difuso de la planta, antes de poner en operación el sistema de control adaptativo.

El análisis estadístico se realiza tomando como patrón de comparación el sistema de

control sin adaptación para poder observar los casos donde la estrategia adaptativa es exitosa. Además, mediante el análisis estadístico se observa el efecto de las posibles configuraciones de los sistemas neuro-difusos considerando el número de entradas y funciones de pertenencia. De esta forma, con los resultados estadísticos y las simulaciones se muestra el proceso adaptativo que se logra con los sistemas neuro-difusos.

10.7.2. Metodología para el análisis estadístico de resultados

Dada la naturaleza estocástica del sistema, se puede presentar variabilidad en los resultados al emplear una configuración particular del sistema de control. Para observar si existe diferencia significativa en los resultados obtenidos por el sistema de control (considerando diferentes aspectos a comparar), se realiza una prueba estadística de hipótesis. Así, las hipótesis a considerar son:

- H_0 Hipótesis nula: Los resultados obtenidos por los sistemas de control presentan valores medios iguales.
- H_1 Hipótesis alternativa: Los resultados obtenidos por los sistemas de control no presentan valores medios iguales.

Al formular estas hipótesis existe la posibilidad de cometer errores tal como se presenta en la tabla 10.3, donde el error tipo I ocurre cuando se rechaza la hipótesis nula siendo esta verdadera, mientras que para el error tipo II se acepta la hipótesis nula siendo falsa [258].

Decisión \ Condición real	H_0 verdadera	H_0 falsa
Rechazar H_0	Error Tipo I	Correcto
Aceptar H_0	Correcto	Error Tipo II

Tabla 10.3: Error tipo I y tipo II.

Comúnmente, la prueba de hipótesis se efectúa considerando un nivel de significancia denominado p -value, correspondiente a la probabilidad de cometer un error tipo I. Con esta orientación, la hipótesis nula se rechaza si el p -value del estadístico de prueba es igual o menor que un nivel de significancia establecido, por lo general del 5% [258].

Pruebas estadísticas

Para su aplicación, las pruebas estadísticas se pueden clasificar principalmente como paramétricas y no paramétricas. Las pruebas paramétricas son robustas pero están basadas en suposiciones de normalidad y de igualdad de varianza de los datos. Por su parte, las pruebas no paramétricas no requieren suposiciones para su aplicación pero pierden información ya que la comparación se realiza con la representación de los datos en una escala ordinal [258].

La metodología para la prueba de hipótesis se puede apreciar en la figura 10.10 donde se tiene:

- Kolmogorov-Smirnov: Prueba utilizada para determinar la normalidad de los datos, como alternativas se tienen las pruebas de Shapiro-Wilk y de Anderson-Darling.
- Levene: Mediante esta prueba se establece la igualdad de las varianzas (homocedasticidad), una alternativa corresponde a la prueba de Bartled.
- Welch: Prueba empleada para comparar varias distribuciones, es una extensión de la T de Student, esta prueba requiere normalidad de los datos.
- ANOVA: Prueba para comparar varias distribuciones, se requiere normalidad y homocedasticidad.
- Kruskal-Wallis: Prueba no paramétrica para comparar varias distribuciones, no requiere suposiciones previas.

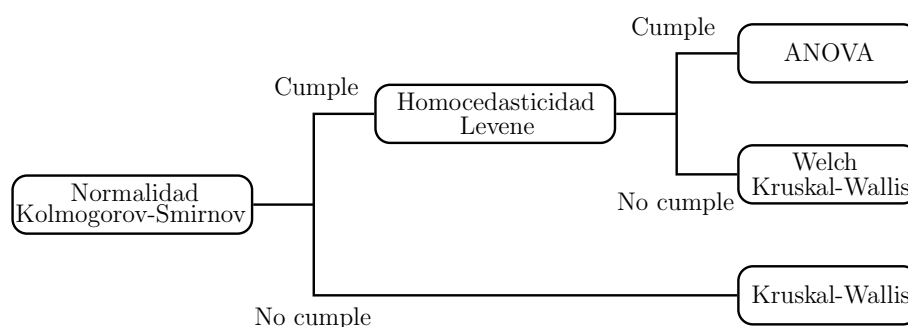


Figura 10.10: Metodología para establecer la prueba de hipótesis a desarrollar.

En el caso de presentarse diferencia significativa entre los grupos experimentales, se procede a realizar pruebas de comparaciones múltiples para establecer estas diferencias [259].

De la misma forma que en la metodología de la figura 10.10, cuando se cumplen los supuestos de normalidad y homocedasticidad, se pueden emplear los contrastes de Duncan, Newman-Keuls, Bonferroni, Scheffé o HSD de Tukey [258, 259]. Por su parte, en el caso de no cumplirse estos supuestos, se utilizan los contrastes no paramétricos de Nemenyi, Holm, Bonferroni-Dunn [260, 261].

Como resultado de realizar el contraste entre grupos, se tienen intervalos de confianza, los cuales permiten determinar si los grupos a comparar presentan diferencias significativas. Una forma para representar este resultado consiste en mostrar gráficamente el *ranking* promedio de cada grupo y un intervalo equivalente, con esta representación dos grupos de resultados se consideran diferentes si sus intervalos no se traslapan [262, 263].

10.7.3. Métrica utilizada

Estos resultados están encaminados a mostrar la capacidad de adaptación que tiene el sistema neuro-difuso, por lo cual, se realiza la comparación con el controlador difuso sin adaptación.

Para realizar la comparación, el índice de desempeño utilizado corresponde al error cuadrático medio MSE (*Mean Squared Error*). Es de notar que esta métrica se utiliza como función objetivo para la optimización del controlador mediante el procedimiento descrito en la sección 9.8. El valor del MSE se puede determinar con la ecuación (10.1), donde r es la señal de referencia, y la salida del sistema (voltaje normalizado) y N_T el número total de datos tomados durante la simulación.

$$MSE = \frac{1}{N_T} \sum_{n=1}^{N_T} (r[n] - y[n])^2 \quad (10.1)$$

10.7.4. Configuración de experimentos

En el diseño experimental se consideran diferentes aspectos para observar las características del sistema de control adaptativo. Un primer aspecto a comparar con los experimentos consiste en el desempeño del controlador con y sin el proceso adaptativo. En segunda instancia se tiene la configuración empleada para el sistema de control, es decir, el número de retardos en la entrada y la salida (ver figuras 9.3 y 8.24).

Considerando que los datos para la variación de la carga se tienen cada hora y empleando una escala en minutos, entonces se toma $T_m = 60min$ y $T_p = 20min$. Por su parte, para el controlador y el modelo de la planta existen diferentes configuraciones dependiendo las entradas y las realimentaciones. En la tabla 10.4 se puede apreciar las configuraciones experimentales consideradas para el caso adaptativo y no adaptativo.

Entradas	Realimentaciones	No Adaptativo	Adaptativo
1	2	CS1AD0	CS1AD1
2	2	CS2AD0	CS1AD1
1	3	CS3AD0	CS1AD1
2	3	CS4AD0	CS1AD1

Tabla 10.4: Configuraciones experimentales.

Teniendo presente las características estocásticas del sistema, cada configuración se debe ejecutar varias veces para tener validez estadística y así realizar el respectivo análisis descrito en la sección 10.7.2 [258].

Con el fin de adquirir los datos experimentales se efectúa la simulación para cada configuración 10 veces con una duración de 10 horas. De esta forma se obtienen simulaciones con señales de 600 minutos y 10 cambios para la potencia de carga. Los datos de

la potencia de carga se generan de forma aleatoria con datos uniformemente distribuidos de 0 a 1000KW.

Para la implementación del sistema de control neuro-difuso adaptativo se tienen diferentes configuraciones dependiendo de la arquitectura y las funciones de pertenencia a utilizar, las cuales se pueden apreciar en la tabla 10.5. El detalle de estas configuraciones se puede apreciar en la sección 9.7.

Configuración Sistema BBR	Planta			Controlador		
	Arquitectura	Gaussianas	Sigmoidales	Arquitectura	Gaussianas	Sigmoidales
A2P1SA2C1S	II	0	1	II	0	1
A2P2SA2C2S	II	0	2	II	0	2
A2P1G2SA2C1G2S	II	1	2	II	1	2
A2P4SA2C4S	II	0	4	II	0	4
A2P2GA2C2S	II	2	0	II	0	2
A2P2SA2C2G	II	0	2	II	2	0
A1P4GA2C2G	I	4	0	II	2	0
A1P4GA2C2S	I	4	0	II	0	2
A1P4SA2C2G	I	0	4	II	2	0
A1P4SA2C2S	I	0	4	II	0	2
A2P2GA1C4G	II	2	0	I	4	0
A2P2GA1C4S	II	2	0	I	0	4
A2P2SA1C4G	II	0	2	I	4	0
A2P2SA1C4S	II	0	2	I	0	4
A1P4GA1C4G	I	4	0	I	4	0
A1P4GA1C4S	I	4	0	I	0	4
A1P4SA1C4G	I	0	4	I	4	0
A1P4SA1C4S	I	0	4	I	0	4

Tabla 10.5: Configuraciones de las arquitecturas de los sistemas difusos.

Adicionalmente, para completar el diseño experimental se consideran los casos cuando se conectan varios generadores por lo cual se tienen las configuraciones:

- 1G: Implementación con un generador.
- 3G: Implementación con tres generadores.

Configuración del sistema de distribución para uno y tres generadores

En primer lugar se aprecia el efecto que tiene el sistema de control al conectar un generador localizado en el nodo 18, por su parte, la carga variable se presenta en el nodo 17 (considerando lo reportado en [257]). Para un generador, el índice de desempeño utilizado consiste en el error cuadrático medio MSE.

Un aspecto de importancia en los sistemas de generación distribuida consiste en la capacidad de conectar y desconectar diferentes generadores en la red de distribución sin llegar a presentar mayores alteraciones en los valores de voltaje de los nodos. Con el fin de observar el desempeño que tiene el sistema de control neuro-difuso adaptativo, se incorporan varios generadores a la red de distribución mostrada en la figura 10.9. Para esto se consideraran tres generadores localizados en los nodos:

- Generador 1: Nodo 18.
- Generador 2: Nodo 25.
- Generador 3: Nodo 33.

Por su parte, la variación de la carga se propone en los nodos:

- Carga variable 1: Nodo 17.
- Carga variable 2: Nodo 24.
- Carga variable 3: Nodo 32.

Los datos para la variación de la carga se generan aleatoriamente de forma uniforme con valores entre $0KW$ y $1000KW$. Los grupos experimentales se toman de la misma manera que en la implementación para un generador. Teniendo presente que por cada generador se tiene un valor de error cuadrático medio MSE , entonces el índice de desempeño considerado para realizar el análisis estadístico consiste en la suma del MSE para los tres generadores:

$$MSE_T = MSE_{G1} + MSE_{G2} + MSE_{G3} \quad (10.2)$$

10.7.5. Resultados para un generador

En esta sección se muestran los resultados obtenidos para el sistema de control propuesto cuando se tiene un generador en el sistema de distribución. En una primera instancia se realiza el proceso demostrativo del análisis de resultados y los resultados cualitativos, posteriormente se muestra el resumen de todos los resultados experimentales obtenidos. Como caso demostrativo de la simulación se toma la configuración A2P2SA2C2S.

Resultados de la configuración A2P2SA2C2S

Luego de realizar las respectivas ejecuciones (10 simulaciones de 600 minutos) se calcula el respectivo error cuadrático medio MSE . El resumen de los resultados de las 10 corridas para cada configuración se puede apreciar en la tabla 10.6, donde se tiene el valor mínimo, máximo, la desviación estándar y el promedio.

Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 794 49	0.001 542 2	0.000 219 46	0.001 065 6
CS2AD0	0.000 772	0.001 501 6	0.000 214 33	0.001 036 3
CS3AD0	0.000 786 94	0.001 528 1	0.000 217 54	0.001 055 7
CS4AD0	0.000 770 9	0.001 499 1	0.000 213 86	0.001 034 7
CS1AD1	8.1748×10^{-5}	0.000 158 36	2.2542×10^{-5}	0.000 109 38
CS2AD1	8.0108×10^{-5}	0.000 155 07	2.2113×10^{-5}	0.000 107 02
CS3AD1	8.0855×10^{-5}	0.000 156 75	2.2326×10^{-5}	0.000 108 25
CS4AD1	7.9557×10^{-5}	0.000 154 1	2.197×10^{-5}	0.000 106 36

Tabla 10.6: Resumen de los valores estadísticos para los resultados obtenidos.

Con los datos obtenidos de grupos experimentales se realizan las respectivas pruebas estadísticas descritas anteriormente. En primer lugar, los resultados de la prueba de normalidad para cada grupo experimental se pueden apreciar en la tabla 10.7, donde también se observa el cumplimiento del requerimiento de normalidad

Configuración	p -value
CS1AD0	0.622 52
CS2AD0	0.619 78
CS3AD0	0.622 24
CS4AD0	0.620 23
CS1AD1	0.622 89
CS2AD1	0.617 68
CS3AD1	0.622 80
CS4AD1	0.617 94

Tabla 10.7: Resumen de la prueba de normalidad.

Posteriormente, al realizar la prueba de homocedasticidad se tiene un p -value de $8,3933 \times 10^{-6}$ lo cual muestra el no cumplimiento de igualdad de varianza. Considerando los resultados de la prueba de normalidad y homocedasticidad, se procede a realizar una prueba de Kruskal-Wallis para observar si existe diferencia significativa entre los grupos experimentales; al efectuar esta prueba se obtiene un p -value de $1,9018 \times 10^{-10}$, lo cual indica que existe diferencia entre grupos. Para determinar las diferencias se procede a la prueba no paramétrica de comparaciones múltiples de Bonferroni, obteniendo el resultado mostrado en la figura 10.11. Esta prueba se lleva a cabo con un nivel de significancia de 0,05.

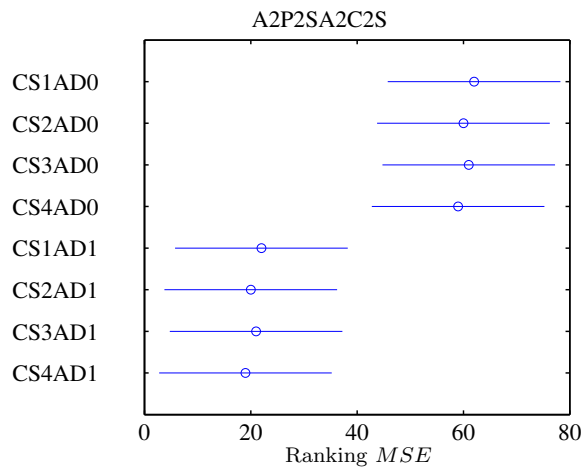


Figura 10.11: Resultado de las comparaciones múltiples.

En la figura 10.11, si los intervalos de dos grupos se traslapan, no existe diferencia estadística significativa entre estos. En estos resultados se aprecia un mejor desempeño cuando se desarrolla el proceso de adaptación del controlador.

Resultados de simulación

Para mostrar de forma cualitativa el comportamiento del sistema de control se toma la configuración CS2AD1; en la figura 10.12 se presenta, de forma gráfica, la respuesta del sistema con el controlador convencional y la implementación con el sistema adaptativo; se aprecia además que el controlador adaptativo realiza ajustes para buscar que la salida del sistema llegue a la referencia después que ha ocurrido la variación de la carga.

El detalle de los ajustes hechos por el sistema adaptativo se puede apreciar en la figura 10.13. Es de notar en la simulación el momento cuando se efectúan los respectivos ajustes progresivos del controlador para corregir el cambio que existe al variar la carga.

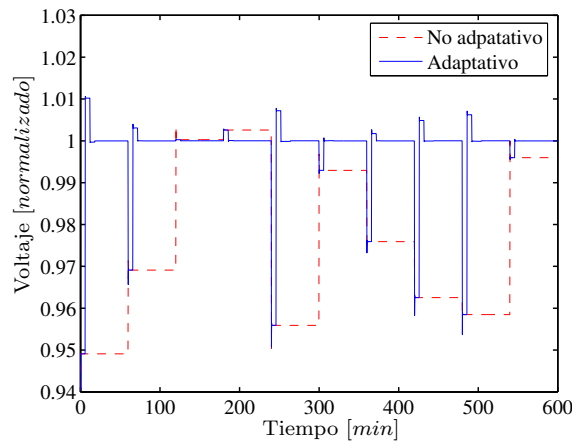


Figura 10.12: Respuesta del sistema de control con y sin proceso de adaptación.

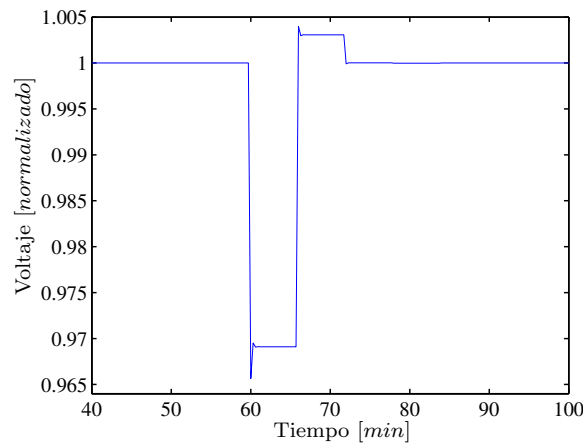


Figura 10.13: Detalle de la respuesta del sistema de control adaptativo.

10.7.6. Análisis estadístico de resultados para un generador

Para la implementación con un generador, el resumen de los resultados de las corridas para cada configuración se presenta en la tabla 10.8.

Configuración A2P1SA2C1S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 814 25	0.001 580 9	0.000 225 04	0.001 092 2
CS2AD0	0.000 796 96	0.001 549 2	0.000 220 9	0.001 069 5
CS3AD0	0.000 807 91	0.001 568 7	0.000 223 29	0.001 083 7
CS4AD0	0.000 794 62	0.001 544 2	0.000 220 1	0.001 066 3
CS1AD1	8.3315×10^{-5}	0.000 161 54	2.3007×10^{-5}	0.000 111 57
CS2AD1	8.1686×10^{-5}	0.000 158 29	2.2561×10^{-5}	0.000 109 29
CS3AD1	8.2568×10^{-5}	0.000 160 15	2.281×10^{-5}	0.000 110 6
CS4AD1	8.1343×10^{-5}	0.000 157 69	2.2473×10^{-5}	0.000 108 89

Configuración A2P1G2SA2C1G2S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 802 95	0.001 556 1	0.000 220 88	0.001 076 3
CS2AD0	0.000 818 81	0.001 584 1	0.000 224 31	0.001 096 9
CS3AD0	0.000 798 01	0.001 546 5	0.000 219 52	0.001 069 7
CS4AD0	0.000 815 27	0.001 577 1	0.000 223 3	0.001 092 1
CS1AD1	8.5232×10^{-5}	0.000 165 89	2.3694×10^{-5}	0.000 114 41
CS2AD1	8.4258×10^{-5}	0.000 164 42	2.3435×10^{-5}	0.000 113 48
CS3AD1	8.2278×10^{-5}	0.000 160 04	2.28×10^{-5}	0.000 110 48
CS4AD1	8.4044×10^{-5}	0.000 163 97	2.337×10^{-5}	0.000 113 17

Configuración A2P2GA2C2S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 794 49	0.001 542 2	0.000 219 46	0.001 065 6
CS2AD0	0.000 772	0.001 501 6	0.000 214 33	0.001 036 3
CS3AD0	0.000 786 94	0.001 528 1	0.000 217 54	0.001 055 7
CS4AD0	0.000 770 9	0.001 499 1	0.000 213 86	0.001 034 7
CS1AD1	8.0765×10^{-5}	0.000 156 42	2.2244×10^{-5}	0.000 108 09
CS2AD1	7.8969×10^{-5}	0.000 152 86	2.1779×10^{-5}	0.000 105 54
CS3AD1	7.9867×10^{-5}	0.000 154 8	2.2029×10^{-5}	0.000 106 95
CS4AD1	7.8455×10^{-5}	0.000 151 92	2.1638×10^{-5}	0.000 104 92

Configuración A1P4GA2C2G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 939 61	0.001 812 1	0.000 255 56	0.001 256 7
CS2AD0	0.001 019	0.001 945 7	0.000 270 85	0.001 356 8
CS3AD0	0.000 919 62	0.001 778 1	0.000 251 63	0.001 231 3
CS4AD0	0.000 974 48	0.001 871 4	0.000 262 44	0.001 301
CS1AD1	9.5251×10^{-5}	0.000 185 24	2.6271×10^{-5}	0.000 128 14
CS2AD1	$0.000 103 59$	0.000 201 09	2.8285×10^{-5}	0.000 139 63
CS3AD1	9.2969×10^{-5}	0.000 181 01	2.5737×10^{-5}	0.000 125 08
CS4AD1	0.000 100 05	0.000 195 06	2.7627×10^{-5}	0.000 134 99

Configuración A1P4SA2C2G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 939 61	0.001 812 1	0.000 255 56	0.001 256 7
CS2AD0	0.001 019	0.001 945 7	0.000 270 85	0.001 356 8
CS3AD0	0.000 919 62	0.001 778 1	0.000 251 63	0.001 231 3
CS4AD0	0.000 974 48	0.001 871 4	0.000 262 44	0.001 301
CS1AD1	9.755×10^{-5}	0.000 189 74	2.6951×10^{-5}	0.000 131 13
CS2AD1	$0.000 103 24$	0.000 201 25	2.8392×10^{-5}	0.000 139 59
CS3AD1	9.716×10^{-5}	0.000 189 33	2.7055×10^{-5}	0.000 130 7
CS4AD1	0.000 100 8	0.000 196 53	2.7861×10^{-5}	0.000 135 98

Configuración A2P2GA1C4G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS2AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS3AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS4AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS1AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS2AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS3AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS4AD1	0.011 865	0.016 341	0.001 303 5	0.013 545

Configuración A2P2SA1C4G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 794 49	0.001 542 2	0.000 219 46	0.001 065 6
CS2AD0	0.000 772	0.001 501 6	0.000 214 33	0.001 036 3
CS3AD0	0.000 786 94	0.001 528 1	0.000 217 54	0.001 055 7
CS4AD0	0.000 770 9	0.001 499 1	0.000 213 86	0.001 034 7
CS1AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS2AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS3AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS4AD1	0.011 865	0.016 341	0.001 303 5	0.013 545

Configuración A1P4GA1C4G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS2AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS3AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS4AD0	0.011 865	0.016 341	0.001 303 5	0.013 545
CS1AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS2AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS3AD1	0.011 865	0.016 341	0.001 303 5	0.013 545
CS4AD1	0.011 865	0.016 341	0.001 303 5	0.013 545

Configuración A1P4SA1C4S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 797 55	0.001 552	0.000 221 7	0.001 070 7
CS2AD0	0.000 774 99	0.001 513 4	0.000 217 39	0.001 041 6
CS3AD0	0.000 803 1	0.001 561 1	0.000 222 61	0.001 077 7
CS4AD0	0.000 776 22	0.001 514 7	0.000 217 3	0.001 043
CS1AD1	8.1653×10^{-5}	0.000 157 62	2.2429×10^{-5}	0.000 108 9
CS2AD1	8.016×10^{-5}	0.000 153 88	2.1921×10^{-5}	0.000 106 28
CS3AD1	8.2074×10^{-5}	0.000 158 68	2.2577×10^{-5}	0.000 109 63
CS4AD1	8.113×10^{-5}	0.000 156 3	2.2298×10^{-5}	0.000 107 85

Configuración A1P4SA1C4S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.000 794 49	0.001 542 2	0.000 219 46	0.001 065 6
CS2AD0	0.000 772	0.001 501 6	0.000 214 33	0.001 036 3
CS3AD0	0.000 786 94	0.001 528 1	0.000 217 54	0.001 055 7
CS4AD0	0.000 770 9	0.001 499 1	0.000 213 86	0.001 034 7
CS1AD1	8.367×10^{-5}	0.000 161 56	2.3029×10^{-5}	0.000 111 51
CS2AD1	8.0365×10^{-5}	0.000 154 28	2.1983×10^{-5}	0.000 106 54
CS3AD1	8.5661×10^{-5}	0.000 165 73	2.3674×10^{-5}	0.000 114 36
CS4AD1	8.1296×10^{-5}	0.000 158	2.2682×10^{-5}	0.000 108 74

Tabla 10.8: Resumen de los valores estadísticos para los resultados obtenidos.

Los resultados de la prueba de normalidad se tienen en la tabla 10.9 y los resultados de la prueba de homocedasticidad, Kruskal-Wallis y ANOVA se muestran en la tabla 10.10. Finalmente, en la figura 10.14 se presenta el resultado de la comparación entre grupos. De esta forma se puede establecer que las configuraciones A2P2GA1C4G, A1P4GA1C4G y A1P4SA1C4G no logran realizar adecuadamente el proceso adaptativo, por lo cual, no se aprecia diferencia significativa entre los casos con y sin adaptación. En estos resultados también se aprecia que no se presenta diferencia para las configuraciones donde se cambia el número de retardos de la entrada y la salida en la estructura de los sistemas neuro-difusos. Es de notar que de 18 configuraciones experimentales de los sistemas neuro-difusos se logra el comportamiento adaptativo en 15 configuraciones.

Configuración	CS1AD0	CS2AD0	CS3AD0	CS4AD0	CS1AD1	CS2AD1	CS3AD1	CS4AD1
A2P1SA2C1S	0.62254	0.62092	0.62255	0.62130	0.62384	0.62067	0.62404	0.62143
A2P2SA2C2S	0.62252	0.61978	0.62224	0.62023	0.62289	0.61768	0.62280	0.61794
A2P1G2SA2C1G2S	0.62493	0.62700	0.62492	0.62706	0.63291	0.63372	0.62914	0.63367
A2P4SA2C4S	0.62092	0.61829	0.62136	0.61922	0.62193	0.61285	0.62214	0.61611
A2P2GA2C2S	0.62252	0.61978	0.62224	0.62023	0.62066	0.61492	0.62063	0.61615
A2P2SA2C2G	0.62252	0.61978	0.62224	0.62023	0.63925	0.65369	0.63617	0.64831
A1P4GA2C2G	0.63017	0.63855	0.62784	0.63427	0.63742	0.65119	0.63412	0.64594
A1P4GA2C2S	0.62252	0.61978	0.62224	0.62023	0.62103	0.61507	0.62050	0.61600
A1P4SA2C2G	0.63017	0.63855	0.62784	0.63427	0.63954	0.65620	0.65179	0.65033
A1P4SA2C2S	0.62252	0.61978	0.62224	0.62023	0.62456	0.61713	0.63807	0.62449
A2P2GA1C4G	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317
A2P2GA1C4S	0.62252	0.61978	0.62224	0.62023	0.61281	0.59598	0.61639	0.60170
A2P2SA1C4G	0.62252	0.61978	0.62224	0.62023	0.54317	0.54317	0.54317	0.54317
A2P2SA1C4S	0.62252	0.61978	0.62224	0.62023	0.61333	0.59720	0.61667	0.60361
A1P4GA1C4G	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317
A1P4GA1C4S	0.61903	0.61307	0.62059	0.61465	0.61271	0.59663	0.61608	0.60204
A1P4SA1C4S	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317	0.54317
A1P4SA1C4S	0.62252	0.61978	0.62224	0.62023	0.61625	0.59822	0.62789	0.61874

Tabla 10.9: Resumen de la prueba de normalidad (p -value).

Configuración	Cumplen normalidad	Homocedasticidad	Kruskal-wallis	ANOVA
A2P1SA2C1S	8	8.2207×10^{-6}	1.9282×10^{-10}	0
A2P2SA2C2S	8	8.3933×10^{-6}	1.9018×10^{-10}	0
A2P1G2SA2C1G2S	8	8.5798×10^{-6}	1.8719×10^{-10}	0
A2P4SA2C4S	8	8.5676×10^{-6}	1.9174×10^{-10}	0
A2P2GA2C2S	8	8.0398×10^{-6}	1.8838×10^{-10}	0
A2P2SA2C2G	8	1.6915×10^{-5}	1.5308×10^{-10}	0
A1P4GA2C2G	8	7.8682×10^{-6}	1.32×10^{-10}	0
A1P4GA2C2S	8	8.211×10^{-6}	1.9018×10^{-10}	0
A1P4SA2C2G	8	8.4393×10^{-6}	1.495×10^{-10}	0
A1P4SA2C2S	8	8.899×10^{-6}	1.7482×10^{-10}	0
A2P2GA1C4G	8	1	1	1
A2P2GA1C4S	8	8.0983×10^{-6}	1.8251×10^{-10}	0
A2P2SA1C4G	8	5.9863×10^{-6}	1.9321×10^{-10}	0
A2P2SA1C4S	8	8.6837×10^{-6}	1.8084×10^{-10}	0
A1P4GA1C4G	8	1	1	1
A1P4GA1C4S	8	8.4675×10^{-6}	1.8173×10^{-10}	0
A1P4SA1C4G	8	1	1	1
A1P4SA1C4S	8	9.0551×10^{-6}	1.7272×10^{-10}	0

Tabla 10.10: Resumen de las pruebas estadísticas.

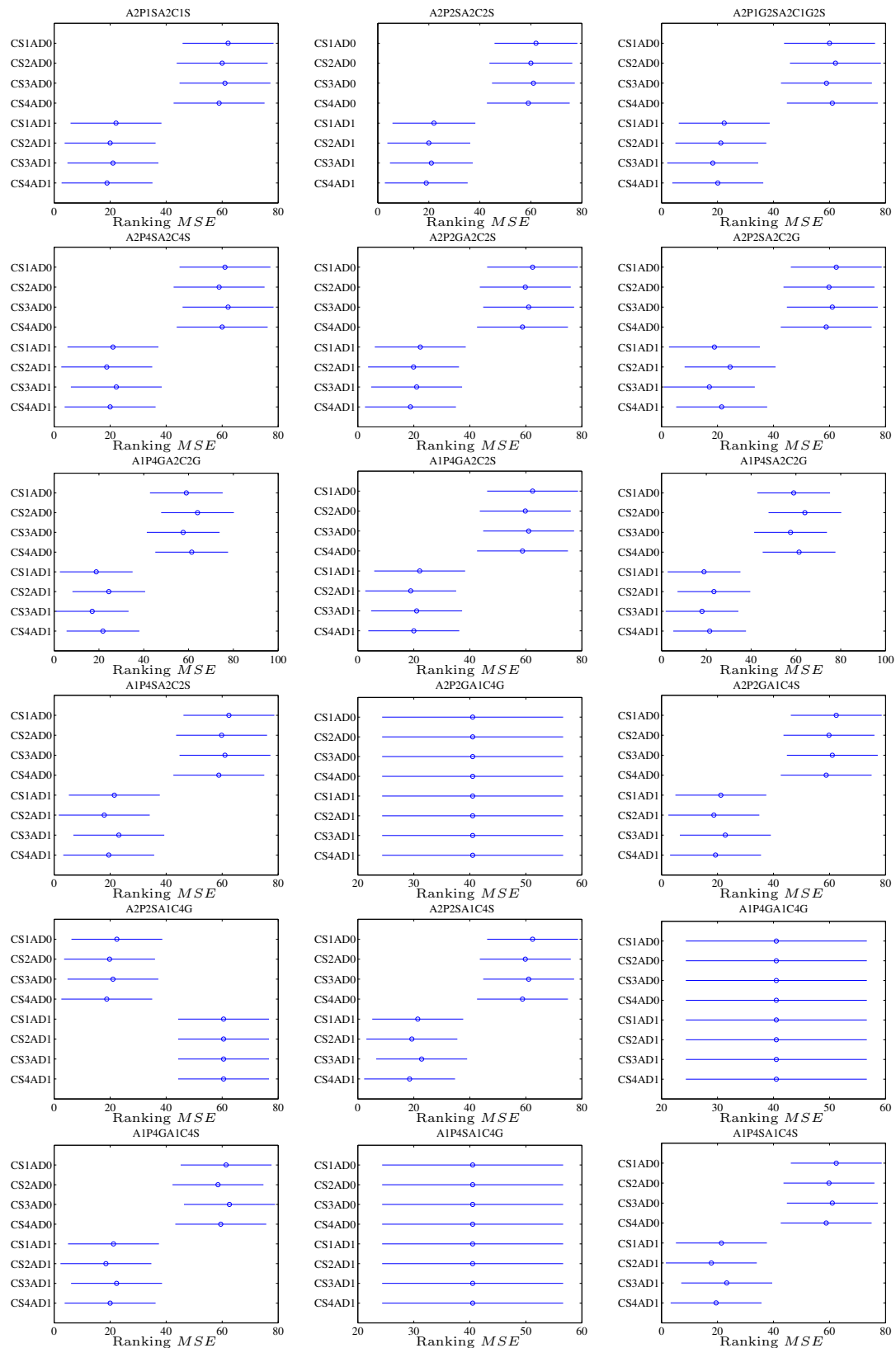


Figura 10.14: Resultado de las comparaciones múltiples.

10.7.7. Resultados para tres generadores

En esta sección se observan los resultados de los sistemas de control cuando se presentan tres generadores en el sistema de distribución. De forma demostrativa se toma el caso para la configuración A2P2SA2C2S.

Resultados de la configuración A2P2SA2C2S

Para esta implementación los valores mínimo, máximo, desviación estándar y promedio de los resultados de las 10 corridas para cada configuración se pueden apreciar en la tabla 10.11.

Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 520 2	0.003 031 6	0.000 517 18	0.002 292 3
CS2AD0	0.001 549 6	0.003 085 4	0.000 525 39	0.002 334 2
CS3AD0	0.001 546	0.003 080 8	0.000 524 23	0.002 33
CS4AD0	0.001 535 6	0.003 059 5	0.000 520 81	0.002 314 4
CS1AD1	8.3829×10^{-5}	0.000 161 81	2.6401×10^{-5}	0.000 124 98
CS2AD1	8.5712×10^{-5}	0.000 165 11	2.713×10^{-5}	0.000 126 64
CS3AD1	8.4852×10^{-5}	0.000 163 98	2.6979×10^{-5}	0.000 125 59
CS4AD1	8.4648×10^{-5}	0.000 163 33	2.687×10^{-5}	0.000 125 21

Tabla 10.11: Resumen de los valores estadísticos para los resultados obtenidos.

Con los datos obtenidos se realizan las respectivas pruebas estadísticas, los resultados de la prueba de normalidad para cada grupo experimental se pueden apreciar en la tabla 10.12, donde se cumple con el requerimiento de normalidad.

Configuración	p -value
CS1AD0	0.38164
CS2AD0	0.38300
CS3AD0	0.39065
CS4AD0	0.39015
CS1AD1	0.63348
CS2AD1	0.42060
CS3AD1	0.42941
CS4AD1	0.42671

Tabla 10.12: Resumen de la prueba de normalidad.

Para la prueba de homocedasticidad se tiene un p -value de $9,6967 \times 10^{-13}$ por lo cual no se cumple la igualdad de varianzas. Considerando los resultados de la prueba de normalidad y homocedasticidad se procede a realizar la prueba de Kruskal-Wallis obteniendo un p -value de $1,8195 \times 10^{-10}$ mostrando que existe diferencia entre grupos.

Para establecer las diferencias entre grupos, con un nivel de significancia de 0,05 se lleva a cabo la prueba no paramétrica de comparaciones múltiples de Bonferroni, obteniendo el resultado mostrado en la figura 10.15, donde se aprecia de forma estadística que el sistema adaptativo presenta un mejor desempeño.

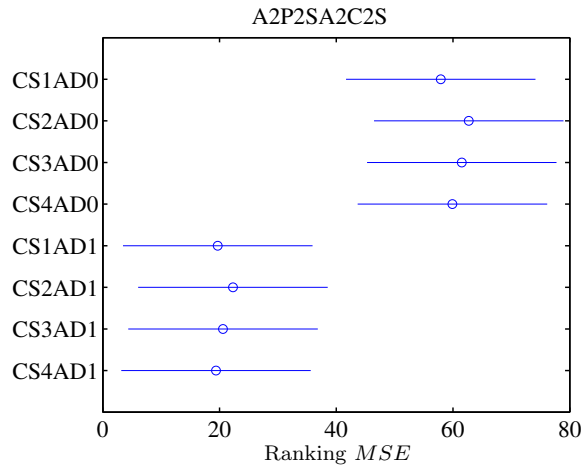


Figura 10.15: Resultado de las comparaciones múltiples.

Resultados de simulación

Con el fin de mostrar la simulación del sistema de control se toma la configuración CS2AD1; la figura 10.16 muestra la respuesta del sistema para los tres generadores con el control adaptativo. Por su parte, la figura 10.17 presenta la simulación cuando no se realiza el proceso adaptativo.

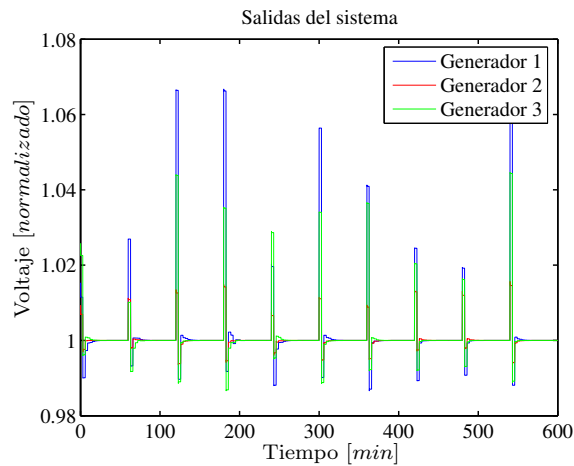


Figura 10.16: Respuesta del sistema de control con proceso de adaptación.

Es de notar en la figura 10.16, que el controlador adaptativo hace varios ajustes para lograr que la salida del sistema llegue a la referencia después que ha ocurrido la variación de la carga.

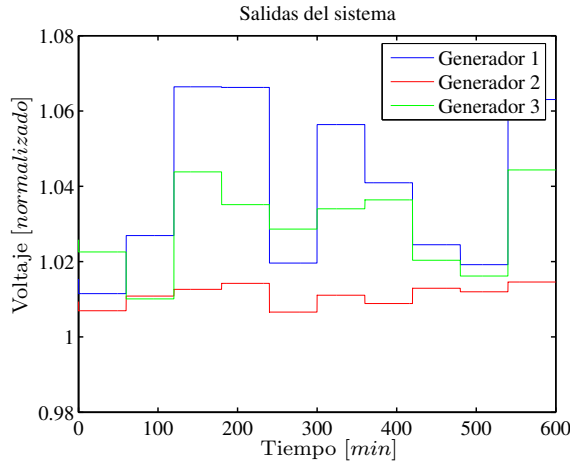


Figura 10.17: Respuesta del sistema de control sin proceso de adaptación.

El detalle de los ajustes hechos por el sistema adaptativo cuando existe variación de la carga se puede apreciar en la figura 10.18. Es de señalar que se efectúan varios ajustes para lograr la regulación de los voltajes en los generadores.

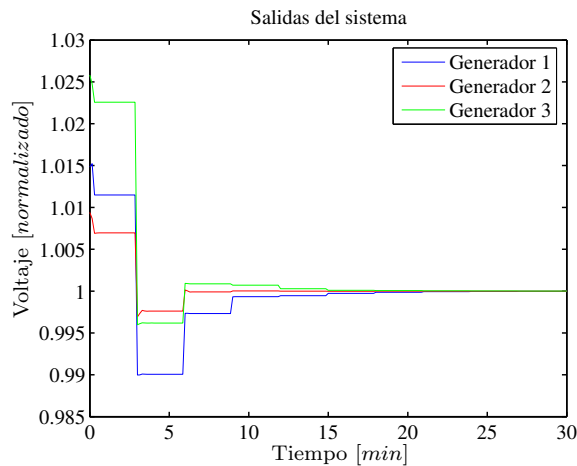


Figura 10.18: Detalle de la respuesta del sistema de control adaptativo.

10.7.8. Análisis estadístico de resultados para tres generadores

Para esta configuración, el resumen de los resultados de las ejecuciones para cada configuración se presenta en la tabla 10.13. Los resultados de la prueba de normalidad se tienen en la tabla 10.14 y los resultados de la prueba de homocedasticidad, Kruskal-Wallis y ANOVA se muestran en la tabla 10.15. Finalmente, en la figura 10.19 se presenta el resultado de la comparación entre grupos.

Con estos resultados y análisis estadísticos se puede establecer que para las configuraciones A2P2GA1C4G, A2P2GA1C4S, A2P2SA1C4G, A1P4GA1C4G y A1P4SA1C4G no se presenta diferencia significativa entre el caso adaptativo y no adaptativo. Para el caso experimental A1P4SA1C4S se aprecia que solamente las configuraciones CS3AD1 y CS4AD1 logran el proceso adaptativo.

Configuración A2P2SA1C4S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 539	0.003 064 1	0.000 521 8	0.002 317 7
CS2AD0	0.001 538 2	0.003 061 5	0.000 521 34	0.002 316 2
CS3AD0	0.001 531 6	0.003 048	0.000 519 41	0.002 306 1
CS4AD0	0.001 535 5	0.003 055 9	0.000 520 43	0.002 312 1
CS1AD1	8.4867×10^{-5}	0.000 163 88	2.4438×10^{-5}	0.000 130 63
CS2AD1	8.4993×10^{-5}	24.776	7.8348	2.4777
CS3AD1	8.4392×10^{-5}	0.000 163 05	2.6684×10^{-5}	0.000 125 73
CS4AD1	8.4826×10^{-5}	0.000 163 87	2.68×10^{-5}	0.000 126 26

Configuración A2P2SA2C2S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 520 2	0.003 031 6	0.000 517 18	0.002 292 3
CS2AD0	0.001 549 6	0.003 085 4	0.000 525 39	0.002 334 2
CS3AD0	0.001 546	0.003 080 8	0.000 524 23	0.002 33
CS4AD0	0.001 535 6	0.003 059 5	0.000 520 81	0.002 314 4
CS1AD1	8.5031×10^{-5}	0.000 165 89	2.7576×10^{-5}	0.000 126 44
CS2AD1	8.1122×10^{-5}	0.000 156 51	2.5401×10^{-5}	0.000 119 5
CS3AD1	7.9652×10^{-5}	0.000 153 06	2.5211×10^{-5}	0.000 118 02
CS4AD1	7.8907×10^{-5}	0.000 152 49	2.491×10^{-5}	0.000 116 64

Configuración A2P1G2SA2C1G2S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 548 4	0.003 082 2	0.000 524 69	0.002 332
CS2AD0	0.001 477 4	0.002 942 3	0.000 499 8	0.002 225 1
CS3AD0	0.001 447	0.002 885 4	0.000 491 08	0.002 182
CS4AD0	0.001 442 9	0.002 878 7	0.000 489 8	0.002 176 1
CS1AD1	8.4865×10^{-5}	0.000 161 78	2.6237×10^{-5}	0.000 124 59
CS2AD1	8.1122×10^{-5}	0.000 156 51	2.5401×10^{-5}	0.000 119 5
CS3AD1	7.9652×10^{-5}	0.000 153 06	2.5211×10^{-5}	0.000 118 02
CS4AD1	7.8907×10^{-5}	0.000 152 49	2.491×10^{-5}	0.000 116 64

Configuración A2P2GA2C2S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 520 2	0.003 031 6	0.000 517 18	0.002 292 3
CS2AD0	0.001 549 6	0.003 085 4	0.000 525 39	0.002 334 2
CS3AD0	0.001 546	0.003 080 8	0.000 524 23	0.002 33
CS4AD0	0.001 535 6	0.003 059 5	0.000 520 81	0.002 314 4
CS1AD1	8.5031×10^{-5}	0.000 165 89	2.7576×10^{-5}	0.000 126 44
CS2AD1	8.1122×10^{-5}	0.000 156 51	2.5401×10^{-5}	0.000 119 5
CS3AD1	7.9652×10^{-5}	0.000 153 06	2.5211×10^{-5}	0.000 118 02
CS4AD1	7.8907×10^{-5}	0.000 152 49	2.491×10^{-5}	0.000 116 64

Configuración A2P2SA2C2G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 520 2	0.003 031 6	0.000 517 18	0.002 292 3
CS2AD0	0.001 549 6	0.003 085 4	0.000 525 39	0.002 334 2
CS3AD0	0.001 546	0.003 080 8	0.000 524 23	0.002 33
CS4AD0	0.001 535 6	0.003 059 5	0.000 520 81	0.002 314 4
CS1AD1	8.5454×10^{-5}	0.000 164 93	2.6942×10^{-5}	0.000 127 06
CS2AD1	8.5934×10^{-5}	0.000 165 35	2.7133×10^{-5}	0.000 126 81
CS3AD1	8.817×10^{-5}	0.000 170 42	2.7989×10^{-5}	0.000 130 33
CS4AD1	8.5068×10^{-5}	0.000 163 94	2.6934×10^{-5}	0.000 125 6

Configuración A1P4GA2C2G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 577 1	0.003 146 9	0.000 533 99	0.002 378 8
CS2AD0	0.001 520 2	0.003 022 2	0.000 514 41	0.002 287 1
CS3AD0	0.001 467 2	0.002 911 2	0.000 496 3	0.002 204 3
CS4AD0	0.001 607 8	0.003 213 7	0.000 544 73	0.002 427 8
CS1AD1	8.9423×10^{-5}	0.000 173 49	2.8506×10^{-5}	0.000 132 48
CS2AD1	8.4942×10^{-5}	0.000 164 67	2.7252×10^{-5}	0.000 125 98
CS3AD1	8.1433×10^{-5}	0.000 158 73	2.6438×10^{-5}	0.000 121 1
CS4AD1	9.3159×10^{-5}	0.000 180 17	2.9316×10^{-5}	0.000 137 31

Configuración A1P4SA2C2G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 606	0.003 185 5	0.000 541 35	0.002 411 8
CS2AD0	0.001 580 2	0.003 151 8	0.000 535 06	0.002 382 7
CS3AD0	0.001 549	0.003 081 8	0.000 523 93	0.002 331 5
CS4AD0	0.001 579 9	0.003 138 1	0.000 533 68	0.002 375 2
CS1AD1	8.4356×10^{-5}	0.000 162 36	2.7557×10^{-5}	0.000 126 01
CS2AD1	8.7661×10^{-5}	0.000 168 3	2.8081×10^{-5}	0.000 131 4
CS3AD1	8.5013×10^{-5}	0.000 162 04	2.6084×10^{-5}	0.000 125 54
CS4AD1	8.5897×10^{-5}	0.000 164 95	2.6625×10^{-5}	0.000 127 26

Configuración A2P2GA1C4G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS2AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS3AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS4AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS1AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS2AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS3AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS4AD1	0.020 651	0.025 987	0.001 865 8	0.023 001

Configuración A2P2SA1C4G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 520 2	0.003 031 6	0.000 517 18	0.002 292 3
CS2AD0	0.001 549 6	0.003 085 4	0.000 525 39	0.002 334 2
CS3AD0	0.001 546	0.003 080 8	0.000 524 23	0.002 33
CS4AD0	0.001 535 6	0.003 059 5	0.000 520 81	0.002 314 4
CS1AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS2AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS3AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS4AD1	0.020 651	0.025 987	0.001 865 8	0.023 001

Configuración A1P4GA1C4G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS2AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS3AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS4AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS1AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS2AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS3AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS4AD1	0.020 651	0.025 987	0.001 865 8	0.023 001

Configuración A1P4SA1C4G				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS2AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS3AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS4AD0	0.020 651	0.025 987	0.001 865 8	0.023 001
CS1AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS2AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS3AD1	0.020 651	0.025 987	0.001 865 8	0.023 001
CS4AD1	0.020 651	0.025 987	0.001 865 8	0.023 001

Configuración A2P2GA1C4S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 520 2	0.003 031 6	0.000 517 18	0.002 292 3
CS2AD0	0.001 549 6	0.003 085 4	0.000 525 39	0.002 334 2
CS3AD0	0.001 546	0.003 080 8	0.000 524 23	0.002 33
CS4AD0	0.001 535 6	0.003 059 5	0.000 520 81	0.002 314 4
CS1AD1	8.9671×10^{-5}	0.000 169 45	2.7688×10^{-5}	0.000 131 74
CS2AD1	9.2197×10^{-5}	0.000 263 98	4.9895×10^{-5}	0.000 149 91
CS3AD1	8.0898×10^{-5}	0.000 155 84	2.5913×10^{-5}	0.000 122
CS4AD1	7.5389×10^{-5}	0.000 149 68	2.5033×10^{-5}	0.000 114 05

Configuración A1P4GA1C4S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 425	0.002 843 1	0.000 484 17	0.002 149 3
CS2AD0	0.001 420 7	0.002 827 7	0.000 482	0.002 138 9
CS3AD0	0.001 367 8	0.002 724 2	0.000 464 62	0.002 059 3
CS4AD0	0.001 359 6	0.002 714 7	0.000 462 88	0.002 050 9
CS1AD1	8.1057×10^{-5}	0.000 157 06	2.594×10^{-5}	0.000 120 14
CS2AD1	7.9577×10^{-5}	0.000 154 39	2.5596×10^{-5}	0.000 118 04
CS3AD1	7.6138×10^{-5}	0.000 149 21	2.4907×10^{-5}	0.000 113 52
CS4AD1	7.9715×10^{-5}	0.000 152 55	2.5058×10^{-5}	0.000 117 17

Configuración A1P4SA1C4S				
Configuración	Mínimo	Máximo	Desviación STD	Promedio
CS1AD0	0.001 520 2	0.003 031 6	0.000 517 18	0.002 292 3
CS2AD0	0.001 549 6	0.003 085 4	0.000 525 39	0.002 334 2
CS3AD0	0.001 546	0.003 080 8	0.000 524 23	0.002 33
CS4AD0	0.001 535 6	0.003 059 5	0.000 520 81	0.002 314 4
CS1AD1	8.183×10^{-5}	0.000 156 53	2.5628×10^{-5}	0.000 121 74
CS2AD1	1.3124	1.3478	0.010 935	1.328
CS3AD1	7.853×10^{-5}	0.000 150 08	2.4483×10^{-5}	0.000 115 7
CS4AD1	7.478×10^{-5}	0.000 143 97	2.3574×10^{-5}	0.000 110 84

Tabla 10.13: Resumen de los valores estadísticos para los resultados obtenidos.

En estos resultados también se observa que en la mayoría de los casos experimentales no se presenta diferencia para las configuraciones donde se cambia el número de retardos de la entrada y la salida en la estructura de los sistemas neuro-difusos. Es de notar que de 18 configuraciones experimentales de los sistemas neuro-difusos se logra el comportamiento adaptativo en 12,5 configuraciones, lo cual muestra que al incrementar el número de generadores hay más configuraciones que no operan correctamente.

Configuración	CS1AD0	CS2AD0	CS3AD0	CS4AD0	CS1AD1	CS2AD1	CS3AD1	CS4AD1
A2P1SA2C1S	0.38089	0.37981	0.37640	0.37920	0.61214	0.00237	0.51565	0.52595
A2P2SA2C2S	0.38164	0.38300	0.39065	0.39015	0.63348	0.42060	0.42941	0.42671
A2P1G2SA2C1G2S	0.38397	0.39511	0.39242	0.39559	0.43875	0.49285	0.42496	0.46168
A2P4SA2C4S	0.39162	0.37812	0.39676	0.36010	0.43050	0.42061	0.45056	0.39251
A2P2GA2C2S	0.38164	0.38300	0.39065	0.39015	0.40145	0.41364	0.73615	0.49739
A2P2SA2C2G	0.38164	0.38300	0.39065	0.39015	0.58622	0.42162	0.39922	0.42285
A1P4GA2C2G	0.41192	0.37617	0.36123	0.42966	0.44515	0.41612	0.39327	0.45484
A1P4GA2C2S	0.38164	0.38300	0.39065	0.39015	0.41555	0.41493	0.43015	0.43229
A1P4SA2C2G	0.37128	0.40706	0.38564	0.37438	0.44101	0.41371	0.55701	0.39727
A1P4SA2C2S	0.38164	0.38300	0.39065	0.39015	0.43602	0.41748	0.45954	0.43100
A2P2GA1C4G	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128
A2P2GA1C4S	0.38164	0.38300	0.39065	0.39015	0.47877	0.66229	0.57981	0.47881
A2P2SA1C4G	0.38164	0.38300	0.39065	0.39015	0.53128	0.53128	0.53128	0.53128
A2P2SA1C4S	0.38164	0.38300	0.39065	0.39015	0.53480	0.41200	0.42653	0.48634
A1P4GA1C4G	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128
A1P4GA1C4S	0.38982	0.37045	0.37171	0.38850	0.41494	0.40795	0.40933	0.38078
A1P4SA1C4G	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128	0.53128
A1P4SA1C4S	0.38164	0.38300	0.39065	0.39015	0.52126	0.50185	0.42664	0.48841

Tabla 10.14: Resumen de la prueba de normalidad.

Configuración	Cumplen normalidad	Homocedasticidad	Kruskal-wallis	ANOVA
A2P1SA2C1S	7	0.000 104 56	1.8995×10^{-9}	0
A2P2SA2C2S	8	9.6967×10^{-13}	1.8195×10^{-10}	0
A2P1G2SA2C1G2S	8	8.8052×10^{-13}	1.3646×10^{-10}	0
A2P4SA2C4S	8	8.8152×10^{-13}	1.3026×10^{-10}	0
A2P2GA2C2S	8	1.0659×10^{-12}	1.6451×10^{-10}	0
A2P2SA2C2G	8	1.0083×10^{-12}	1.6606×10^{-10}	0
A1P4GA2C2G	8	8.9595×10^{-13}	1.0354×10^{-10}	0
A1P4GA2C2S	8	1.0943×10^{-12}	1.617×10^{-10}	0
A1P4SA2C2G	8	8.9773×10^{-13}	1.5795×10^{-10}	0
A1P4SA2C2S	8	9.4746×10^{-13}	1.683×10^{-10}	0
A2P2GA1C4G	8	1	1	1
A2P2GA1C4S	8	8.0343×10^{-11}	2.6317×10^{-12}	0
A2P2SA1C4G	8	3.6497×10^{-6}	1.8402×10^{-10}	0
A2P2SA1C4S	8	1.4519×10^{-12}	1.0214×10^{-10}	0
A1P4GA1C4G	8	1	1	1
A1P4GA1C4S	8	9.2704×10^{-13}	1.4075×10^{-10}	0
A1P4SA1C4G	8	1	1	1
A1P4SA1C4S	8	0	1.2499×10^{-11}	0

Tabla 10.15: Resumen de las pruebas estadísticas.

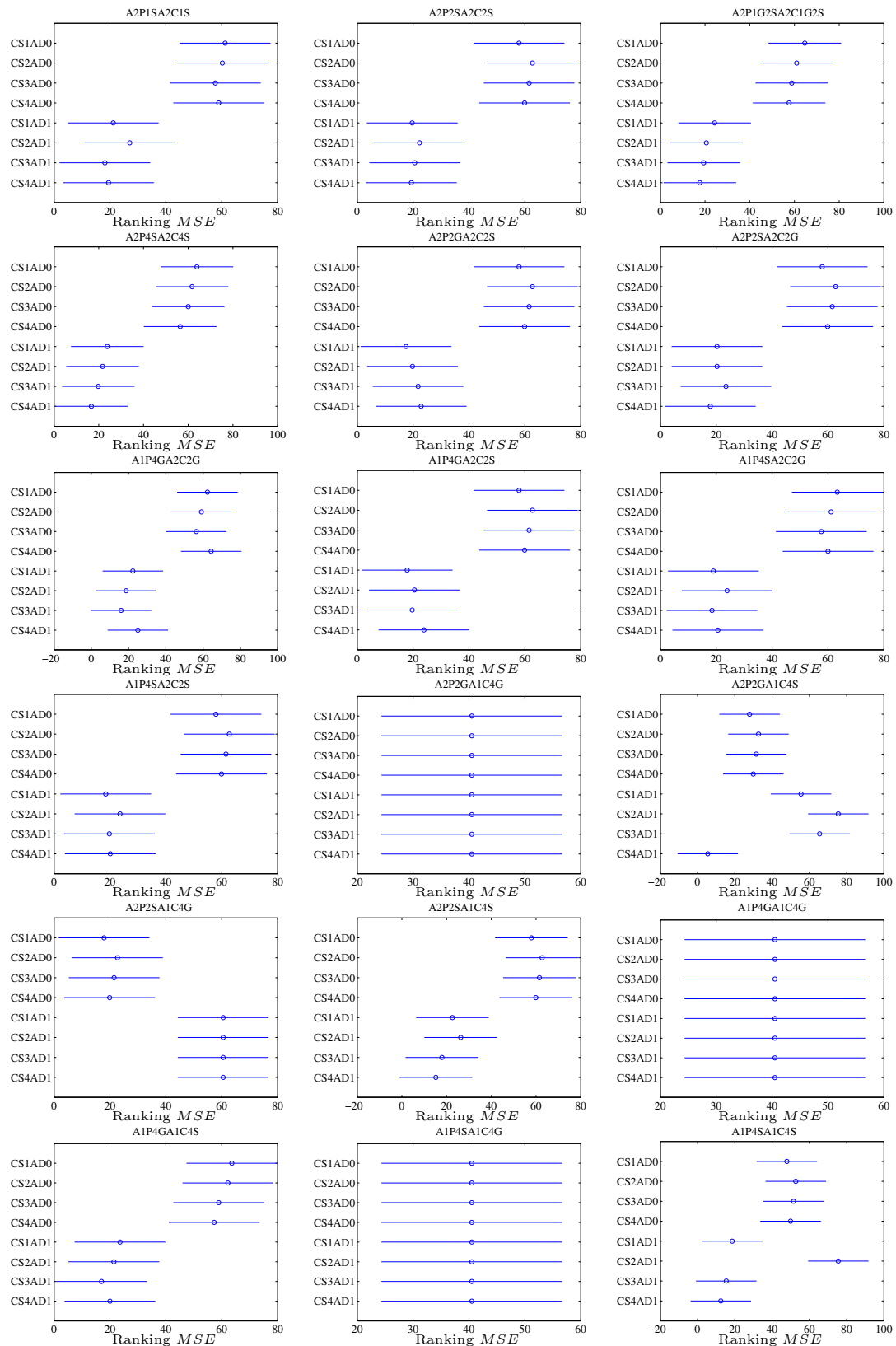


Figura 10.19: Resultado de las comparaciones múltiples.

Capítulo 11

Control de sistemas hidráulicos SISO

11.1. Introducción

Este capítulo presenta el desarrollo de dos aplicaciones de control de sistemas hidráulicos SISO (*Single Input Single Output*). En el primer caso se utiliza la arquitectura I para la identificación de la planta y la arquitectura II para el controlador, con esta configuración se efectúa la regulación del nivel de un tanque cónico considerando diferentes valores de referencia. En el segundo caso de aplicación se realiza el control de nivel de un sistema de dos tanques esféricos acoplados, considerando la variación de parámetros de estos tanques manteniendo la altura del tanque principal en un valor deseado. En esta aplicación se utiliza la arquitectura II para la identificación de la planta y la implementación del controlador.

11.2. Aplicación del sistema de control adaptativo para regular el llenado de un tanque cónico

En esta aplicación se utiliza la arquitectura I para la identificación de la planta y la arquitectura II para el controlador, con esta configuración se regula el nivel de un tanque cónico para diferentes valores. En este caso, también se utiliza un modelo de referencia (comportamiento deseado) para efectuar el entrenamiento del controlador.

11.2.1. Esquema de tiempo discreto utilizado para la identificación de la planta

Se debe tener presente que una aproximación para el modelo del sistema consiste en estimar una estructura neuronal que pueda desempeñar la misma función [6]. Para la identificación con el sistema neuro-difuso se toman tres muestras de la entrada y dos de la salida, teniendo el esquema mostrado en la figura 11.1, donde la salida se puede ver como una función no lineal de las muestras tomadas, siendo z^{-1} un elemento de memoria.

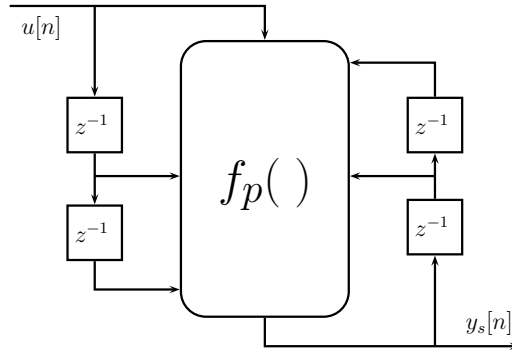


Figura 11.1: Sistema neuronal para la planta.

Desde este punto de vista, se tiene un conjunto de datos de salida $y_s[n]$ dada por la ecuación (11.1), donde \mathbf{H}_p es el vector de parámetros del sistema neuro-difuso.

$$y_s[n] = f_p(u[n], u[n-1], u[n-2], y[n-1], y[n-2], \mathbf{H}_p) \quad (11.1)$$

11.2.2. Esquema de tiempo discreto utilizado para el controlador

Para implementar el controlador se utiliza un esquema con dos elementos de retardo para la entrada y dos elementos de retardo para la salida, tal como se puede apreciar en la figura 11.2. En este esquema $e[n]$ es la señal de error que ingresa al controlador y $u[n]$ la salida del controlador correspondiente a la acción de control.

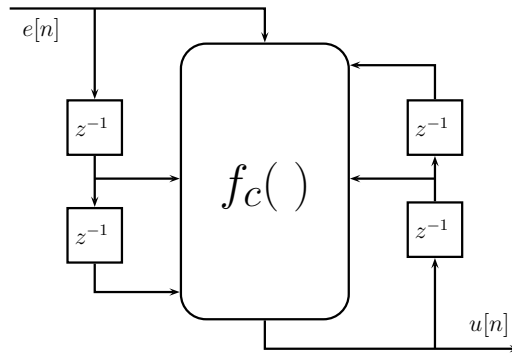


Figura 11.2: Esquema de entradas y salida utilizado para el controlador.

Bajo este enfoque, la acción de control está dada por la ecuación (11.2), donde, \mathbf{H}_c corresponde al vector de parámetros del sistema neuro-difuso.

$$u[n] = f_c(u[n-1], u[n-2], e[n], e[n-1], e[n-2], \mathbf{H}_c) \quad (11.2)$$

11.3. Arquitectura para la identificación de la planta

Con la finalidad de realizar la identificación de la planta se emplea la arquitectura I (sistema difuso compacto) utilizando en este conjuntos difusos gaussianos. Un ejemplo de

estos conjuntos se puede apreciar en la figura 11.3. Esta arquitectura es muy similar a la empleada convencionalmente en redes neuronales de base radial [226].

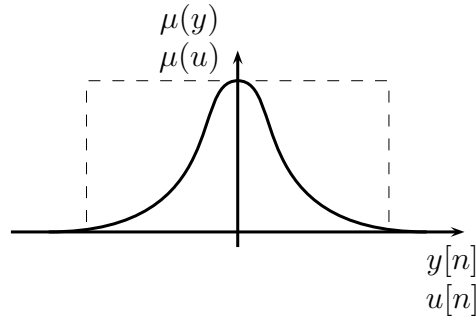


Figura 11.3: Conjunto gaussiano.

La ecuación 11.3 muestra la expresión de entrada salida, empleando como t-norma el producto y utilizando conjuntos difusos gaussianos. Las ecuaciones asociadas al proceso de inferencia son:

$$f(w) = \sum_{l=1}^M y_l \left[\prod_{i=1}^N \exp \left(- \left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right) \right] \quad (11.3)$$

La respectiva función de activación es:

$$Y_l = \prod_{i=1}^N \exp \left(- \left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right) \quad (11.4)$$

De esta forma, la salida del sistema se calcula como:

$$f = \sum_{l=1}^M y_l Y_l \quad (11.5)$$

Donde M es el número de reglas difusas, N el número de entradas al sistema, y_l es el respectivo actuador virtual, δ_{il} y ρ_{il} son el centro y la desviación estándar del conjunto gaussiano. Adicionalmente, $w_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2]\}$ es el dato de entrada al sistema, de manera general, todas las variables de entrada se representan como w .

La representación de la red neuro-difusa asociada se puede apreciar en la figura 11.4. La primera capa corresponde a la productoria de las funciones gaussianas, es decir, corresponde al cálculo de Y_l . En la segunda se realiza el producto de los consecuentes multiplicados por Y_l . Finalmente, en la tercera capa se determina la salida de la inferencia $f(w)$.

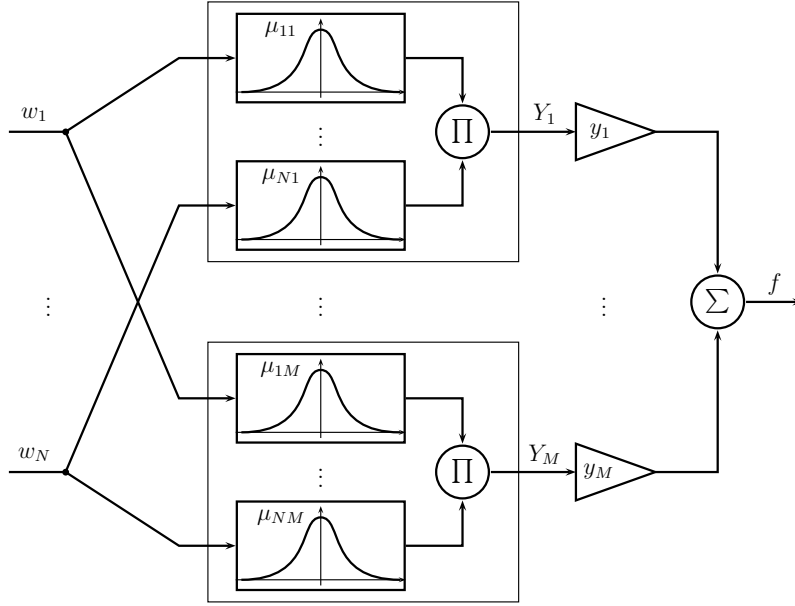


Figura 11.4: Representación de la red neuro-difusa.

11.4. Ecuaciones de entrenamiento para la identificación de la planta

Teniendo en cuenta el algoritmo *Back Propagation* para un sistema neuro-difuso y usando el algoritmo de gradiente descendiente para determinar los parámetros del sistema, se busca minimizar el error correspondiente a la ecuación (11.6).

$$J_p = \frac{1}{2} [f(w[n]) - y[n]]^2 \quad (11.6)$$

Donde $w[n]$ son los datos de entrada al sistema y $y[n]$ los datos de salida deseados para un tiempo n . Estos datos forman las respectivas parejas de entrada salida de entrenamiento.

Para realizar el entrenamiento de los parámetros se calculan las derivadas del error con $f(w)$ teniendo:

$$y_l(k+1) = y_l(k) - \alpha \left. \frac{\partial J_p}{\partial y_l} \right|_n \quad (11.7)$$

$$\delta_{il}(k+1) = \delta_{il}(k) - \alpha \left. \frac{\partial J_p}{\partial \delta_{il}} \right|_n \quad (11.8)$$

$$\rho_{il}(k+1) = \rho_{il}(k) - \alpha \left. \frac{\partial J_p}{\partial \sigma_{il}} \right|_n \quad (11.9)$$

De esta forma, para actualizar los parámetros se utilizan las ecuaciones:

$$y_l(k+1) = y_l(k) - \alpha (f - y) Y_l \quad (11.10)$$

$$\delta_{il}(k+1) = \delta_{il}(k) - \alpha (f - y) y_l(k) Y_l \frac{2(w_i[n] - \delta_{il}(k))}{[\rho_{il}(k)]^2} \quad (11.11)$$

$$\rho_{il}(k+1) = \rho_{il}(k) - \alpha (f - y) y_l(k) Y_l \frac{2(w_i[n] - \delta_{il}(k))^2}{[\rho_{il}(k)]^3} \quad (11.12)$$

El algoritmo para la adaptación (entrenamiento) de parámetros del sistema neuro-difuso comprende los siguientes pasos:

1. Determinar el sistema neuro-difuso estableciendo M y N . Cuanto mayor sea M , resultarán más parámetros permitiendo mayor adaptabilidad pero con mayor requerimiento de cálculo. También se deben especificar los parámetros $y_l(0)$, $\delta_{il}(0)$ y $\rho_{il}(0)$. Estos parámetros iniciales pueden ser determinados de acuerdo a las reglas lingüísticas del experto, o escogidas de tal manera que las funciones de pertenencia cubran uniformemente el espacio de entrada.
2. Calcular la salida del sistema neuro-difuso para un par de entrada-salida $(w[n], y[n])$, $n = 1, 2, \dots, N_T$, en la k -ésima etapa de entrenamiento, $k = 0, 1, 2, \dots, K_T$.
3. Actualizar los parámetros $y_l(k+1)$, $\delta_{il}(k+1)$ y $\rho_{il}(k+1)$. Donde α corresponde al ritmo de aprendizaje. Si α es escogido con un valor muy grande puede causar que el algoritmo no converja, mientras escoger un valor muy pequeño puede ocasionar que el algoritmo requiera más tiempo para converger.
4. Volver al paso 2 con $k = k+1$, hasta que el error $J_p(k)$ sea menor que un ε definido, o hasta que k sea igual a un número determinado.
5. Volver al paso 2 con $n = n+1$ para actualizar los parámetros usando la siguiente pareja de entrada-salida $(w[n+1], y[n+1])$.

El algoritmo anterior es definido para una época en la cual cada elemento del conjunto de entrenamiento es usado solo una vez, y los parámetros son actualizados usando una función de error J_p que depende solamente de una pareja de entrenamiento en el tiempo.

11.5. Arquitectura del controlador

La implementación del controlador se realiza con un sistema difuso compacto con arquitectura II utilizando los conjuntos difusos mostrados en la figura 11.5. Particularmente en la figura 11.5(a), se tiene un conjunto difuso sigmoideal para modelar los valores positivos del universo de discurso, mientras que en la figura 11.5(b), se representan los valores negativos para el error $e[n]$ y la acción de control $u[n]$.

Considerando los conjuntos difusos de la figura 11.5 y la estructura general del controlador dada por la ecuación (11.13) se tiene el esquema de la figura 11.6, donde se muestra el controlador difuso propuesto.

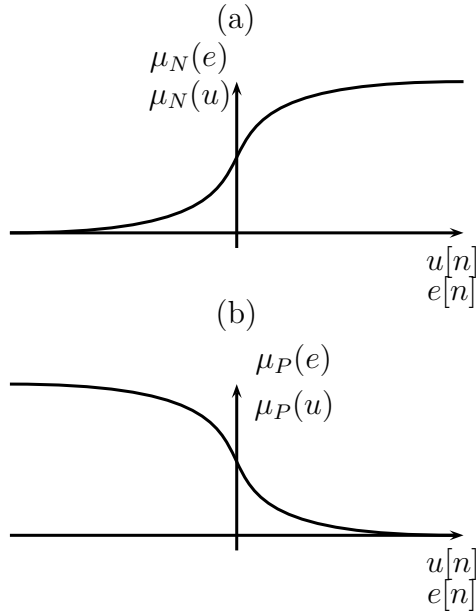


Figura 11.5: Funciones de pertenencia empleadas.

La salida del controlador se puede calcular como:

$$u[n] = \sum_{i=1}^5 \sum_{j=1}^2 v_{ij} \mu_{ij}(x_i) \quad (11.13)$$

Donde $x_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2], \}$. Para cada entrada x_i se puede definir una función f_i de la forma:

$$f_i = \sum_{j=1}^2 v_{ij} \mu_{ij}(x_i) \quad (11.14)$$

es decir:

$$f_i = v_{i1} \mu_{i1}(x_i) + v_{i2} \mu_{i2}(x_i) \quad (11.15)$$

Por su parte, la función de pertenencia $\mu_{ij}(x_i)$ corresponde a:

$$\mu_{ij}(x_i) = \frac{1}{1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})}} \quad (11.16)$$

Para llevar a cabo el cálculo de las derivadas la ecuación (11.16) se puede representar como:

$$\mu_{ij}(x_i) = (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} \quad (11.17)$$

El conjunto de parámetros del controlador corresponde a $\mathbf{H}_c = \{v_{ij}, \sigma_{ij}, \gamma_{ij}\}$ y un posible parámetro de este es $h_c = h_{ij} \in \mathbf{H}_c$.

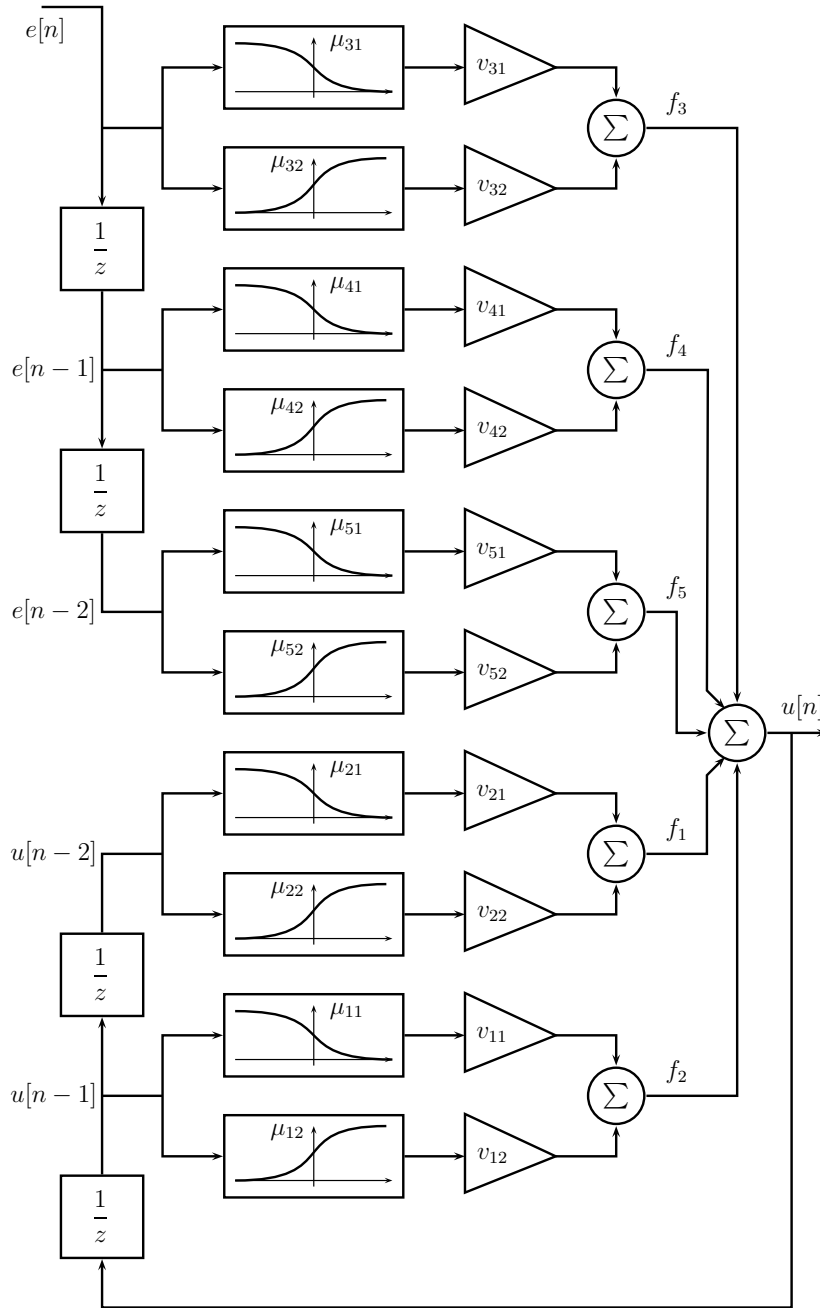


Figura 11.6: Esquema del sistema de control neuro-difuso.

11.6. Ecuaciones para el entrenamiento de parámetros del controlador

Teniendo presente que y_r es la respuesta del modelo de referencia (comportamiento deseado), en primer lugar se tiene que la función de desempeño es:

$$J_p = \frac{1}{2} (y_r[n] - y[n])^2 = \frac{1}{2} (e[n])^2 \quad (11.18)$$

En segundo lugar se debe tener presente que la ecuación de la planta es:

$$y[n] = f_p(y[n-1], y[n-2], u[n], u[n-1], u[n-2]) \quad (11.19)$$

Por su parte la ecuación del error es:

$$e[n] = y_r[n] - y[n] \quad (11.20)$$

Considerando la arquitectura del controlador (figura 11.6) y las expresiones para f_i (ecuación (11.14)) la dinámica del controlador es dada por:

$$u[n] = f_1(u[n-1]) + f_2(u[n-2]) + f_3(e[n]) + f_4(e[n-1]) + f_5(e[n-2]) \quad (11.21)$$

La derivada de la salida de la planta con respecto a los parámetros del controlador es:

$$\frac{dy[n]}{dh_c} = \frac{d}{dh_c} f_p(y[n-1], y[n-2], u[n], u[n-1], u[n-2]) \quad (11.22)$$

La respectiva derivada del error $e[n] = y_r[n] - y[n]$ para un parámetro del controlador h_c es:

$$\frac{de[n]}{dh_c} = -\frac{dy[n]}{dh_c} \quad (11.23)$$

De la misma manera, la derivada de la acción de control $u[n]$ con respecto a h_{ij} es:

$$\frac{du[n]}{dh_{ij}} = \frac{df_1(e[n])}{dh_{ij}} + \frac{df_2(e[n-1])}{dh_{ij}} + \frac{df_3(e[n-2])}{dh_{ij}} + \frac{df_4(u[n-1])}{dh_{ij}} + \frac{df_5(u[n-2])}{dh_{ij}} \quad (11.24)$$

Como es de apreciar, se deben calcular las respectivas derivadas de los parámetros respecto a la planta y el controlador.

11.6.1. Derivadas de los parámetros del controlador con respecto a la planta

Para tener una expresión general de la dinámica de la planta f_p se tiene $w_i \in \{y[n-1], y[n-2], u[n], u[n-1], u[n-2]\}$. Así, la ecuación para calcular la salida de la planta es:

$$y = \sum_{l=1}^M y_l \left[\prod_{i=1}^N \exp \left(- \left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right) \right] \quad (11.25)$$

con

$$\mu_{il}(w_i) = \exp \left(- \left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right) \quad (11.26)$$

entonces

$$y = \sum_{l=1}^M y_l \left[\prod_{i=1}^N \mu_{il}(w_i) \right] \quad (11.27)$$

$$y = y_1 \prod_{i=1}^N \mu_{i1}(w_i) + y_2 \prod_{i=1}^N \mu_{i2}(w_i) + y_3 \prod_{i=1}^N \mu_{i3}(w_i) + \dots \quad (11.28)$$

Utilizando el índice auxiliar j para considerar el caso donde $j \neq i$, entonces, la derivada de los productos es:

$$\frac{d}{dh_c} \left(\prod_{i=1}^N \mu_{il}(w_i) \right) = \sum_{i=1}^N \left(\left(\prod_{\substack{j=1 \\ j \neq i}}^N \mu_{jl}(w_j) \right) \frac{d\mu_{il}(w_i)}{dw_i} \frac{dw_i}{dh_c} \right) \quad (11.29)$$

Tomando:

$$P_{il} = \frac{d\mu_{il}(w_i)}{dw_i} \prod_{\substack{j=1 \\ j \neq i}}^N \mu_{jl}(w_j) \quad (11.30)$$

con:

$$\frac{d\mu_{il}(w_i)}{dw_i} = \frac{d}{dw_i} \left(e^{-\left(\frac{w_i - \delta_{il}}{\rho_{il}}\right)^2} \right) = -2e^{-\left(\frac{w_i - \delta_{il}}{\rho_{il}}\right)^2} \frac{w_i - \delta_{il}}{\rho_{il}^2} \quad (11.31)$$

De esta forma se tiene:

$$\frac{dy}{dh_c} = y_1 \sum_{i=1}^N P_{i1} \frac{dw_i}{dh_c} + y_2 \sum_{i=1}^N P_{i2} \frac{dw_i}{dh_c} + y_3 \sum_{i=1}^N P_{i3} \frac{dw_i}{dh_c} + \dots \quad (11.32)$$

$$\frac{dy}{dh_c} = \sum_{i=1}^N (y_1 P_{i1} + y_2 P_{i2} + y_3 P_{i3} + \dots) \frac{dw_i}{dh_c} \quad (11.33)$$

$$\frac{dy}{dh_c} = \sum_{i=1}^N \left(\sum_{l=1}^M y_l P_{il} \right) \frac{dw_i}{dh_c} \quad (11.34)$$

Es decir:

$$\frac{dy}{dh_c} = \left(\sum_{l=1}^M y_l P_{1l} \right) \frac{dw_1}{dh_c} + \left(\sum_{l=1}^M y_l P_{2l} \right) \frac{dw_2}{dh_c} + \left(\sum_{l=1}^M y_l P_{3l} \right) \frac{dw_3}{dh_c} + \dots \quad (11.35)$$

Con:

$$C_i = \sum_{l=1}^M y_l P_{il} \quad (11.36)$$

Entonces, finalmente se establece que:

$$\frac{dy[n]}{dh_{ij}} = C_1 \frac{dw_1}{dh_{ij}} + C_2 \frac{dw_2}{dh_{ij}} + C_3 \frac{dw_3}{dh_{ij}} + C_4 \frac{dw_4}{dh_{ij}} + C_5 \frac{dw_5}{dh_{ij}} \quad (11.37)$$

11.6.2. Derivadas de los parámetros del controlador con respecto al controlador

En este procedimiento se utiliza el índice auxiliar l con el fin de considerar el caso donde $l \neq i$, de esta manera, para determinar las respectivas derivadas de los parámetros del controlador considerando las ecuaciones del controlador se tiene que:

$$\frac{df_l(x_i)}{dh_{ij}} = \frac{d}{dh_{ij}} (v_{l1} \mu_{l1}(x_i)) + \frac{d}{dh_{ij}} (v_{l2} \mu_{l2}(x_i)) \quad (11.38)$$

Donde $l = 1, \dots, 5$, $i = 1, \dots, 5$ y $j = 1, 2$, por lo cual se tienen diferentes casos para valores de i y l . Para el caso cuando $l \neq i$ se tiene:

$$\begin{aligned}\frac{df_l(x_l)}{dh_{ij}} &= \frac{d}{dx_l}(v_{l1}\mu_{l1}(x_l))\frac{dx_l}{dh_{ij}} + \frac{d}{dx_l}(v_{l2}\mu_{l2}(x_l))\frac{dx_l}{dh_{ij}} \\ \frac{df_l(x_l)}{dh_{ij}} &= \left[\frac{d}{dx_l}(v_{l1}\mu_{l1}(x_l)) + \frac{d}{dx_l}(v_{l2}\mu_{l2}(x_l)) \right] \frac{dx_l}{dh_{ij}}\end{aligned}\quad (11.39)$$

Para $j = 1, 2$ las respectivas derivadas son:

$$\begin{aligned}\frac{d}{dx_l}(v_{lj}\mu_{lj}(x_l)) &= \frac{d}{dx_l} \left(v_{lj} (1 + e^{-\sigma_{lj}(x_l - \gamma_{lj})})^{-1} \right) \\ \frac{d}{dx_l}(v_{lj}\mu_{lj}(x_l)) &= v_{lj} (1 + e^{-\sigma_{lj}(x_l - \gamma_{lj})})^{-2} e^{-\sigma_{lj}(x_l - \gamma_{lj})} \sigma_{lj}\end{aligned}\quad (11.40)$$

Por su parte, cuando $l = i$ y $j = 1$ se tiene:

$$\frac{df_i(x_i)}{dh_{i1}} = \frac{d}{dh_{i1}}(v_{i1}\mu_{i1}(x_i)) + \frac{d}{dx_i}(v_{i2}\mu_{i2}(x_i))\frac{dx_i}{dh_{i1}}\quad (11.41)$$

También cuando $l = i$ y $j = 2$ se tiene:

$$\frac{df_i(x_i)}{dh_{i2}} = \frac{d}{dx_i}(v_{i1}\mu_{i1}(x_i))\frac{dx_i}{dh_{i2}} + \frac{d}{dh_{i2}}(v_{i2}\mu_{i2}(x_i))\quad (11.42)$$

Para desarrollar las anteriores ecuaciones, en primer lugar:

$$\frac{d}{dx_i}(v_{ij}\mu_{ij}(x_i)) = v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij}\quad (11.43)$$

En segundo lugar, para las otras derivadas:

$$\frac{d}{dh_{ij}}(v_{ij}\mu_{ij}(x_i)) = \frac{d}{dh_{ij}} \left(v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} \right)\quad (11.44)$$

Para el parámetro $h_{ij} = v_{ij}$ se tiene:

$$\begin{aligned}\frac{d}{dv_{ij}}(v_{ij}\mu_{ij}(x_i)) &= (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} - \\ &v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} (-\sigma_{ij}) \frac{dx_i}{dv_{ij}}\end{aligned}\quad (11.45)$$

Tomando el parámetro $h_{ij} = \sigma_{ij}$ se establece que:

$$\frac{d}{d\sigma_{ij}}(v_{ij}\mu_{ij}(x_i)) = -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \left(\gamma_{ij} - x_i - \sigma_{ij} \frac{dx_i}{d\sigma_{ij}} \right)\quad (11.46)$$

Con el parámetro $h_{ij} = \gamma_{ij}$ se obtiene:

$$\frac{d}{d\gamma_{ij}}(v_{ij}\mu_{ij}(x_i)) = -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \left(\sigma_{ij} - \sigma_{ij} \frac{dx_i}{d\gamma_{ij}} \right)\quad (11.47)$$

De forma general, estas ecuaciones se pueden escribir como:

$$\begin{aligned}
\frac{d}{dv_{ij}}(v_{ij}\mu_{ij}(x_i)) &= F_{v_{ij}} + K_{v_{ij}} \frac{dx_i}{dv_{ij}} \\
\frac{d}{d\sigma_{ij}}(v_{ij}\mu_{ij}(x_i)) &= F_{\sigma_{ij}} + K_{\sigma_{ij}} \frac{dx_i}{d\sigma_{ij}} \\
\frac{d}{d\gamma_{ij}}(v_{ij}\mu_{ij}(x_i)) &= F_{\gamma_{ij}} + K_{\gamma_{ij}} \frac{dx_i}{d\gamma_{ij}}
\end{aligned} \tag{11.48}$$

Donde:

$$\begin{aligned}
F_{v_{ij}} &= (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} \\
F_{\sigma_{ij}} &= -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} (\gamma_{ij} - x_i) \\
F_{\gamma_{ij}} &= -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \\
K_{v_{ij}} &= v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \\
K_{\sigma_{ij}} &= v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \\
K_{\gamma_{ij}} &= v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij}
\end{aligned} \tag{11.49}$$

En términos generales, considerando h_{ij} la derivada se puede representar de la forma:

$$\frac{d}{dh_{ij}}(v_{ij}\mu_{ij}(x_i)) = F_{h_{ij}} + K_{h_{ij}} \frac{dx_i}{dh_{ij}} \tag{11.50}$$

Retomando el caso cuando $l = i$ y $j = 1$ de la ecuación (11.41) se tiene:

$$\begin{aligned}
\frac{df_i(x_i)}{dh_{i1}} &= \frac{d}{dh_{i1}}(v_{i1}\mu_{i1}(x_i)) + \frac{d}{dx_i}(v_{i2}\mu_{i2}(x_i)) \frac{dx_i}{dh_{i1}} \\
&= F_{h_{i1}} + K_{h_{i1}} \frac{dx_i}{dh_{i1}} + \frac{d}{dx_i}(v_{i2}\mu_{i2}(x_i)) \frac{dx_i}{dh_{i1}} \\
&= F_{h_{i1}} + \left(K_{h_{i1}} + \frac{d}{dx_i}(v_{i2}\mu_{i2}(x_i)) \right) \frac{dx_i}{dh_{i1}}
\end{aligned} \tag{11.51}$$

También cuando $l = i$ y $j = 2$ de la ecuación (11.42) se establece:

$$\begin{aligned}
\frac{df_i(x_i)}{dh_{i2}} &= \frac{d}{dx_i}(v_{i1}\mu_{i1}(x_i)) \frac{dx_i}{dh_{i2}} + \frac{d}{dh_{i2}}(v_{i2}\mu_{i2}(x_i)) \\
&= \frac{d}{dx_i}(v_{i1}\mu_{i1}(x_i)) \frac{dx_i}{dh_{i2}} + F_{h_{i2}} + K_{h_{i2}} \frac{dx_i}{dh_{i2}} \\
&= F_{h_{i2}} + \left(\frac{d}{dx_i}(v_{i1}\mu_{i1}(x_i)) + K_{h_{i2}} \right) \frac{dx_i}{dh_{i2}}
\end{aligned} \tag{11.52}$$

11.6.3. Proceso para el entrenamiento del controlador

Tomando $w_i \in \{y[n-1], y[n-2], u[n], u[n-1], u[n-2]\}$ y $x_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2]\}$, entonces en términos generales las ecuaciones que implementan la dinámica de los parámetros de entrenamiento son:

$$\frac{de[n]}{dh_{ij}} = -\frac{dy}{dh_{ij}} \tag{11.53}$$

$$\frac{dy[n]}{dh_{ij}} = C_1 \frac{dw_1}{dh_{ij}} + C_2 \frac{dw_2}{dh_{ij}} + C_3 \frac{dw_3}{dh_{ij}} + C_4 \frac{dw_4}{dh_{ij}} + C_5 \frac{dw_5}{dh_{ij}} \quad (11.54)$$

$$\frac{du[n]}{dh_{ij}} = \frac{df_1(x_1)}{dh_{ij}} + \frac{df_2(x_2)}{dh_{ij}} + \frac{df_3(x_3)}{dh_{ij}} + \frac{df_4(x_4)}{dh_{ij}} + \frac{df_5(x_5)}{dh_{ij}} \quad (11.55)$$

En esta última ecuación si $l \neq i$ entonces:

$$\frac{df_l(x_l)}{dh_{ij}} = \left[\frac{d}{dx_l}(v_{l1}\mu_{l1}(x_l)) + \frac{d}{dx_l}(v_{l2}\mu_{l2}(x_l)) \right] \frac{dx_l}{dh_{ij}} \quad (11.56)$$

Por otra parte, si $l = i$ se tiene:

$$\frac{df_i(x_i)}{dh_{i1}} = F_{h_{i1}} + \left(K_{h_{i1}} + \frac{d}{dx_i}(v_{i2}\mu_{i2}(x_i)) \right) \frac{dx_i}{dh_{i1}} \quad (11.57)$$

$$\frac{df_i(x_i)}{dh_{i2}} = F_{h_{i2}} + \left(\frac{d}{dx_i}(v_{i1}\mu_{i1}(x_i)) + K_{h_{i2}} \right) \frac{dx_i}{dh_{i2}} \quad (11.58)$$

Finalmente la actualización de los parámetros se realiza de la forma:

$$h_{ij}(k+1) = h_{ij}(k) - \alpha \frac{de[n]}{dh_{ij}} \quad (11.59)$$

Donde α corresponde a la tasa de aprendizaje. Considerando cada parámetro se tiene:

$$v_{ij}(k+1) = v_{ij}(k) - \alpha \frac{de[n]}{dv_{ij}} \quad (11.60)$$

$$\sigma_{ij}(k+1) = \sigma_{ij}(k) - \alpha \frac{de[n]}{d\sigma_{ij}} \quad (11.61)$$

$$\gamma_{ij}(k+1) = \gamma_{ij}(k) - \alpha \frac{de[n]}{d\gamma_{ij}} \quad (11.62)$$

Los pasos del algoritmo utilizado para el entrenamiento del controlador neuro-difuso son los siguientes:

1. Establecer el modelo de la planta y escoger la configuración inicial de parámetros del controlador.
2. Calcular la respuesta del modelo de referencia obteniendo y_r .
3. En la k -ésima etapa de entrenamiento, $k = 0, 1, 2, \dots, K_T$, para el valor actual de n de simulación se calcula la salida del sistema de control utilizando el modelo neuro-difuso de la planta.
4. Realizar el ajuste de los parámetros del controlador neuro-difuso utilizando las respectivas ecuaciones que involucran la dinámica del sistema de control y las derivadas de los parámetros (ecuaciones (11.59), (11.53) y subsecuentes). Es importante señalar que los parámetros que se ajustan se almacenan en variables auxiliares ya que durante este paso el controlador no utiliza estos valores.

5. Regresar al paso 3 para el siguiente paso de simulación $n = n + 1$ (donde se calcula la salida del sistema de control) hasta que n sea igual al valor definido N_T (tiempo de simulación).
6. En el caso de completar el tiempo de simulación, se actualizan los nuevos parámetros optimizados del controlador y se regresa al paso 3 de evaluación del sistema de control para una nueva iteración $k = k + 1$, hasta que $J_c(k) = \sum_{n=1}^{N_T} J_c(k, n)$ sea menor que un ε definido, o hasta que k sea igual a un número definido K_T .

11.7. Modelo del tanque cónico

El sistema de tanque cónico es un proceso SISO (Single Input Single Output). La salida de este proceso es el nivel $h(t)$ y la entrada al proceso es el flujo de líquido F_i . El esquema del tanque cónico considerado se puede apreciar en la figura 11.7.

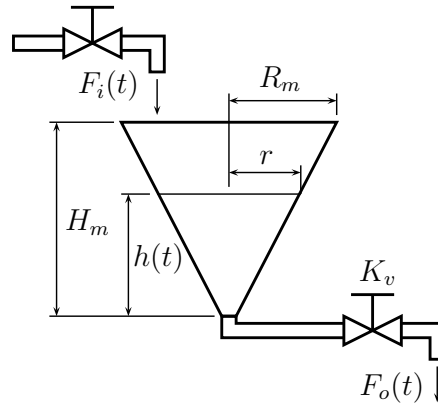


Figura 11.7: Diagrama del tanque cónico.

En este sistema un líquido de densidad constante ingresa a una tasa volumétrica F_i en un tanque cónico de altura H_m y radio máximo R_m . La salida del tanque es $F_o = K_v \sqrt{h}$, donde h es la altura del líquido en el tanque y K_v es el coeficiente de la válvula. El proceso tiene una alta no linealidad debido a los cambios en la ganancia del proceso y la constante de tiempo con respecto a la altura del líquido en el tanque [264].

Considerando la entrada F_i y la salida F_o , según la ley de conservación de masa, la acumulación es la masa que ingresa menos la que sale. Dado que se manifiesta como un aumento o disminución de volumen, la acumulación es el cambio de volumen con respecto al tiempo.

$$\frac{dV}{dt} = F_i - F_o \quad (11.63)$$

El cono circular tiene volumen:

$$V = \frac{\pi}{3} r^2 h \quad (11.64)$$

De la geometría del tanque se observa que:

$$r = \frac{R_m}{H_m} h \quad (11.65)$$

Entonces:

$$V = \frac{\pi}{3} \left(\frac{R_m}{H_m} \right)^2 h^3 \quad (11.66)$$

Derivado con respecto al tiempo:

$$\frac{dV}{dt} = \pi \left(\frac{R_m}{H_m} \right)^2 h^2 \frac{dh}{dt} \quad (11.67)$$

De esta forma:

$$\frac{dh}{dt} = \frac{F_i - K_v h}{\pi \left(\frac{R_m}{H_m} \right)^2 h^2} \quad (11.68)$$

El diagrama de bloques de este sistema se muestra en la figura 11.8, donde se pueden apreciar las no linealidades presentes en el sistema hidráulico.

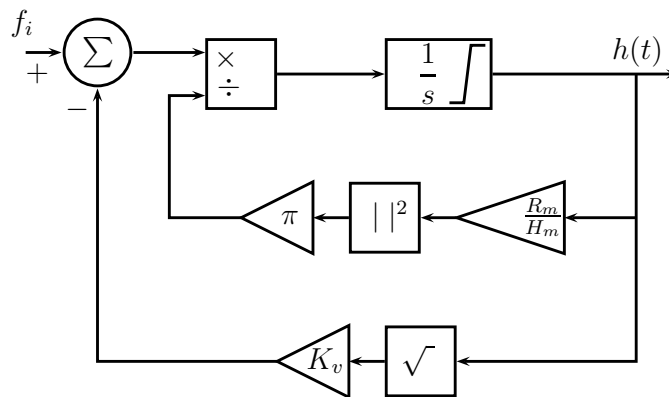


Figura 11.8: Diagrama de bloques del tanque cónico.

Los parámetros del tanque se pueden observar en la tabla 11.1.

Parámetro	Descripción	Valor
R_m	Radio máximo	0,48m
H_m	Altura máxima	1,2m
F_i	Flujo de entrada	0,0072m ³ /s (max)
K_v	Coefficiente de la válvula	0,0067m ² /s
h	Altura	Variable
F_o	Flujo de salida	Variable
r	Radio dependiente de h	Variable

Tabla 11.1: Parámetros del tanque cónico.

11.8. Resultados

Para la configuración inicial del sistema utilizado para identificar la planta se lleva a cabo una localización sistemática de las funciones de pertenencia gaussianas en el rango

$[-1,5 \ 1,5]$ con una varianza de 2 y los actuadores virtuales en cero. Como esta es una asignación sistemática se requiere un entrenamiento previo de la planta antes de utilizarla en el sistema de control adaptativo. Los datos empleados para efectuar el entrenamiento del modelo de la planta se observan en la figura 11.9.

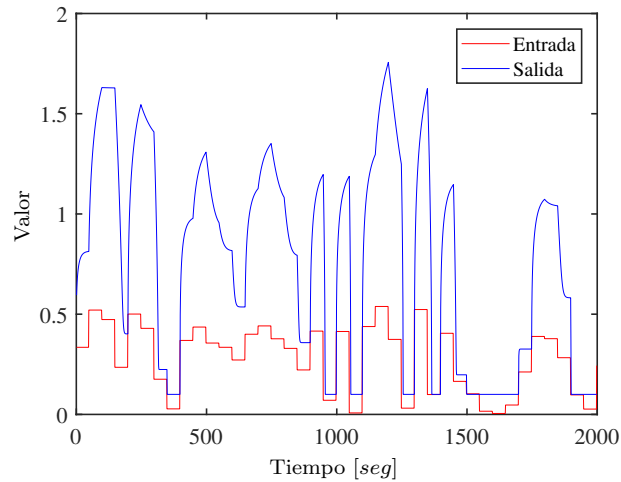


Figura 11.9: Datos de la planta en lazo abierto.

Utilizando 400 épocas de entrenamiento, se efectúa la identificación de la planta obteniendo el resultado mostrado en la figura 11.10 con un MSE (*Mean Squared Error*) de 0,0012.

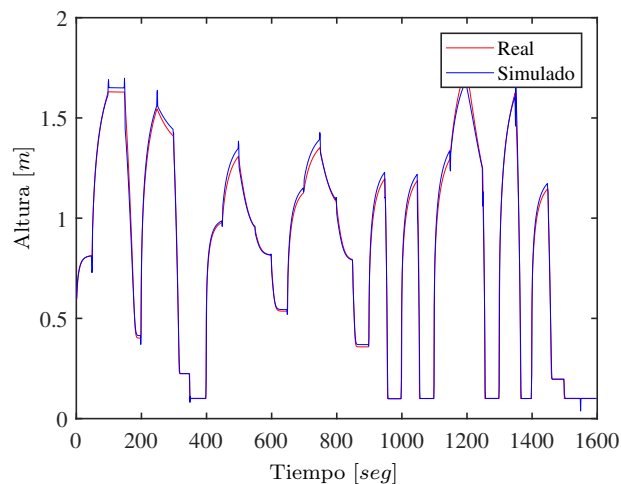


Figura 11.10: Resultado del proceso de identificación en lazo abierto.

Para la configuración inicial del controlador se considera que las funciones f_i presentan un comportamiento lineal en el rango $[-2,0 \ 2,0]$, tal como se muestra en la figura 11.11, adicionalmente, al efectuar la optimización de parámetros, los actuadores virtuales se inician en cero.

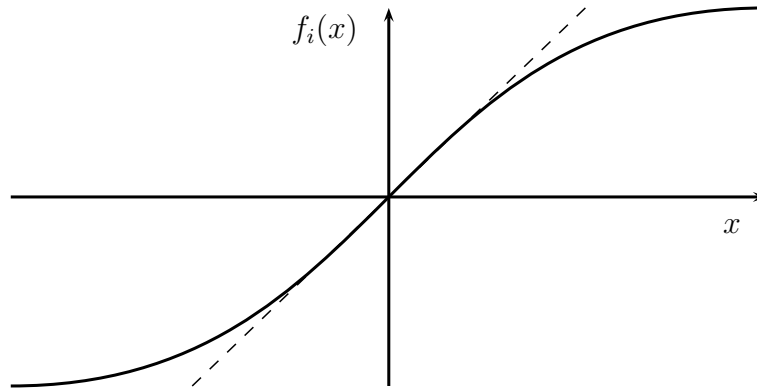


Figura 11.11: Forma de la aproximación para f_i .

Con la finalidad de llevar a cabo el entrenamiento del controlador para el comportamiento deseado (modelo de referencia), se utiliza un sistema lineal dado por la función de transferencia de la ecuación (11.69), observando que este sistema tiene un tiempo de establecimiento de 200 segundos. En la figura 11.12 se puede apreciar la respuesta que tiene el modelo de referencia, es decir, el tipo de comportamiento deseado que se quiere tener al efectuar la optimización del controlador durante cada proceso adaptativo.

$$G(s) = \frac{Y(s)}{R(s)} = \frac{1}{50s + 1} \quad (11.69)$$

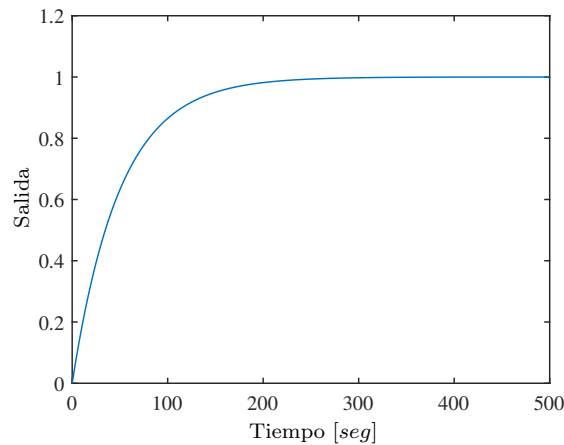


Figura 11.12: Respuesta del modelo de referencia.

Considerando diferentes valores de referencia, se realiza la simulación del sistema de control adaptativo donde de forma iterativa se lleva a cabo la identificación de la planta y el entrenamiento del controlador, de esta forma, se obtiene el resultado mostrado en la figura 11.13. En la parte izquierda, figura 11.13(a) se muestra el nivel del tanque y en la parte derecha, figura 11.13(b) el flujo de entrada al tanque cónico en m^3/s .

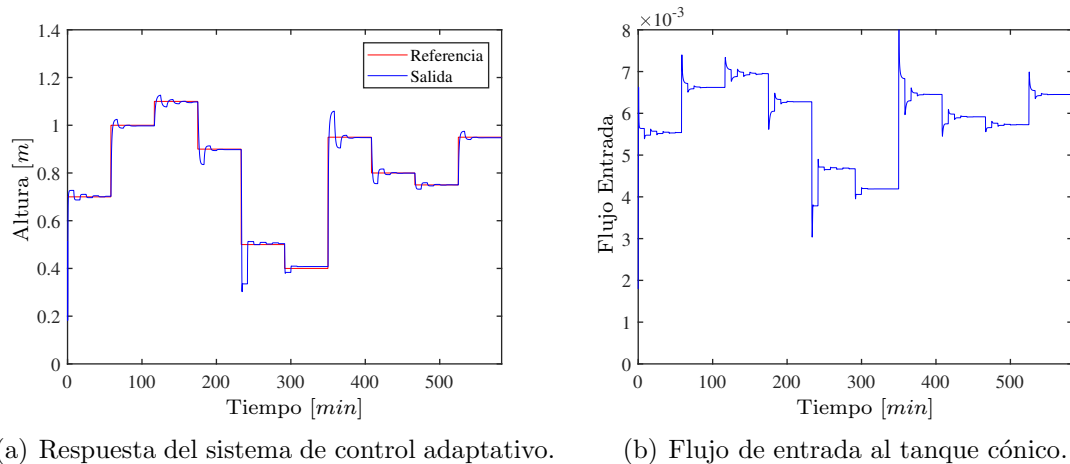


Figura 11.13: Respuesta del sistema de control adaptativo.

Adicionalmente, para observar de forma detallada el funcionamiento del sistema de control adaptativo, en la figura 11.14 se muestran los ajustes progresivos para la identificación de la planta y entrenamiento del controlador de tal forma que el nivel del tanque llegue al valor de referencia.

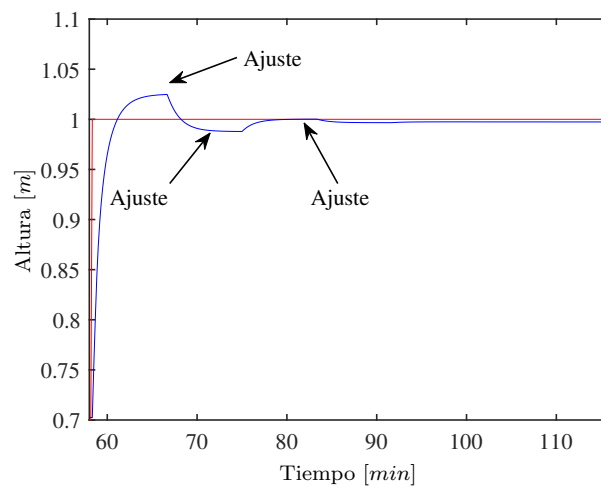


Figura 11.14: Detalle de la respuesta del sistema de control adaptativo.

11.9. Aplicación del sistema de control adaptativo para regular el llenado de tanques esféricos

En este caso, se implementa el control del nivel de un sistema compuesto por dos tanques esféricos acoplados considerando la variación de parámetros de estos tanques manteniendo la altura del tanque principal en un valor deseado. En esta aplicación se utiliza la arquitectura II para la identificación de la planta y la implementación del controlador. Con el fin de efectuar el entrenamiento del controlador, se pondera la acción de control y la señal del error en la función de ajuste.

11.9.1. Arquitectura del sistema neuro-difuso para identificación y control

Tanto para la identificación de la planta como para el control se toman muestras de la entrada y salida, de tal forma que la respuesta del sistema neuro-difuso se puede ver como una función no lineal de estas señales.

En esta aplicación se consideran diferentes configuraciones del sistema neuro-difuso variando el número de retardos de la entrada y salida, de esta manera, la figura 11.15 muestra el esquema empleado presentando además los elementos de memoria (retardos).

Para el caso de identificación $w[u]$ es la entrada de la planta $u[n]$ y $g[n]$ la salida $y[n]$, mientras que para el controlador $w[u]$ es la señal de error $e[n]$ y $g[n]$ la acción de control $u[n]$.

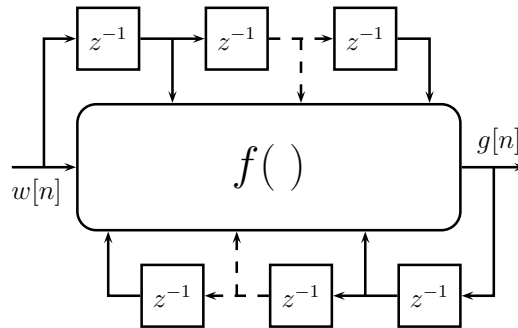


Figura 11.15: Arquitectura general del sistema neuro-difuso para identificación y control.

Con la finalidad de implementar el sistema difuso compacto se consideran los conjuntos mostrados en la figura 11.16, donde se tienen conjuntos difusos sigmoidales para representar los valores positivos del universo de discurso μ_P , como también para modelar valores negativos μ_N .

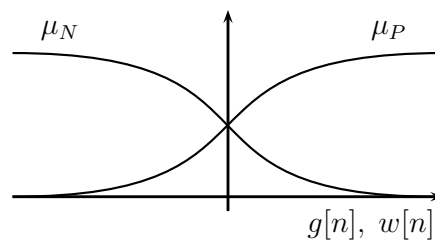


Figura 11.16: Funciones de pertenencia sigmoidales empleadas.

Considerando los conjuntos difusos de la figura 11.16 y la estructura II del sistema compacto, la figura 11.17 muestra el sistema difuso utilizado para identificación y control.

En la figura 11.17 los subíndices g y w hacen referencia a la entrada considerada (entrada directa o realimentación), sin embargo, estas entradas se pueden organizar en un solo arreglo (vector) de tamaño $M + N + 1$ de tal forma que la salida del sistema

neuro-difuso compacto se puede calcular como:

$$g[n] = \sum_{i=1}^{M+N+1} \sum_{j=1}^2 v_{ij} \mu_{ij}(x_i) \quad (11.70)$$

donde $x_i \in \{w[n], w[n-1], w[n-2], \dots, w[n-M], g[n-1], g[n-2], \dots, g[n-N]\}$. Para cada entrada x_i se puede definir una función f_i de la forma:

$$f_i = \sum_{j=1}^2 v_{ij} \mu_{ij}(x_i) = v_{i1} \mu_{i1}(x_i) + v_{i2} \mu_{i2}(x_i) \quad (11.71)$$

Por su parte, la función de pertenencia $\mu_{ij}(x_i)$ es:

$$\mu_{ij}(x_i) = \frac{1}{1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})}} \quad (11.72)$$

De este modo, el conjunto de parámetros del sistema neuro-difuso corresponde a $\mathbf{H} = \{v_{ij}, \sigma_{ij}, \gamma_{ij}\}$ y de forma general, un posible parámetro de este conjunto es $h_{ij} \in \mathbf{H}$.

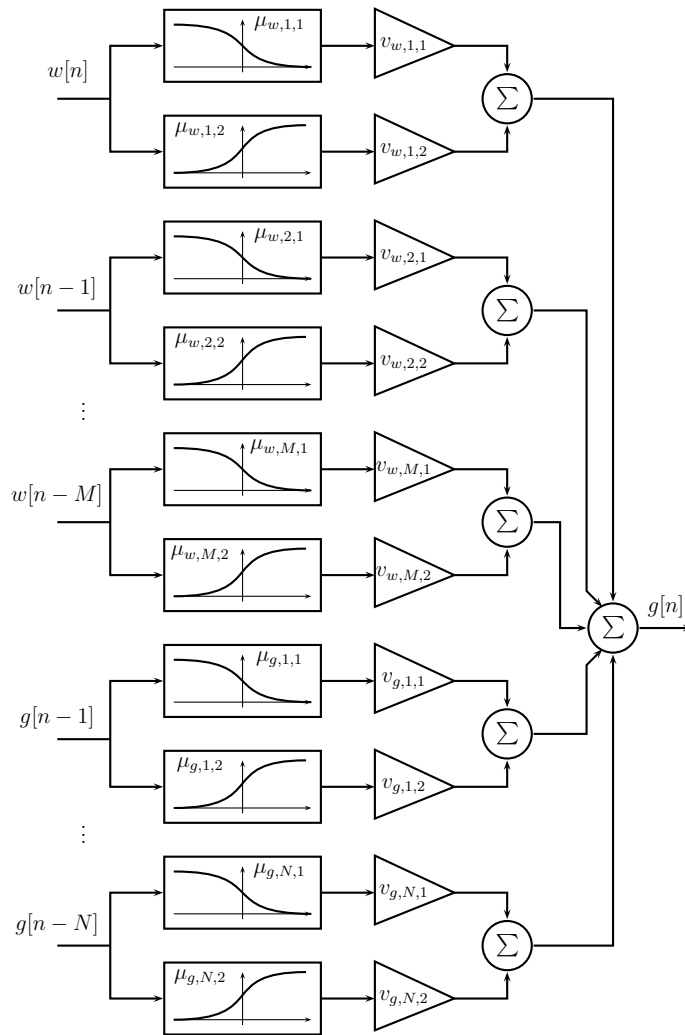


Figura 11.17: Esquema del sistema neuro-difuso compacto.

11.10. Ecuaciones de entrenamiento para la identificación de la planta

La ecuación (11.73) muestra la expresión de entrada-salida para el sistema utilizado en la identificación. La salida de la planta se puede calcular como:

$$f[n] = y_c + \sum_{l=1}^M \sum_{r=1}^N y_{lr} \mu_{lr}(w_l) \quad (11.73)$$

Para mostrar la deducción de las ecuaciones se considera $M = 5$ y $N = 2$, así, $w_l \in \{y[n-1], y[n-2], u[n], u[n-1], u[n-2], \}$. Para cada entrada w_l se puede definir una función f_l de la forma:

$$f_l = \sum_{r=1}^2 y_{lr} \mu_{lr}(w_l) = y_{l1} \mu_{l1}(w_l) + y_{l2} \mu_{l2}(w_l) \quad (11.74)$$

Por su parte, la función de pertenencia $\mu_{lr}(w_l)$ es:

$$\mu_{lr}(w_l) = (1 + e^{-\rho_{lr}(w_l - \delta_{lr})})^{-1} \quad (11.75)$$

De esta manera el conjunto de parámetros \mathbf{H}_p corresponde a $h_{lr} \in \{y_c, y_{lr}, \rho_{lr}, \delta_{lr}\}$; donde y_c es un valor constante, y_{lr} es el actuador virtual, w_l es el dato de entrada al sistema, δ_{lr} y ρ_{lr} son parámetros de la respectiva función de pertenencia.

Con la finalidad de establecer las ecuaciones para el ajuste de los parámetros del sistema neuro-difuso se busca minimizar el error:

$$J_p = \frac{1}{2} [f(w[n]) - y[n]]^2 \quad (11.76)$$

Donde $w[n]$ son los datos de entrada y $y[n]$ los datos del comportamiento de la salida del sistema. Para un tiempo n estos datos forman las respectivas parejas de entrada salida de entrenamiento. Se debe tener presente que α corresponde al ritmo de aprendizaje.

Para llevar a cabo el entrenamiento de los parámetros, se calculan las derivadas del error teniendo:

$$y_c(k+1) = y_c(k) - \alpha \left. \frac{dJ_p}{dy_c} \right|_n \quad (11.77)$$

$$y_{lr}(k+1) = y_{lr}(k) - \alpha \left. \frac{dJ_p}{dy_{lr}} \right|_n \quad (11.78)$$

$$\delta_{lr}(k+1) = \delta_{lr}(k) - \alpha \left. \frac{dJ_p}{d\delta_{lr}} \right|_n \quad (11.79)$$

$$\rho_{lr}(k+1) = \rho_{lr}(k) - \alpha \left. \frac{dJ_p}{d\rho_{lr}} \right|_n \quad (11.80)$$

De esta forma, para actualizar los parámetros se utilizan las ecuaciones:

$$y_c(k+1) = y_c(k) - \alpha (f - y) \quad (11.81)$$

$$y_{lr}(k+1) = y_{lr}(k) - \alpha (f - y) \mu_{lr}(w_l[n]) \quad (11.82)$$

$$\delta_{lr}(k+1) = \delta_{lr}(k) - \alpha (f - y) y_{lr}(-1) (1 + e^{-\rho_{lr}(w_l - \delta_{lr})})^{-2} e^{-\rho_{lr}(w_l - \delta_{lr})} \rho_{lr} \quad (11.83)$$

$$\rho_{lr}(k+1) = \rho_{lr}(k) - \alpha (f - y) y_{lr} (1 + e^{-\rho_{lr}(w_l - \delta_{lr})})^{-2} e^{-\rho_{lr}(w_l - \delta_{lr})} (w_l - \delta_{lr}) \quad (11.84)$$

En la figura 11.18 se puede apreciar el algoritmo para la identificación de la planta mediante la adaptación de parámetros del sistema neuro-difuso.

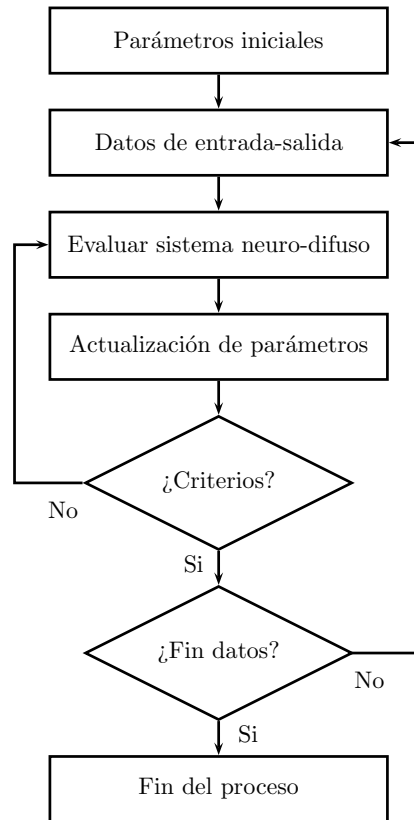


Figura 11.18: Algoritmo para la identificación de la planta.

El primer paso consiste en escoger el sistema neuro-difuso y los parámetros iniciales, los cuales se pueden determinar mediante un conocimiento preliminar del proceso. Luego, se toma un par de datos de entrada salida $(u[n], y[n])$ con $n = 1, 2, \dots, N_T$, en el siguiente paso se calcula la salida para el par de entrada-salida en la k -ésima etapa de entrenamiento con $k = 0, 1, 2, \dots, K_T$, paso seguido, se actualizan los parámetros del sistema neuro-difuso $y_c(k+1)$, $y_{lr}(k+1)$, $\delta_{lr}(k+1)$ y $\rho_{lr}(k+1)$. Posteriormente, se regresa al paso de la evaluación del sistema neuro-difuso con $k = k + 1$, hasta que la función de ajuste $J_p(k)$ sea menor que un ε definido, o hasta que k sea igual a un número definido K_T . En el caso que se cumpla con alguna de las anteriores condiciones (criterios), se toma la siguiente pareja de entrada-salida $(u[n+1], y[n+1])$, haciendo $n = n + 1$ y se repite todo el proceso hasta completar el número total de datos N_T .

11.10.1. Ecuaciones de entrenamiento para el controlador

Teniendo presente la arquitectura II del sistema neuro-difuso compacto utilizado, la salida del controlador se puede calcular como:

$$u[n] = \sum_{i=1}^M \sum_{j=1}^N v_{ij} \mu_{ij}(x_i) \quad (11.85)$$

Para determinar las ecuaciones de entrenamiento, a manera de ejemplo se considera el caso donde $M = 5$ y $N = 2$, de tal forma que se tiene $x_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2], \}$. Para cada entrada x_i se puede definir una función f_i de la forma:

$$f_i = \sum_{j=1}^2 f_{ij} = \sum_{j=1}^2 v_{ij} \mu_{ij}(x_i) = v_{i1} \mu_{i1}(x_i) + v_{i2} \mu_{i2}(x_i) \quad (11.86)$$

Como es de apreciar $f_{ij} = v_{ij} \mu_{ij}(x_i)$, donde la función de pertenencia $\mu_{ij}(x_i)$ corresponde a la siguiente ecuación:

$$\mu_{ij}(x_i) = \frac{1}{1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})}} \quad (11.87)$$

El conjunto de parámetros del controlador corresponde a $\mathbf{H}_c = \{v_{ij}, \sigma_{ij}, \gamma_{ij}\}$, donde un posible parámetro puede ser $h_c = h_{ij} \in \mathbf{H}_c$. Con la finalidad de establecer las ecuaciones para el entrenamiento de parámetros, en primer lugar, se tiene que la función de desempeño es:

$$J_c = \frac{1}{2} [P(e[n])^2 + Q(u[n])^2] \quad (11.88)$$

En esta ecuación $P \in \mathbb{R}^+$ es un valor que pondera el error y $Q \in \mathbb{R}^+$ es un valor que pondera la energía suministrada a la planta. De esta forma se logra un balance entre la señal de error y la energía entregada a la planta, permitiendo ajustar el comportamiento dinámico del sistema en lazo cerrado. En segundo lugar, la ecuación de la planta para el caso considerado es:

$$y[n] = f_p(y[n-1], y[n-2], u[n], u[n-1], u[n-2]) \quad (11.89)$$

Es de señalar que en este caso se considera un modelo de referencia ideal de tal forma que $y_r = r$. Por su parte, la ecuación del error es:

$$e[n] = r[n] - y[n] \quad (11.90)$$

Considerando la arquitectura del controlador (figura 11.17) y las expresiones para f_i (ecuaciones (11.14) y (11.16)) la dinámica del controlador se encuentra dada por:

$$u[n] = f_1(u[n-1]) + f_2(u[n-2]) + f_3(e[n]) + f_4(e[n-1]) + f_5(e[n-2]) \quad (11.91)$$

La derivada de la planta con respecto a los parámetros del controlador es:

$$\frac{dy[n]}{dh_c} = \frac{d}{dh_c} f_p(y[n-1], y[n-2], u[n], u[n-1], u[n-2]) \quad (11.92)$$

La respectiva derivada del error $e[n] = r[n] - y[n]$ para un parámetro del controlador h_c es:

$$\frac{de[n]}{dh_c} = -\frac{dy[n]}{dh_c} \quad (11.93)$$

De la misma forma, la derivada de la acción de control $u[n]$ con respecto a un posible parámetro del controlador h_{ij} es:

$$\frac{du[n]}{dh_{ij}} = \frac{df_1(u[n-1])}{dh_{ij}} + \frac{df_2(u[n-2])}{dh_{ij}} + \frac{df_3(e[n])}{dh_{ij}} + \frac{df_4(e[n-1])}{dh_{ij}} + \frac{df_5(e[n-2])}{dh_{ij}} \quad (11.94)$$

Como es de apreciar, se deben calcular las derivadas de los parámetros con respecto a la planta y el controlador.

11.10.2. Derivadas de los parámetros del controlador respecto a la planta

Para establecer una expresión general de la dinámica de la planta f_p se tiene que $w_l \in \{y[n-1], y[n-2], u[n], u[n-1], u[n-2]\}$. De este modo, la ecuación para calcular la salida de la planta es:

$$y = y_c + \sum_{l=1}^M \sum_{r=1}^N y_{lr} \mu_{lr}(w_l) \quad (11.95)$$

con:

$$\mu_{lr}(x_l) = (1 + e^{-\sigma_{lr}(x_l - \gamma_{lr})})^{-1} \quad (11.96)$$

entonces, la derivada de $y[n]$ respecto a los parámetros del controlador $h_c = h_{ij} \in \mathbf{H}_c$ es:

$$\frac{dy[n]}{dh_{ij}} = \sum_{l=1}^M \sum_{r=1}^N y_{lr} \frac{d\mu_{lr}(w_l)}{dh_{ij}} = \sum_{l=1}^M \sum_{r=1}^N y_{lr} \frac{d\mu_{lr}(w_l)}{dw_l} \frac{dw_l}{dh_{ij}} \quad (11.97)$$

Es decir:

$$\frac{dy}{dh_c} = \left(\sum_{r=1}^N y_{1,r} \frac{d\mu_{1r}(w_1)}{dw_1} \right) \frac{dw_1}{dh_c} + \left(\sum_{r=1}^N y_{2,r} \frac{d\mu_{2r}(w_2)}{dw_2} \right) \frac{dw_2}{dh_c} + \left(\sum_{r=1}^N y_{3,r} \frac{d\mu_{3r}(w_3)}{dw_3} \right) \frac{dw_3}{dh_c} + \dots \quad (11.98)$$

Con:

$$\frac{d\mu_{ij}(w_l)}{dw_l} = (-1) (1 + e^{-\rho_{lr}(w_l - \delta_{lr})})^{-2} e^{-\rho_{lr}(w_l - \delta_{lr})} (-\rho_{lr}) \quad (11.99)$$

y

$$C_l = \sum_{r=1}^N y_{lr} \frac{d\mu_{lr}(w_l)}{dw_l} \quad (11.100)$$

Para el caso en consideración donde $M = 5$ y $N = 2$, se tiene:

$$\frac{dy[n]}{dh_{ij}} = C_1 \frac{dw_1}{dh_{ij}} + C_2 \frac{dw_2}{dh_{ij}} + C_3 \frac{dw_3}{dh_{ij}} + C_4 \frac{dw_4}{dh_{ij}} + C_5 \frac{dw_5}{dh_{ij}} \quad (11.101)$$

11.10.3. Derivadas de los parámetros el controlador respecto al controlador

Para determinar las respectivas derivadas de los parámetros del controlador h_{ij} considerando las ecuaciones del controlador f_{km} se tiene que:

$$\frac{df_k(x_k)}{dh_{ij}} = \sum_{m=1}^2 \frac{f_{km}(x_k)}{dh_{ij}} = \frac{df_{k1}(x_k)}{dh_{ij}} + \frac{df_{k2}(x_k)}{dh_{ij}} \quad (11.102)$$

Como es de apreciar, se presentan diferentes casos para los valores de i y k . Considerando la ecuación (11.14), para el caso cuando $k \neq i$ se tiene:

$$\begin{aligned} \frac{df_k(x_k)}{dh_{ij}} &= \frac{df_{k1}(x_k)}{dx_k} \frac{dx_k}{dh_{ij}} + \frac{df_{k2}(x_k)}{dx_k} \frac{dx_k}{dh_{ij}} \\ &= \left[\frac{df_{k1}(x_k)}{dx_k} + \frac{df_{k2}(x_k)}{dx_k} \right] \frac{dx_k}{dh_{ij}} \end{aligned} \quad (11.103)$$

Con $m = 1, 2$ las respectivas derivadas son:

$$\begin{aligned} \frac{df_{km}(x_k)}{dx_k} &= \frac{d}{dx_k} \left(v_{km} (1 + e^{-\sigma_{km}(x_k - \gamma_{km})})^{-1} \right) \\ &= v_{km} (1 + e^{-\sigma_{km}(x_k - \gamma_{km})})^{-2} e^{-\sigma_{km}(x_k - \gamma_{km})} \sigma_{km} \end{aligned} \quad (11.104)$$

Para el caso donde $k = i$ y $m = j$ considerando que f_{ij} depende de forma implícita y explícita de h_{ij} , la respectiva derivada se puede escribir como:

$$\frac{df_{ij}(x_i)}{dh_{ij}} = F_{h_{ij}} + K_{h_{ij}} \frac{dx_i}{dh_{ij}} \quad (11.105)$$

Por su parte cuando $k = i$ y $m = 1$ se tiene:

$$\begin{aligned} \frac{df_i(x_i)}{dh_{i1}} &= \frac{df_{i1}(x_i)}{dh_{i1}} + \frac{df_{i2}(x_i)}{dx_i} \frac{dx_i}{dh_{i1}} \\ &= F_{h_{i1}} + K_{h_{i1}} \frac{dx_i}{dh_{i1}} + \frac{df_{i2}(x_i)}{dx_i} \frac{dx_i}{dh_{i1}} \\ &= F_{h_{i1}} + \left(K_{h_{i1}} + \frac{df_{i2}(x_i)}{dx_i} \right) \frac{dx_i}{dh_{i1}} \end{aligned} \quad (11.106)$$

También cuando $k = i$ y $m = 2$ se tiene:

$$\begin{aligned} \frac{df_i(x_i)}{dh_{i2}} &= \frac{df_{i1}(x_i)}{dx_i} \frac{dx_i}{dh_{i2}} + \frac{df_{i2}(x_i)}{dh_{i2}} \\ &= \frac{df_{i1}(x_i)}{dx_i} \frac{dx_i}{dh_{i2}} + F_{h_{i2}} + K_{h_{i2}} \frac{dx_i}{dh_{i2}} \\ &= F_{h_{i2}} + \left(\frac{df_{i1}(x_i)}{dx_i} + K_{h_{i2}} \right) \frac{dx_i}{dh_{i2}} \end{aligned} \quad (11.107)$$

Para calcular las derivadas que se requieren en las anteriores ecuaciones cuando $k = i$ y $m = j$ en primer lugar se tiene:

$$\frac{df_{ij}(x_i)}{dx_i} = v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \quad (11.108)$$

En segundo lugar para las otras derivadas:

$$\frac{df_{ij}(x_i)}{dh_{ij}} = \frac{d}{dh_{ij}} \left(v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} \right) \quad (11.109)$$

Para el parámetro $h_{ij} = v_{ij}$ se tiene:

$$\frac{df_{ij}(x_i)}{dv_{ij}} = (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} - v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} (-\sigma_{ij}) \frac{dx_i}{dv_{ij}} \quad (11.110)$$

Tomando el parámetro $h_{ij} = \sigma_{ij}$ se tiene:

$$\frac{df_{ij}(x_i)}{d\sigma_{ij}} = -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \left(\gamma_{ij} - x_i - \sigma_{ij} \frac{dx_i}{d\sigma_{ij}} \right) \quad (11.111)$$

Con el parámetro $h_{ij} = \gamma_{ij}$ se tiene:

$$\frac{df_{ij}(x_i)}{d\gamma_{ij}} = -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \left(\sigma_{ij} - \sigma_{ij} \frac{dx_i}{d\gamma_{ij}} \right) \quad (11.112)$$

De forma general estas ecuaciones se pueden escribir como:

$$\begin{aligned} \frac{df_{ij}(x_i)}{dv_{ij}} &= F_{v_{ij}} + K_{v_{ij}} \frac{dx_i}{dv_{ij}} \\ \frac{df_{ij}(x_i)}{d\sigma_{ij}} &= F_{\sigma_{ij}} + K_{\sigma_{ij}} \frac{dx_i}{d\sigma_{ij}} \\ \frac{df_{ij}(x_i)}{d\gamma_{ij}} &= F_{\gamma_{ij}} + K_{\gamma_{ij}} \frac{dx_i}{d\gamma_{ij}} \end{aligned} \quad (11.113)$$

Donde:

$$\begin{aligned} F_{v_{ij}} &= (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-1} \\ F_{\sigma_{ij}} &= -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} (\gamma_{ij} - x_i) \\ F_{\gamma_{ij}} &= -v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \\ K_{v_{ij}} &= v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \\ K_{\sigma_{ij}} &= v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \\ K_{\gamma_{ij}} &= v_{ij} (1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})})^{-2} e^{-\sigma_{ij}(x_i - \gamma_{ij})} \sigma_{ij} \end{aligned} \quad (11.114)$$

11.10.4. Proceso para el entrenamiento del controlador

Tomando $w_i \in \{y[n-1], y[n-2], u[n], u[n-1], u[n-2]\}$ y $x_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2]\}$, entonces, las ecuaciones que implementan la dinámica de los parámetros de entrenamiento son:

$$\frac{de[n]}{dh_{ij}} = -\frac{dy}{dh_{ij}} \quad (11.115)$$

$$\frac{dy[n]}{dh_{ij}} = C_1 \frac{dw_1}{dh_{ij}} + C_2 \frac{dw_2}{dh_{ij}} + C_3 \frac{dw_3}{dh_{ij}} + C_4 \frac{dw_4}{dh_{ij}} + C_5 \frac{dw_5}{dh_{ij}} \quad (11.116)$$

$$\frac{du[n]}{dh_{ij}} = \frac{df_1(x_1)}{dh_{ij}} + \frac{df_2(x_2)}{dh_{ij}} + \frac{df_3(x_3)}{dh_{ij}} + \frac{df_4(x_4)}{dh_{ij}} + \frac{df_5(x_5)}{dh_{ij}} \quad (11.117)$$

En esta última ecuación si $k \neq i$ entonces:

$$\frac{df_k(x_k)}{dh_{ij}} = \left[\frac{df_{k1}(x_k)}{dx_k} + \frac{df_{k2}(x_k)}{dx_k} \right] \frac{dx_k}{dh_{ij}} \quad (11.118)$$

Por otra, parte si $k = i$ se tiene:

$$\frac{df_i(x_i)}{dh_{i1}} = F_{h_{i1}} + \left(K_{h_{i1}} + \frac{df_{i2}(x_i)}{dx_i} \right) \frac{dx_i}{dh_{i1}} \quad (11.119)$$

$$\frac{df_i(x_i)}{dh_{i2}} = F_{h_{i2}} + \left(\frac{df_{i1}(x_i)}{dx_i} + K_{h_{i2}} \right) \frac{dx_i}{dh_{i2}} \quad (11.120)$$

Finalmente, la actualización de los parámetros se hace de la forma:

$$h_{ij}(q+1) = h_{ij}(q) - \alpha \left(Pe[n] \frac{de[n]}{dh_{ij}} + Qu[n] \frac{du[n]}{dh_{ij}} \right) \quad (11.121)$$

Donde α corresponde a la tasa de aprendizaje. Considerando la actualización de cada parámetro se tiene:

$$v_{ij}(q+1) = v_{ij}(q) - \alpha \left(Pe[n] \frac{de[n]}{dv_{ij}} + Qu[n] \frac{du[n]}{dv_{ij}} \right) \quad (11.122)$$

$$\sigma_{ij}(q+1) = \sigma_{ij}(q) - \alpha \left(Pe[n] \frac{de[n]}{d\sigma_{ij}} + Qu[n] \frac{du[n]}{d\sigma_{ij}} \right) \quad (11.123)$$

$$\gamma_{ij}(q+1) = \gamma_{ij}(q) - \alpha \left(Pe[n] \frac{de[n]}{d\gamma_{ij}} + Qu[n] \frac{du[n]}{d\gamma_{ij}} \right) \quad (11.124)$$

El algoritmo para el entrenamiento del controlador neuro-difuso se puede apreciar en la figura 11.19. El primer paso consiste en escoger la configuración inicial de parámetros del controlador. En el siguiente paso, se calcula la salida del sistema de control utilizando el modelo neuro-difuso de la planta. Luego se procede a realizar el ajuste de los parámetros del controlador neuro-difuso utilizando las respectivas ecuaciones que involucran la dinámica del sistema de control y las respectivas derivadas de los parámetros (ecuaciones

(11.121), (11.115) y subsecuentes). Es importante señalar que los parámetros ajustados se almacenan en una variable auxiliar ya que durante este paso el controlador no utiliza estos valores. Luego, se regresa al paso donde se evalúa el sistema de control con $n = n + 1$ repitiendo este proceso hasta completar el tiempo de simulación N_T . En el caso de completar el tiempo de simulación, se actualizan los parámetros del controlador con los valores optimizados y se regresa al paso de evaluación del sistema de control para una nueva iteración $q = q + 1$, hasta que la función objetivo $J_c(q) = \sum_{n=1}^{N_T} J_c(q, n)$ sea menor que un ε definido, o hasta que q sea igual a un número determinado K_T .

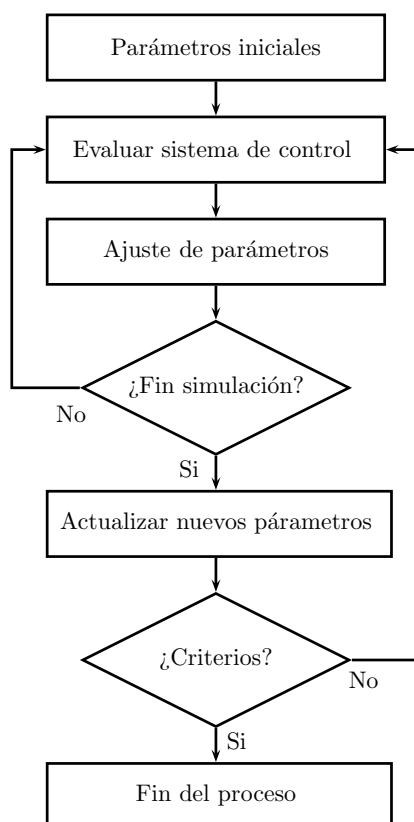


Figura 11.19: Algoritmo para el entrenamiento del controlador neuro-difuso.

11.11. Modelo del sistema hidráulico

El modelo está compuesto por dos tanques esféricos acoplados: uno principal y otro auxiliar (de reserva), que es llenado tomando parte del tanque principal, tal como se aprecia en la figura 11.20. El tanque auxiliar también permite la posibilidad de extraer el fluido por una tubería de menor diámetro.

Este es un esquema donde pueden existir cambios en los parámetros de las llaves cuando se necesite extraer fluido del tanque auxiliar o cuando se manipula la llave que conecta los dos tanques (para suspender o también para el llenado rápido del tanque auxiliar).

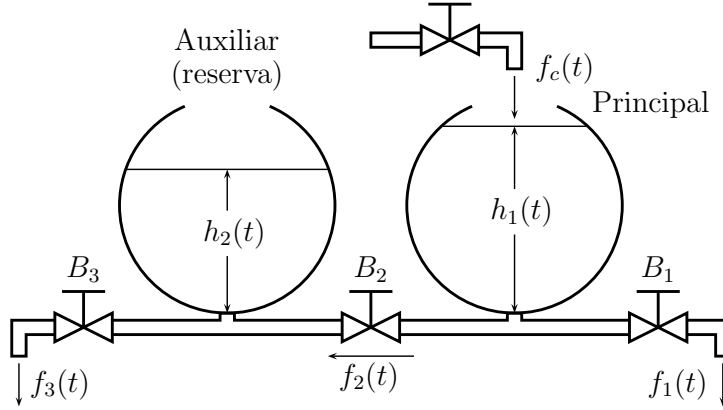


Figura 11.20: Sistema hidráulico de dos tanques esféricos acoplados.

Para un tanque, el líquido ingresa con un flujo F_{in} y sale con F_{out} respectivamente; la altura transitoria $h(t)$ del líquido se puede encontrar mediante un balance de masa en el tanque [265, 266]. De esta manera, la ecuación general de balance hidráulico para un tanque es:

$$\frac{dV}{dt} = F_{in} - F_{out} \quad (11.125)$$

La derivada del volumen se puede calcular como:

$$\frac{dV}{dt} = \frac{dV}{dh} \frac{dh}{dt} \quad (11.126)$$

Considerando que:

$$(R - h)^2 + r_s^2 = R^2 \quad (11.127)$$

Dependiendo de la altura, el volumen del tanque es:

$$V = \frac{1}{3}\pi h^2(3R - h) \quad (11.128)$$

La derivada temporal del volumen es:

$$\frac{dV}{dt} = (2\pi R h - \pi h^2) \frac{dh}{dt} \quad (11.129)$$

$$\frac{dV}{dt} = \pi(2R h - h^2) \frac{dh}{dt} \quad (11.130)$$

Para el sistema en consideración de dos tanques acoplados, la ecuación del tanque 1 (principal) es:

$$\pi(2R_1 h_1 - h_1^2) \frac{dh_1}{dt} = f_c(t) - a_2 \sqrt{2g|h_1 - h_2|} \text{sign}(h_1 - h_2) - a_1 \sqrt{2g h_1} \quad (11.131)$$

Para el tanque 2 (auxiliar) se tiene:

$$\pi(2R_2 h_2 - h_2^2) \frac{dh_2}{dt} = a_2 \sqrt{2g|h_1 - h_2|} \text{sign}(h_1 - h_2) - a_3 \sqrt{2g h_2} \quad (11.132)$$

Donde $a_i = \pi(D_i/2)^2$, siendo D_i el diámetro de tubo i en metros. De forma compacta, la ecuación del tanque 1 se puede escribir como:

$$\frac{dh_1}{dt} = \frac{f_c(t) - B_2\sqrt{|h_1 - h_2|}\text{sign}(h_1 - h_2) - B_1\sqrt{h_1}}{\pi(2R_1h_1 - h_1^2)} \quad (11.133)$$

Para el tanque 2 se tiene:

$$\frac{dh_2}{dt} = \frac{B_2\sqrt{|h_1 - h_2|}\text{sign}(h_1 - h_2) - B_3\sqrt{2gh_2}}{\pi(2R_2h_2 - h_2^2)} \quad (11.134)$$

En la figura 11.21 se puede observar el esquema empleado para la simulación del sistema hidráulico compuesto por dos tanques esféricos acoplados.

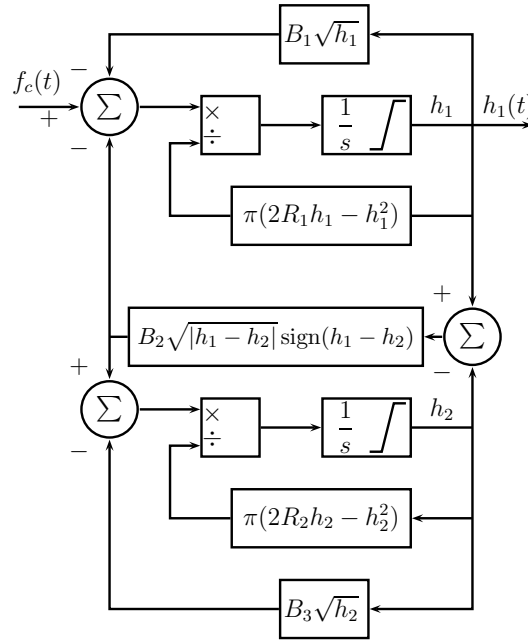


Figura 11.21: Esquema de empleo para la simulación del sistema hidráulico.

Los parámetros del modelo son:

- $D_1 = 0,04 \text{ m}$
- $D_2 = 0,06 \text{ m}$
- $D_3 = 0,02 \text{ m}$
- $R_1 = 0,6 \text{ m}$
- $R_2 = 0,6 \text{ m}$
- $g = 9,8 \text{ m/s}^2$
- $a_1 = \pi(D_1/2)^2 = 0,0012566 \text{ m}^2$
- $a_2 = \pi(D_2/2)^2 = 0,0028274 \text{ m}^2$
- $a_3 = \pi(D_3/2)^2 = 0,0003141 \text{ m}^2$
- $B_1 = a_1\sqrt{2g} = 0,0055634 \text{ m}^{5/2}/\text{s}$
- $B_2 = a_2\sqrt{2g} = 0,0125180 \text{ m}^{5/2}/\text{s}$
- $B_3 = a_3\sqrt{2g} = 0,0013908 \text{ m}^{5/2}/\text{s}$

Para la implementación del modelo se considera variación en los coeficientes B_2 y B_3 (de las llaves). Estos valores se encuentran desde cero para una llave cerrada hasta el valor de apertura total dado por el respectivo diámetro de la llave. También se considera un actuador en la válvula de entrada que permite tener un flujo medio de $0,0015 \text{ m}^3/\text{s}$.

11.12. Configuración de experimentos

Con el fin de realizar el diseño experimental se consideran diferentes aspectos para observar las características del sistema de control adaptativo. Un primer aspecto a comparar con los experimentos es el desempeño del controlador con y sin el proceso adaptativo. En segunda instancia, se tiene la configuración empleada para el sistema de control, es decir, el número de retardos en la entrada y la salida.

Para el controlador y el modelo de la planta existen diferentes configuraciones dependiendo de las entradas y las realimentaciones. La tabla 11.2 muestra las configuraciones experimentales consideradas para los casos adaptativo y no adaptativo.

Entradas	Realimentaciones	No Adaptativo	Adaptativo
2	2	IN2FB2AD0	IN2FB2AD1
2	3	IN2FB3AD0	IN2FB3AD1
3	2	IN3FB2AD0	IN3FB2AD1

Tabla 11.2: Configuraciones experimentales.

También se considera la configuración para los valores de P y Q de la siguiente forma:

- Caso PQ1: $P = 1, Q = 0$.
- Caso PQ2: $P = 1, Q = 0,0001$.
- Caso PQ3: $P = 1, Q = 0,001$.
- Caso PQ4: $P = 1, Q = 0,01$.

En la figura 11.22 se puede observar el esquema general de la arquitectura empleada de los sistemas neuro-difusos para la identificación de la planta y el controlador considerando los retardos de la entrada y la salida.

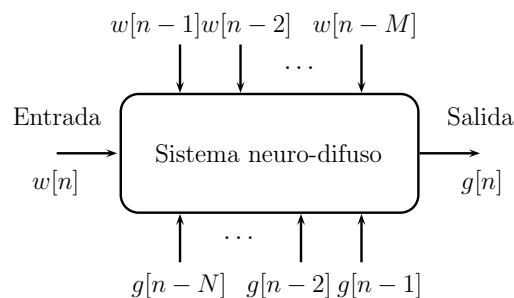


Figura 11.22: Modelo general de los sistemas neuro-difusos.

El tiempo entre variaciones de la planta se toma de 500s, y el tiempo mientras se presenta en funcionamiento del controlador corresponde a 100s, es decir, se realizan cinco procesos de adaptación durante el tiempo que se producen las variaciones de la planta. Cada configuración del sistema neuro-difuso se ejecuta 10 veces y durante cada ejecución se presentan 10 variaciones (de forma aleatoria) de los parámetros de la planta.

11.12.1. Método empleado para el análisis estadístico de resultados

Dada la naturaleza estocástica de la variación de la planta se puede presentar variabilidad en los resultados cuando se emplea una determinada configuración del sistema de control. Para observar si existe diferencia significativa en los resultados obtenidos por el sistema de control, se lleva a cabo una prueba estadística de hipótesis, donde:

- H_0 Hipótesis nula: Los resultados obtenidos por los sistemas de control presentan valores medios iguales.
- H_1 Hipótesis alternativa: Los resultados obtenidos por los sistemas de control no presentan valores medios iguales.

Con la finalidad de realizar la prueba de hipótesis se utiliza el procedimiento de Kruskal-Wallis y para la comparación entre grupos la prueba estadística de Bonferroni-Dunn.

El índice de desempeño considerado corresponde al mostrado en la ecuación (11.135), siendo r el valor de salida deseado, y la respuesta del sistema de control, u la señal de control, P y Q los valores de moderación del error y la acción de control, finalmente N_T el número total de datos.

$$J_{pq} = \frac{1}{N_T} \sum_{n=1}^{N_T} P(r[n] - y[n])^2 + Q(u[n])^2 \quad (11.135)$$

11.13. Resultados experimentales

En esta sección se presentan los resultados para las diferentes implementaciones de los valores de P y Q , considerando el desempeño del sistema de control con y sin proceso de adaptación.

En primer lugar, se muestra el análisis estadístico realizado a los datos obtenidos de las ejecuciones de cada configuración del sistema neuro-difuso. En segundo lugar, se tiene la simulación que permite observar el comportamiento adaptativo del sistema de control propuesto.

El resumen estadístico de los datos obtenidos para el valor de J_{pq} se muestra en la tabla 11.3, donde se tiene el valor mínimo, máximo, la desviación estándar (*Standard Deviation* STD) y el valor promedio. Adicionalmente, en la figura 11.23 se presenta el diagrama de tallos y hojas de los resultados obtenidos.

Caso PQ1					Caso PQ3				
Configuración	Mínimo	Máximo	STD	Promedio	Configuración	Mínimo	Máximo	STD	Promedio
IN2FB2AD0	0.0047906	0.015874	0.0032232	0.009784	IN2FB2AD0	0.023763	0.030369	0.0026901	0.026776
IN2FB3AD0	0.0099272	0.027479	0.0051299	0.018239	IN2FB3AD0	0.023324	0.029607	0.0024872	0.026287
IN3FB2AD0	0.0067105	0.015556	0.0028277	0.010808	IN3FB2AD0	0.024936	0.031622	0.0026473	0.02768
IN2FB2AD1	0.0016929	0.0041944	0.00082633	0.0026377	IN2FB2AD1	0.018467	0.021551	0.0009968	0.020025
IN2FB3AD1	0.002537	0.005007	0.00073616	0.0041495	IN2FB3AD1	0.018279	0.021115	0.001005	0.019767
IN3FB2AD1	0.0026438	0.0039094	0.00038955	0.0032586	IN3FB2AD1	0.018963	0.022322	0.0011157	0.020379

Caso PQ2					Caso PQ4				
Configuración	Mínimo	Máximo	STD	Promedio	Configuración	Mínimo	Máximo	STD	Promedio
IN2FB2AD0	0.0067916	0.017077	0.0030718	0.011438	IN2FB2AD0	0.16418	0.20436	0.014263	0.18387
IN2FB3AD0	0.0064633	0.016248	0.0028845	0.010914	IN2FB3AD0	0.16773	0.21161	0.01521	0.18987
IN3FB2AD0	0.0085679	0.017197	0.0028004	0.01252	IN3FB2AD0	0.16241	0.18512	0.0077866	0.17385
IN2FB2AD1	0.0034752	0.0057876	0.00079407	0.0043508	IN2FB2AD1	0.19301	0.22923	0.011786	0.20876
IN2FB3AD1	0.0034494	0.0060122	0.00080979	0.0043082	IN2FB3AD1	0.19333	0.22909	0.011601	0.20912
IN3FB2AD1	0.0042984	0.0055293	0.00041824	0.0049768	IN3FB2AD1	0.15562	0.18351	0.0090453	0.16787

Tabla 11.3: Resumen estadístico de los datos obtenidos.

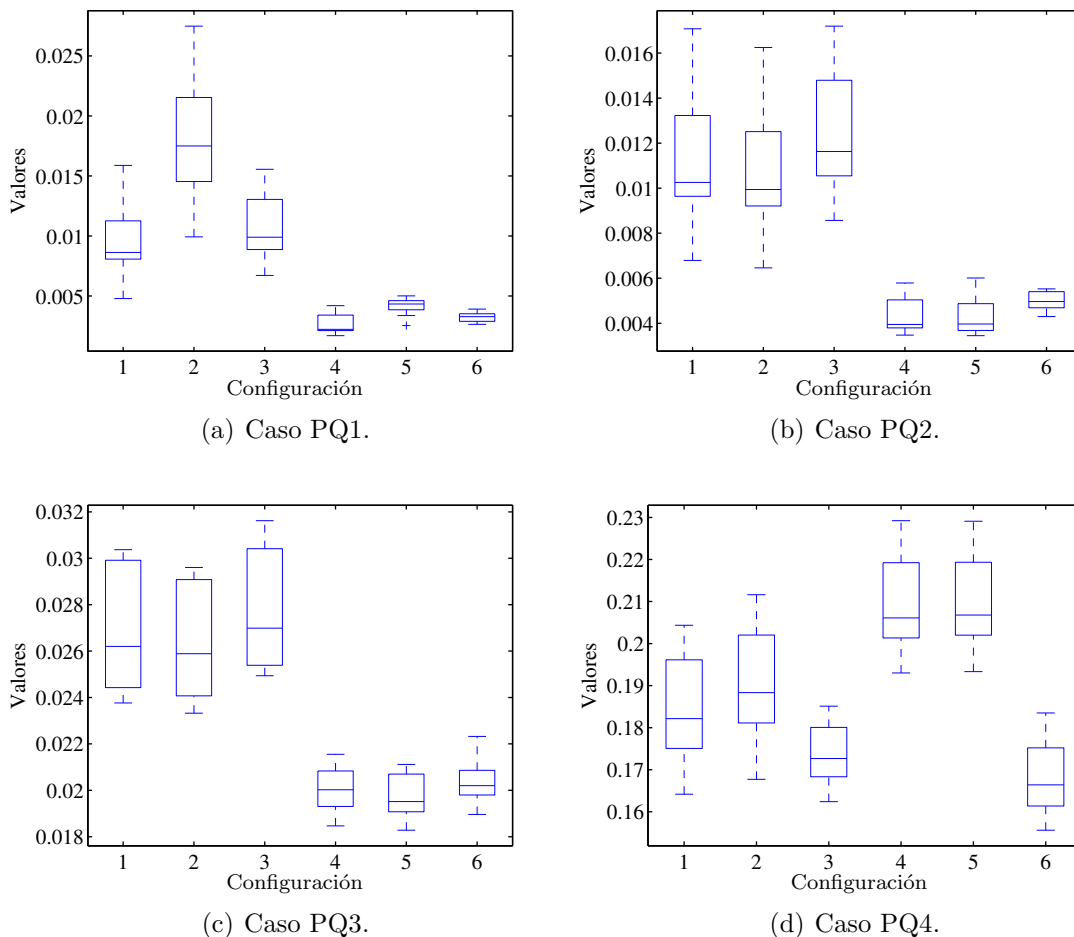


Figura 11.23: Diagrama de tallos y hojas de los resultados obtenidos.

Con los datos obtenidos se realiza la prueba de comparación entre grupos de Kruskal-Wallis obteniendo los respectivos p -value mostrados en la tabla 11.4, donde se aprecia que existen diferencias entre grupos experimentales. Efectuando la prueba de comparación entre grupos de Bonferroni-Dunn se obtiene el resultado de la figura 11.24.

Caso PQ	p -value
1	$8,3413 \times 10^{-10}$
2	$7,7188 \times 10^{-9}$
3	$1,2444 \times 10^{-8}$
4	$3,0143 \times 10^{-7}$

Tabla 11.4: Resultados de la prueba de comparación entre grupos de Kruskal-Wallis.

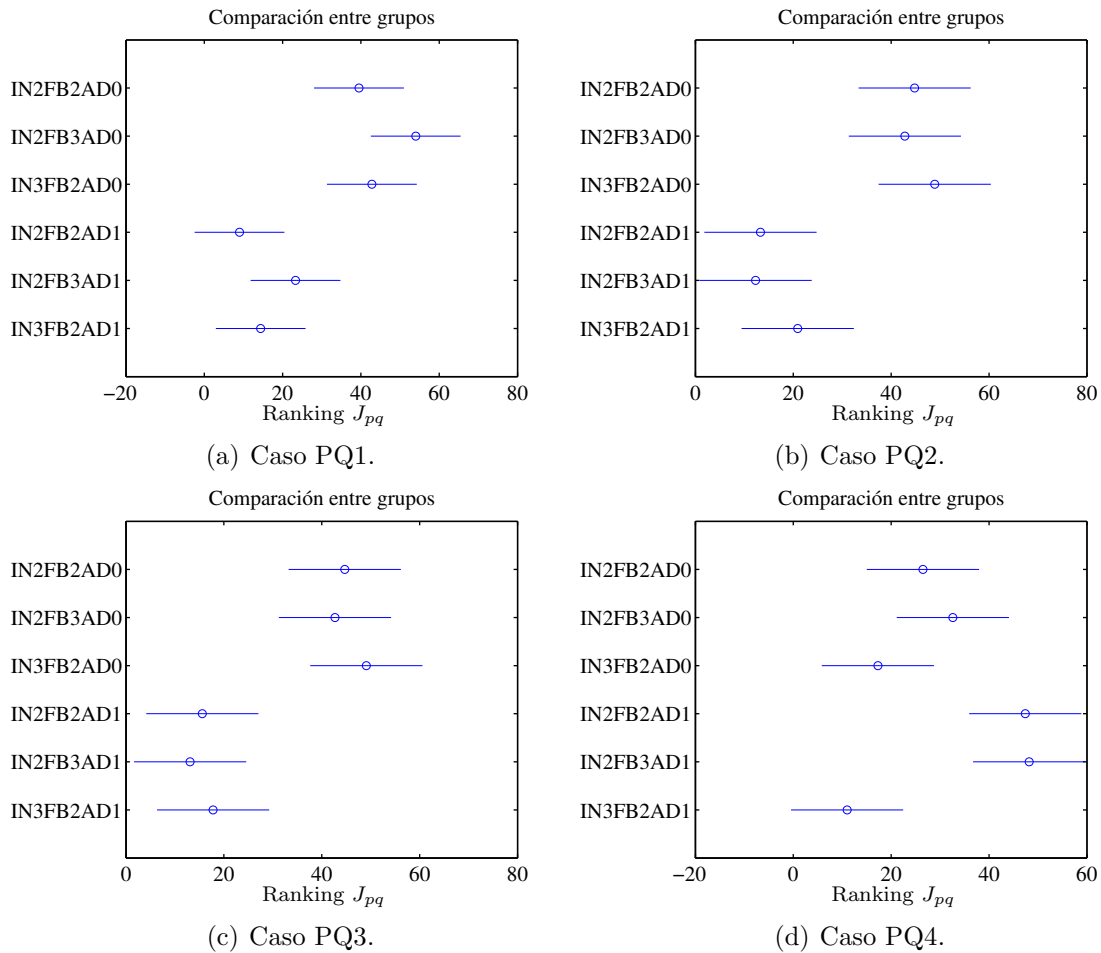


Figura 11.24: Resultados de la comparación entre grupos.

En la figura 11.24 se observa que para los casos PQ1, PQ2 y PQ3 existe diferencia estadística para los grupos que emplean el esquema de control adaptativo y las implementaciones donde no se utiliza. Con estos resultados, se observa que el esquema adaptativo presenta un mejor desempeño para los casos PQ1, PQ2 y PQ3, mientras que para el caso PQ4 no se tiene diferencia estadística significativa. Por lo cual, tomando $P = 1$, un valor adecuado para Q se encuentra entre 0 y 0,001.

11.13.1. Resultados del comportamiento del sistema

Para mostrar de forma cualitativa el comportamiento del sistema de control se toma la configuración IN3FB2AD1; en la figura 11.25 se presenta de forma gráfica la respuesta del

sistema con el controlador convencional y también con el sistema adaptativo. Se aprecia que el controlador efectúa ajustes progresivos para lograr que la salida del sistema llegue a la referencia después que ha ocurrido una variación de la planta.

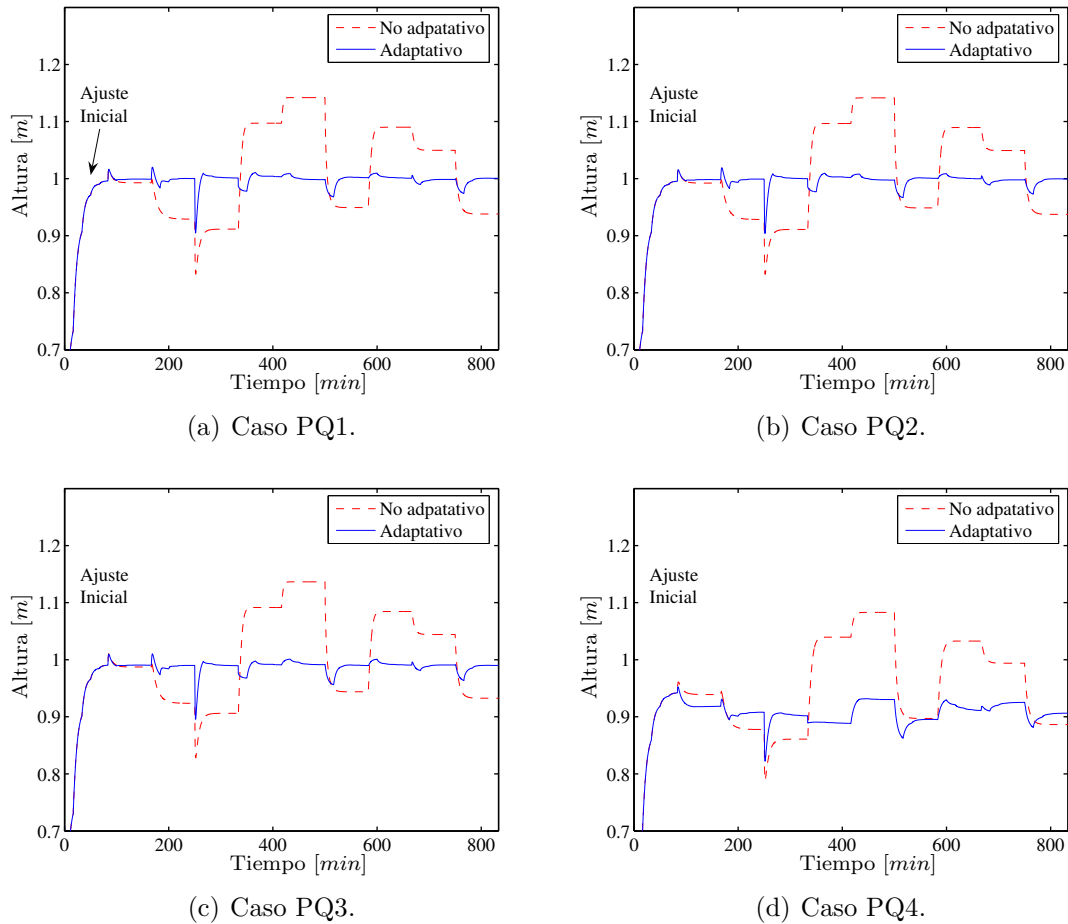
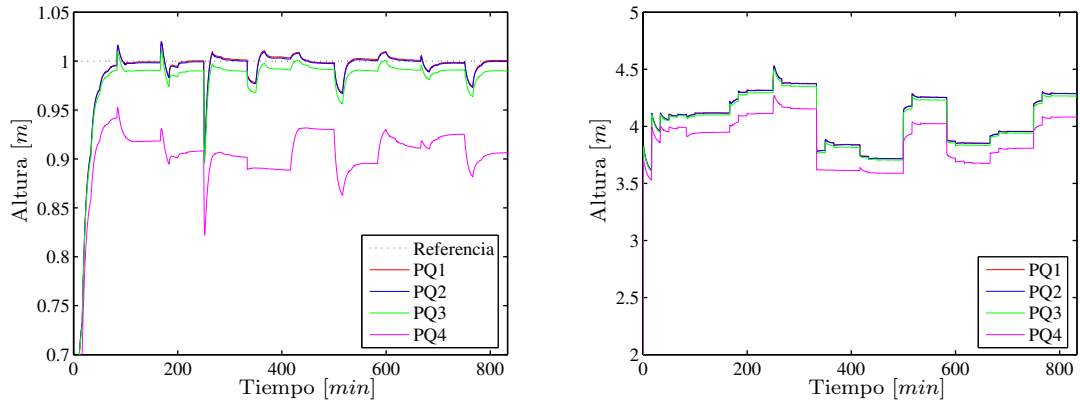


Figura 11.25: Respuesta del sistema de control con y sin proceso de adaptación.

En la figura 11.26(a) se presenta la comparación de la respuesta del sistema para las diferentes configuraciones de P y Q . Por su parte, en la figura 11.26(b) se muestra el resultado del sistema para la acción de control. En estas figuras se aprecia que al disminuir Q se presenta una mayor diferencia entre la salida y la referencia como también un menor valor de la acción de control.

Como aspecto adicional, el detalle de los ajustes realizados por el sistema adaptativo se puede apreciar en la figura 11.27. Es de notar en la simulación el momento cuando se efectúan los respectivos ajustes progresivos del controlador para corregir el cambio que existe al variar la planta. De esta forma, se observa que el sistema logra ajustar el controlador luego de existir una variación de la planta.



(a) Respuesta del sistema de control adaptativo. (b) Acción de control del sistema de adaptativo.

Figura 11.26: Comparación de la respuesta del sistema de control adaptativo para las diferentes configuraciones de P y Q .

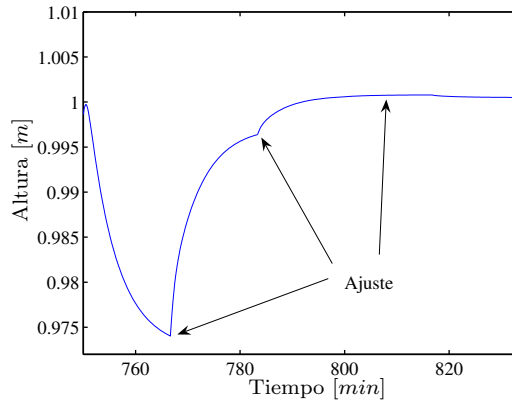


Figura 11.27: Detalle de la respuesta del sistema de control adaptativo.

Capítulo 12

Control de un sistema hidráulico MIMO acoplado

12.1. Introducción

Esta sección describe la implementación de un sistema de control neuro-difuso adaptativo MIMO (*Multiple Input Multiple Output*) que emplea dos sistemas neuro-difusos MIMO, uno para la identificación de la planta y otro para el control. Con este enfoque, el controlador se optimiza utilizando el modelo obtenido durante la identificación de la planta, la cual emplea datos producidos por el funcionamiento del controlador que se optimiza de forma iterativa. La planta a controlar es sistema no lineal hidráulico MIMO compuesto por dos tanques alimentados por una bomba y una válvula de tres vías. Para observar el comportamiento del sistema de control se consideran varias configuraciones experimentales.

12.2. Sistema hidráulico MIMO

El sistema MIMO a controlar consta de dos tanques alimentados por una llave de tres vías y una bomba hidráulica conectada a esta llave para distribuir el flujo a los tanques [267]. La planta se encuentra ubicada en el Departamento de Ingeniería de Sistemas y Automatización de la Universidad de Oviedo. Está diseñada para simular un proceso industrial de transporte y almacenamiento de líquidos. La planta está comandada por un controlador lógico programable (*Programmable Logic Controller* PLC) conectado a un servidor OPC (OLE para control de procesos). Consta de dos tanques inferiores ($D1$ y $D2$) y dos tanques superiores ($D3$ y $D4$), que funcionan mediante un sistema de drenaje en cascada. Los tanques inferiores descargan su caudal a un tanque colector, que actúa como fuente para el bombeo a cada uno de los tanques mediante dos bombas ($P1$ y $P2$) y dos válvulas de tres vías ($V1$ y $V2$) [267]. En la figura 12.1 se puede apreciar la disposición física de las partes de este sistema.

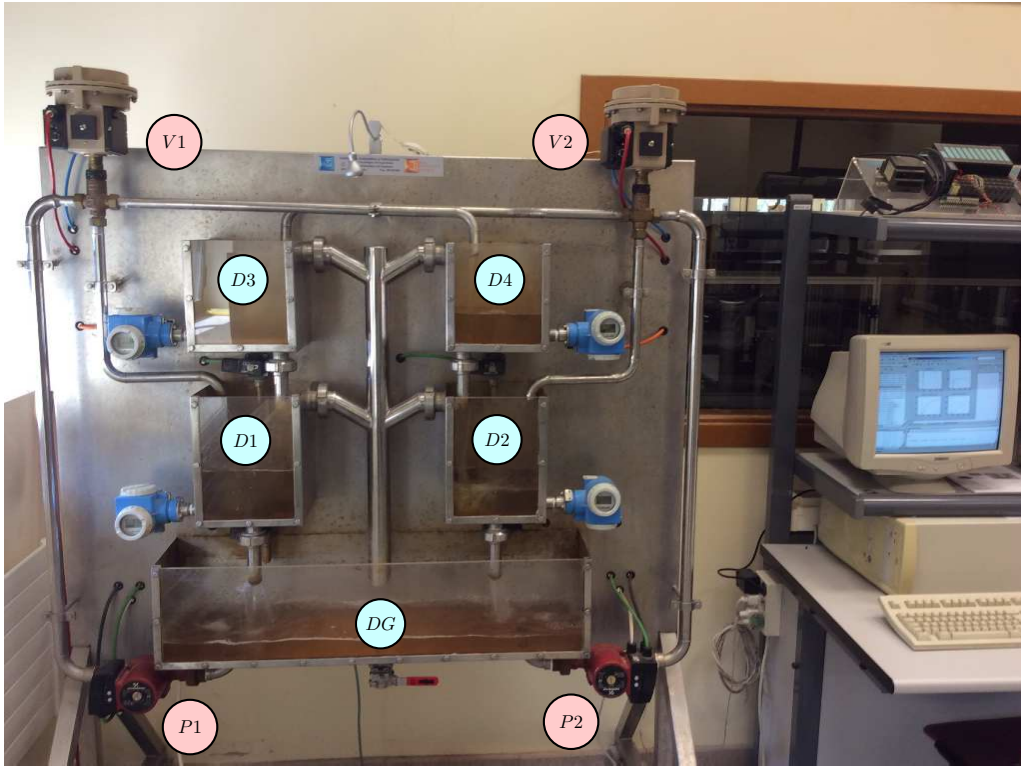


Figura 12.1: Sistema hidráulico MIMO.

En este trabajo se utiliza en el modo de operación denominado 2×2 . La bomba $P1$ suministra flujo a los tanques $D1$ y $D4$, distribuido de forma proporcional mediante la válvula de tres vías $V1$. De esta forma, si el punto de ajuste es igual a 0%, todo el líquido se envía al tanque $D1$; mientras que si el punto de ajuste es igual al 100%, implica que todo el líquido se envía al tanque $D4$. Ambas bombas tienen caudales máximos y mínimos, y la capacidad de los tanques está limitada entre los niveles superior e inferior, oscilando continuamente debido al suministro ininterrumpido de líquido y las tres salidas de fluido de cada tanque: salida inferior, salida superior (establecida como alivio) y salida controlable debido a válvulas solenoides de encendido/apagado [267].

El porcentaje de apertura de la válvula de tres vías $u_2 = s(t)$ y la potencia de la bomba $u_1 = p(t)$ se consideran entradas del sistema, mientras que los niveles de líquido en ambos tanques $y_1(t) = D1$ y $y_2(t) = D4$ son las salidas; todas estas variables oscilan entre 0% y 100%. Como es de apreciar, la planta constituye un sistema MIMO. En la figura 12.2 se puede observar la representación esquemática de esta sistema. Las ecuaciones que describen la dinámica del sistema son las siguientes:

$$\frac{dh_1}{dt} = C_1 q_1(t) - B_1 \sqrt{h_1} \quad (12.1)$$

$$\frac{dh_2}{dt} = C_2 q_2(t) - B_2 \sqrt{h_2} \quad (12.2)$$

$$q_1(t) = u_1(100 - u_2) = p(t)(100 - s(t)) \quad (12.3)$$

$$q_2(t) = u_1 u_2 = p(t) s(t) \quad (12.4)$$

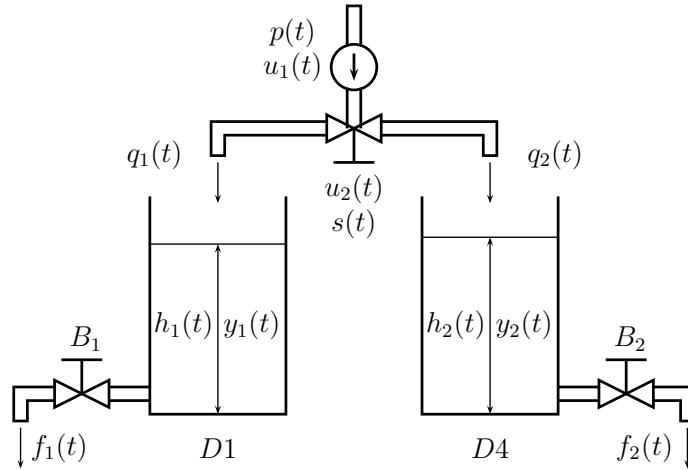


Figura 12.2: Representación del sistema hidráulico MIMO.

En la figura 12.3 se puede observar el esquema en diagrama de bloques del sistema hidráulico MIMO compuesto por dos tanques. En este diagrama se aprecian las no linealidades que presenta la planta.

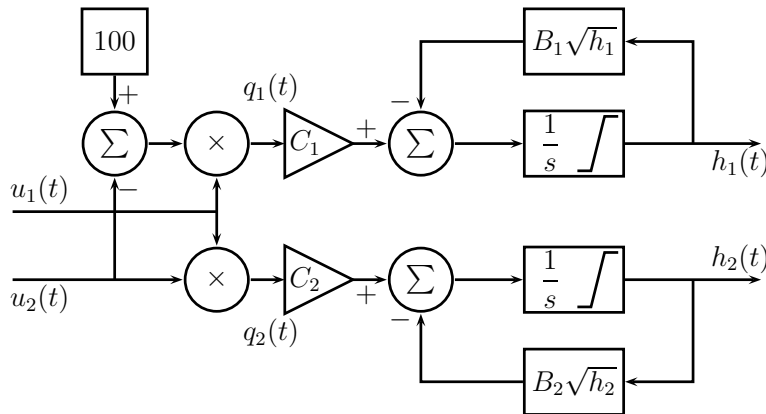


Figura 12.3: Esquema del sistema hidráulico MIMO.

En las figuras 12.4 y 12.5 se muestran ejemplos del comportamiento de la planta en lazo abierto empleando diferentes valores de la bomba y la posición de la válvula de tres vías. En la figura 12.5 se observa que el tiempo de establecimiento de la planta es de 100 segundos.

Utilizando los datos de la figura 12.4 y siguiendo el proceso de optimización de la figura 9.19 tomando $X = [C_1, C_2, B_1, B_2]$ como las variables a optimizar de la función objetivo dada por la ecuación (12.5), donde N_T es el número total de datos, $y_{r,1}$, $y_{r,2}$ los datos reales, con $y_{s,1}$, $y_{s,2}$ los datos simulados, entonces, se establece que $C_1 = 2,95 \times 10^{-4}$, $B_1 = 0,2$, $C_2 = 2,95 \times 10^{-4}$ y $B_2 = 0,2$. Los resultados de simulación se pueden apreciar en la figura 12.6.

$$J_s(X) = \frac{1}{N_T} \sum_{n=1}^{N_T} [(y_{r,1} - y_{s,1})^2 + (y_{r,2} - y_{s,2})^2] \quad (12.5)$$

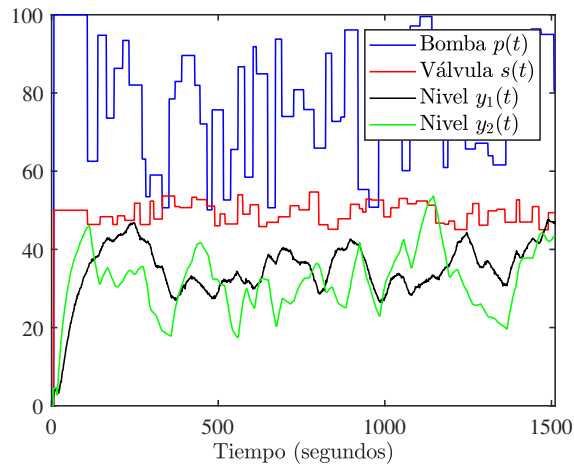


Figura 12.4: Respuesta de la planta en lazo abierto.

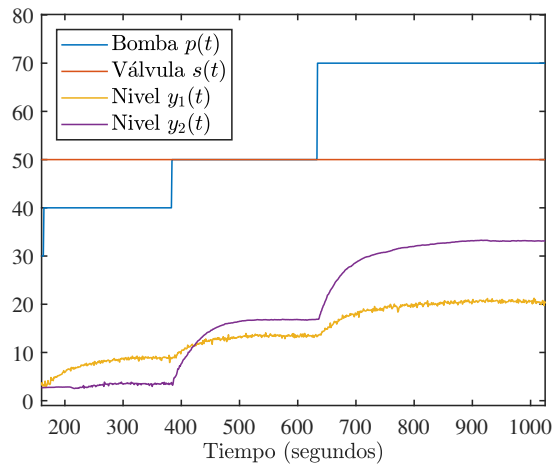


Figura 12.5: Comportamiento de la planta en lazo abierto.

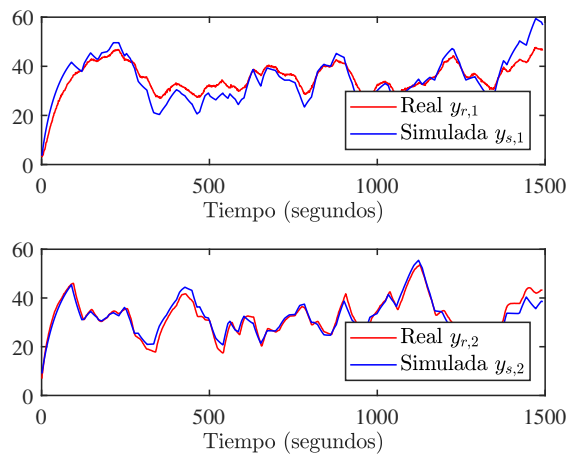


Figura 12.6: Simulación de la planta en lazo abierto.

12.3. Sistema de control neuro-difuso adaptativo MIMO

Al igual que en el caso SISO, la arquitectura considerada utiliza dos sistemas neuro-difusos, uno como controlador y otro para el modelo de la planta. En este esquema, primero se efectúa la identificación de la planta y posteriormente se realiza el entrenamiento del controlador. Sin embargo, se tienen dos entradas que corresponden a los valores de referencia de los tanques, también se tienen dos acciones de control y salidas del sistema. En la figura 12.7 se puede apreciar el esquema del sistema de control neuro-difuso adaptativo MIMO. Además, se tienen dos modelos de referencia correspondientes al comportamiento deseado que se quiere tener del sistema.

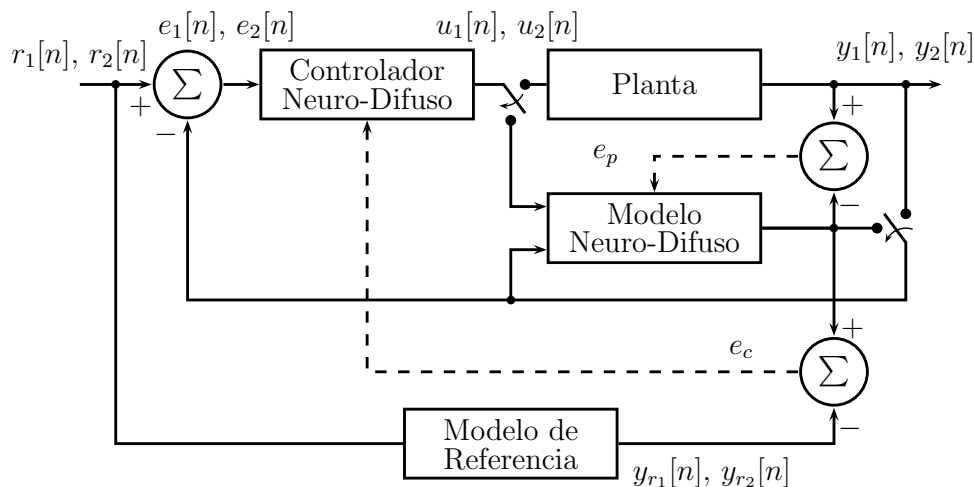


Figura 12.7: Esquema de control mediante sistemas neuro-difusos.

Para la implementación del sistema de control adaptativo, primero se efectúa la identificación de la planta, después, se realiza el entrenamiento del controlador neuro-difuso. Bajo este esquema se integra el modelo de la planta al lazo de control para el entrenamiento del controlador. En el primer paso se establecen las configuraciones iniciales del controlador y del modelo neuro-difuso de la planta, obtenidos de un conocimiento previo. En el segundo paso, se toman datos de entrada-salida de la planta durante el funcionamiento del sistema en lazo cerrado. Posteriormente, con los datos tomados se efectúa una nueva identificación de la planta. Luego, utilizando el modelo actualizado de la planta, se hace el entrenamiento del controlador. Como paso siguiente se pone en funcionamiento el controlador optimizado; de esta forma, se busca corregir la variación presente en el sistema. Si no existe algún requerimiento de parada, este proceso se repite para el siguiente intervalo de tiempo.

Considerando la limitada cantidad de datos producida durante el funcionamiento del controlador luego que se ha optimizado, el proceso de identificación de la planta y entrenamiento del controlador ocurren de forma iterativa. Es de apreciar la importancia de establecer un punto inicial de búsqueda para la identificación de la planta y optimización del controlador, lo cual se logra con los sistemas neuro-difusos establecidos inicialmente.

12.4. Sistema difuso compacto basado en relaciones booleanas

Para el desarrollo de esta aplicación se utiliza la arquitectura II de un sistema neuro-difuso BBR. En este caso, considerando una variable de entrada x_1 ; una salida de activación se puede codificar mediante la tabla 12.1, de esta manera se tiene que $Y_{1,1} = A_{1,1}(x_1)$. De forma general, incrementando las columnas para las salidas de activación se tiene:

$$Y_{1,l} = A_{1,l}(x_1) \quad (12.6)$$

$A_{1,1}(x_1)$	$A_{1,2}(x_1)$	\dots	$A_{1,l}(x_1)$	$Y_{1,1}$
1	X	\dots	X	1
0	X	\dots	X	0

Tabla 12.1: Codificación parcial para una entrada y una función de activación.

Teniendo para cada conjunto $A_{1,l}$ una función de pertenencia $\mu_{1,l}$ y con actuadores virtuales $v_{1,l}$, se tiene una salida parcial f_1 como la mostrada en la figura 12.8.

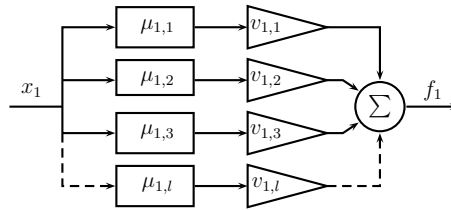


Figura 12.8: Ejemplo de la configuración para una salida parcial.

Extendiendo la tabla 12.1 para más variables de entrada x_1, x_2, \dots, x_j , es posible tener la codificación de la tabla 12.2.

$A_{j,1}(x_j)$	$A_{j,2}(x_j)$	\dots	$A_{j,l}(x_j)$	$Y_{j,1}$	$Y_{j,2}$	\dots	$Y_{j,l}$
1	X	\dots	X	1	0	\dots	0
X	1	\dots	X	0	1	\dots	0
\vdots	\vdots	\ddots	\vdots	\vdots	\vdots	\ddots	\vdots
X	X	\dots	1	0	0	\dots	1

Tabla 12.2: Codificación para el sistema compacto para una entrada x_j .

De esta forma, cada función de activación $Y_{j,l}$ depende directamente de un conjunto $A_{j,l}$ teniendo la expresión:

$$Y_{j,l} = A_{j,l}(x_j) \quad (12.7)$$

Empleando las respectivas funciones de pertenencia $\mu_{j,l}$ asociadas a los conjuntos $A_{j,l}$, se tiene el esquema general mostrado en la figura 12.9.

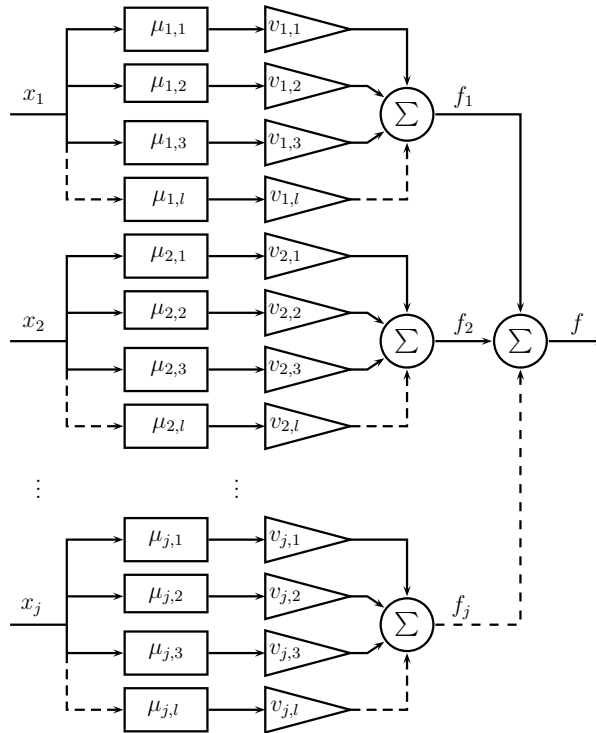


Figura 12.9: Esquema general del sistema compacto.

La salida del proceso de inferencia se puede calcular como:

$$f = \sum_{j=1}^M \sum_{l=1}^N v_{j,l} \mu_{j,l}(x_j) \quad (12.8)$$

La figura 12.10 presenta los conjuntos booleanos considerados (base) y las respectivas funciones de pertenencia utilizadas para implementar los sistemas compactos. De esta forma, se tienen los conjuntos difusos sigmoidales $\mu_{j,1}$ y $\mu_{j,2}$ para modelar los valores negativos y positivos del universo de discurso x_j .

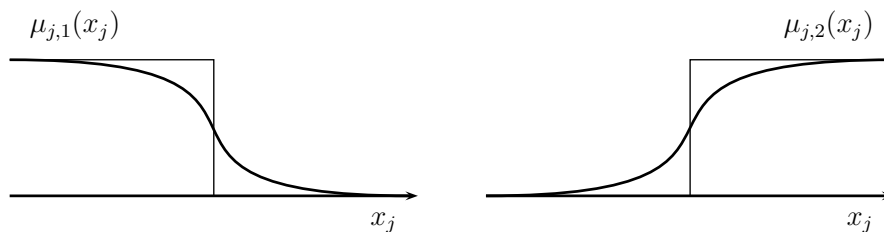


Figura 12.10: Conjuntos booleanos y funciones de pertenencia empleadas.

Empleando los conjuntos difusos de la figura 12.10, se tiene el esquema de la figura 12.11, que muestra el sistema difuso propuesto. Este se considera como un bloque básico para construir los respectivos modelos MIMO para el modelo de la planta y el controlador.

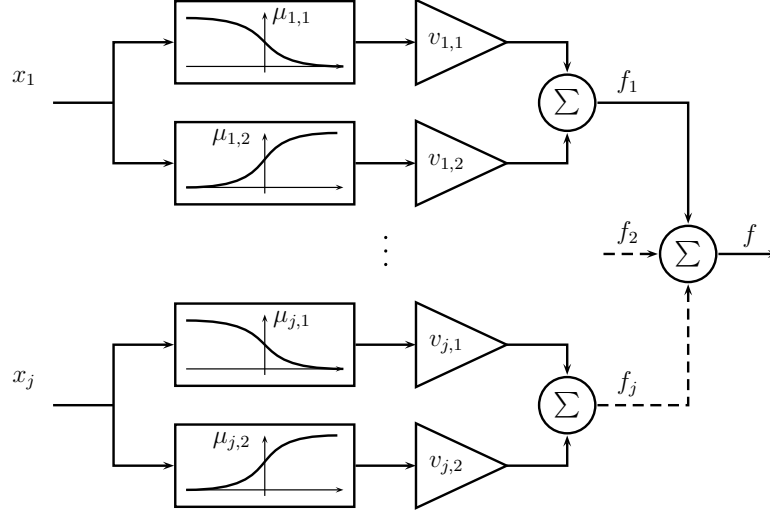


Figura 12.11: Esquema del sistema de control neuro-difuso.

Las funciones de pertenencia de la figura 12.10, se pueden representar por la ecuación (12.9) donde el conjunto de parámetros es $h \in \{v_{j,l}, \sigma_{j,l}, \gamma_{j,l}\}$.

$$\mu_{j,l}(x_j) = (1 + e^{-\sigma_{j,l}(x_j - \gamma_{j,l})})^{-1} \quad (12.9)$$

Finalmente, el proceso de inferencia para este tipo de sistema se puede calcular de la forma:

$$f = \sum_{j=1}^M f_j(x_j) = \sum_{j=1}^M \sum_{l=1}^2 v_{j,l} \mu_{j,l}(x_j) \quad (12.10)$$

12.5. Configuraciones MIMO con sistemas basados en relaciones booleanas

Para la implementación del sistema de control adaptativo MIMO, tanto para la identificación de la planta como para el controlador, se consideran diferentes arreglos de subsistemas BBR de primer orden. Considerando lo descrito en [267], para el control del sistema hidráulico MIMO se utilizan subsistemas BBR de primer orden para la identificación de la planta. Estos subsistemas se establecen considerando la analogía con un sistema lineal de primer orden en tiempo discreto con una función de transferencia de la forma:

$$G(z) = \frac{X(z)}{W(z)} = \frac{b_0}{1 - a_0 z^{-1}} \quad (12.11)$$

La respectiva ecuación en diferencias es:

$$x[n] = b_0 w[n] + a_0 x[n - 1] \quad (12.12)$$

Para el caso del subsistema BBR la anterior ecuación se puede ver como:

$$x[n] = f_1(w[n]) + f_2(x[n - 1]) \quad (12.13)$$

De esta forma, el esquema de un subsistema BBR utilizado para la identificación de la planta se puede apreciar en la figura 12.12.

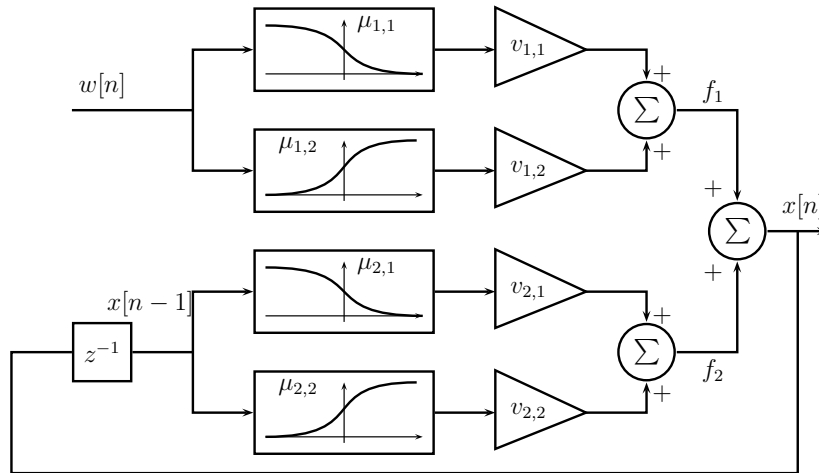


Figura 12.12: Sistema BBR-SISO de primer orden.

Por su parte, para el diseño del controlador se utiliza una configuración compacta BBR como la mostrada en la figura 12.13, la cual permite una analogía con un compensador de cero polo y ganancia que tiene una función de transferencia de la forma:

$$C(z) = \frac{W(z)}{E(z)} = \frac{b_1 + b_0 z^{-1}}{1 - a_0 z^{-1}} \quad (12.14)$$

Que en tiempo discreto se puede describir como:

$$w[n] = b_1 e[n] + b_0 e[n - 1] + a_0 w[n - 1] \quad (12.15)$$

Para el caso del subsistema BBR se puede ver como:

$$w[n] = f_1(e[n]) + f_2(e[n - 1]) + f_3(w[n - 1]) \quad (12.16)$$

Para el sistema MIMO, principalmente se consideran dos posibles configuraciones dependiendo de la relación que se puede tener entre las salidas de los subsistemas BBR. Estas configuraciones se proponen considerando las relaciones presentes en el modelo no lineal de la planta real y corresponden a:

- Suma de las salidas.
- Producto de las salidas.

La figura 12.14 muestra la configuración que suma las salidas de los subsistemas BBR. Por su parte, la figura 12.15 muestra un esquema donde se tiene el producto de las salidas de los subsistemas BBR.

12.5.1. Configuraciones MIMO utilizadas para el sistema de control adaptativo

Para la implementación del controlador se emplea la configuración mostrada en la figura 12.16 (suma de las salidas). Por su parte, para la identificación de la planta se utiliza la combinación mostrada en la figura 12.17, compuesta por la conexión en cascada de las configuraciones de las figuras 12.14 y 12.15. Al utilizar este arreglo se logra una adecuada identificación de la planta para efectuar el proceso de control adaptativo, ya que se tiene una estructura que permite representar las conexiones del diagrama de bloques de la figura 12.3, el cual representa la planta real.

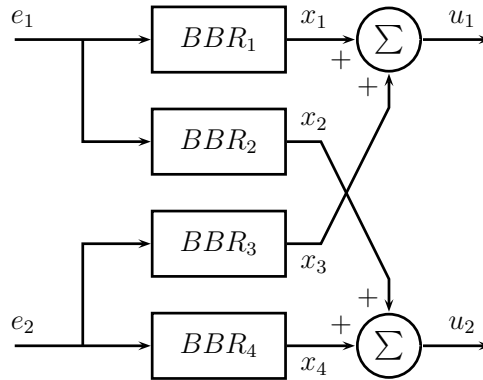


Figura 12.16: Sistema MIMO BBR utilizado para el controlador.

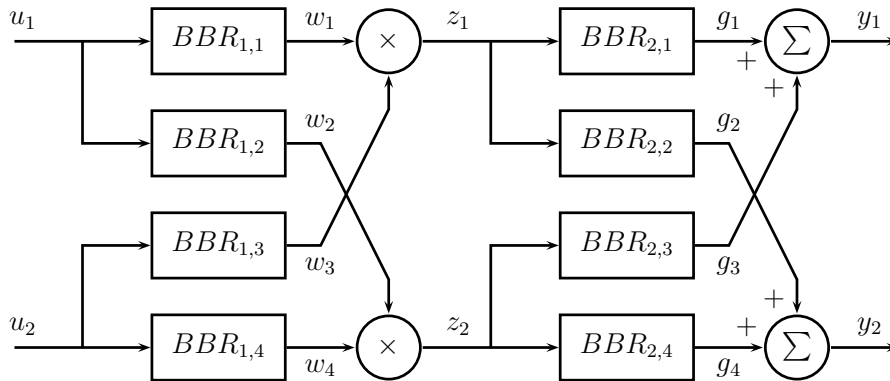


Figura 12.17: Sistema MIMO BBR (compuesto) utilizado para la identificación de la planta.

12.6. Ecuaciones para el entrenamiento del controlador

Esta sección muestra el proceso para establecer las ecuaciones para el entrenamiento del controlador que corresponde al algoritmo *Dynamic Back Propagation* (DBP). Para esto, debe considerarse tanto la arquitectura del controlador MIMO como la utilizada para la identificación de la planta. Integrando los esquemas de las figuras 12.16 y 12.17 se

obtiene el diagrama de la figura 12.18. De esta forma, se tienen las siguientes ecuaciones asociadas a la dinámica presente en el entrenamiento del controlador:

$$\begin{aligned}
 y_1 &= g_1 + g_3 & u_1 &= x_1 + x_3 \\
 y_2 &= g_2 + g_4 & u_2 &= x_2 + x_4 \\
 z_1 &= w_1 \cdot w_3 & e_1 &= r_1 - y_1 \\
 z_2 &= w_2 \cdot w_4 & e_2 &= r_2 - y_2
 \end{aligned}
 \tag{12.17}$$

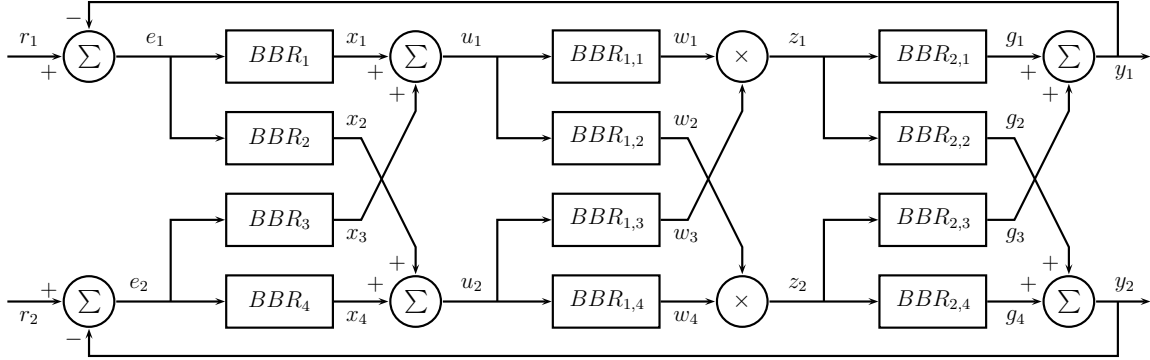


Figura 12.18: Integración de los sistemas MIMO BBR.

Considerando la configuración mostrada en la figura 12.12, las salidas de cada subsistema BBR son:

$$\begin{aligned}
 g_1 &= f_{g,1,1}(z_1) + f_{g,1,2}(g_1) & w_1 &= f_{w,1,1}(u_1) + f_{w,1,2}(w_1) \\
 g_2 &= f_{g,2,1}(z_1) + f_{g,2,2}(g_2) & w_2 &= f_{w,2,1}(u_1) + f_{w,2,2}(w_2) \\
 g_3 &= f_{g,3,1}(z_2) + f_{g,3,2}(g_3) & w_3 &= f_{w,3,1}(u_2) + f_{w,3,2}(w_3) \\
 g_4 &= f_{g,4,1}(z_2) + f_{g,4,2}(g_4) & w_4 &= f_{w,4,1}(u_2) + f_{w,4,2}(w_4)
 \end{aligned}
 \tag{12.18}$$

$$\begin{aligned}
 x_1[n+1] &= f_{x,1,1}(e_1[n+1]) + f_{x,1,2}(e_1[n]) + f_{x,1,3}(x_1[n]) \\
 x_2[n+1] &= f_{x,2,1}(e_1[n+1]) + f_{x,2,2}(e_1[n]) + f_{x,2,3}(x_2[n]) \\
 x_3[n+1] &= f_{x,3,1}(e_2[n+1]) + f_{x,3,2}(e_2[n]) + f_{x,3,3}(x_3[n]) \\
 x_4[n+1] &= f_{x,4,1}(e_2[n+1]) + f_{x,4,2}(e_2[n]) + f_{x,4,3}(x_4[n])
 \end{aligned}
 \tag{12.19}$$

Tomando los subíndices para el bloque $s \in \{g, w, x\}$, para el subsistema $i = 1, 2, 3, 4$, para la conexión interna $j = 1, 2$ y para la salida parcial $l = 1, 2$, entonces, una función $f_{s,i,j}$ corresponde a:

$$f_{s,i,j}(\chi) = \sum_{l=1}^2 f_{s,i,j,l}(\chi) = \sum_{l=1}^2 v_{sijl} (1 + e^{-\sigma_{sijl}(\chi - \gamma_{sijl})})^{-1}
 \tag{12.20}$$

donde:

$$f_{s,i,j,l}(\chi) = v_{sijl} (1 + e^{-\sigma_{sijl}(\chi - \gamma_{sijl})})^{-1}
 \tag{12.21}$$

es decir:

$$f_{s,i,j}(\chi) = v_{sij1} \left(1 + e^{-\sigma_{sij1}(\chi - \gamma_{sij1})}\right)^{-1} + v_{sij2} \left(1 + e^{-\sigma_{sij2}(\chi - \gamma_{sij2})}\right)^{-1} \quad (12.22)$$

La adaptación de los parámetros se hace de la forma:

$$h_c(k+1) = h_c(k) - \eta \frac{dJ_c}{dh_c} \quad (12.23)$$

Donde η es la tasa de aprendizaje y J_c es la función de ajuste definida como:

$$J_c = \frac{1}{2} [(y_{r1} - y_1)^2 + (y_{r2} - y_2)^2] \quad (12.24)$$

En esta ecuación, y_{r1} y y_{r2} son las salidas deseadas (obtenidas del modelo de referencia) y y_1 , y_2 las salidas obtenidas del controlador con el modelo neuro-difuso de la planta.

Considerando un parámetro de ajuste h_c , para el entrenamiento del controlador la variación de J_c en función de los parámetros de ajuste es:

$$\frac{dJ_c}{dh_c} = -e_1 \frac{dy_1}{dh_c} - e_2 \frac{dy_2}{dh_c} \quad (12.25)$$

Derivadas para la planta

En esta parte se establecen las respectivas derivadas de los componentes de la planta con respecto a los parámetros del controlador. De esta forma, para el módulo correspondiente a la salida de la planta se tiene:

$$\begin{aligned} \frac{dy_1}{dh_c} &= \frac{dg_1}{dh_c} + \frac{dg_3}{dh_c} \\ \frac{dy_2}{dh_c} &= \frac{dg_2}{dh_c} + \frac{dg_4}{dh_c} \end{aligned} \quad (12.26)$$

Puesto que los parámetros de la planta son diferentes a los empleados en el controlador, la función $f_{g,i,j}$ no depende directamente de h_c , pero las variables z_i y g_i dependen de forma implícita de h_c , entonces, las respectivas derivadas son:

$$\begin{aligned} \frac{dg_1}{dh_c}[n+1] &= \frac{df_{g,1,1}}{dz_1} \frac{dz_1}{dh_c}[n] + \frac{df_{g,1,2}}{dg_1} \frac{dg_1}{dh_c}[n] \\ \frac{dg_2}{dh_c}[n+1] &= \frac{df_{g,2,1}}{dz_1} \frac{dz_1}{dh_c}[n] + \frac{df_{g,2,2}}{dg_2} \frac{dg_2}{dh_c}[n] \\ \frac{dg_3}{dh_c}[n+1] &= \frac{df_{g,3,1}}{dz_2} \frac{dz_2}{dh_c}[n] + \frac{df_{g,3,2}}{dg_3} \frac{dg_3}{dh_c}[n] \\ \frac{dg_4}{dh_c}[n+1] &= \frac{df_{g,4,1}}{dz_2} \frac{dz_2}{dh_c}[n] + \frac{df_{g,4,2}}{dg_4} \frac{dg_4}{dh_c}[n] \end{aligned} \quad (12.27)$$

Donde se tiene que:

$$\frac{df_{g,i,j}}{d\chi} = \sum_{l=1}^2 v_{gijl} \left(1 + e^{-\sigma_{gijl}(\chi - \gamma_{gijl})}\right)^{-2} e^{-\sigma_{gijl}(\chi - \gamma_{gijl})} \sigma_{gijl} \quad (12.28)$$

Por su parte, para el módulo correspondiente a la entrada de la planta:

$$\begin{aligned}\frac{dz_1}{dh_c} &= w_3 \frac{dw_1}{dh_c} + w_1 \frac{dw_3}{dh_c} \\ \frac{dz_2}{dh_c} &= w_4 \frac{dw_2}{dh_c} + w_2 \frac{dw_4}{dh_c}\end{aligned}\tag{12.29}$$

De la misma manera, la función $f_{w,i,j}$ no depende directamente de h_c , pero las variables u_i y w_i dependen de forma implícita de h_c , por lo tanto:

$$\begin{aligned}\frac{dw_1}{dh_c}[n+1] &= \frac{df_{w,1,1}}{du_1} \frac{du_1}{dh_c}[n] + \frac{df_{w,1,2}}{dw_1} \frac{dw_1}{dh_c}[n] \\ \frac{dw_2}{dh_c}[n+1] &= \frac{df_{w,2,1}}{du_1} \frac{du_1}{dh_c}[n] + \frac{df_{w,2,2}}{dw_2} \frac{dw_2}{dh_c}[n] \\ \frac{dw_3}{dh_c}[n+1] &= \frac{df_{w,3,1}}{du_2} \frac{du_2}{dh_c}[n] + \frac{df_{w,3,2}}{dw_3} \frac{dw_3}{dh_c}[n] \\ \frac{dw_4}{dh_c}[n+1] &= \frac{df_{w,4,1}}{du_2} \frac{du_2}{dh_c}[n] + \frac{df_{w,4,2}}{dw_4} \frac{dw_4}{dh_c}[n]\end{aligned}\tag{12.30}$$

De esta forma las respectivas derivadas son:

$$\frac{df_{w,i,j}}{d\chi} = \sum_{l=1}^2 v_{wijkl} (1 + e^{-\sigma_{wijkl}(\chi - \gamma_{wijkl})})^{-2} e^{-\sigma_{wijkl}(\chi - \gamma_{wijkl})} \sigma_{wijkl}\tag{12.31}$$

Derivadas para el controlador

En esta parte, se calculan las derivadas de los componentes del controlador respecto a sus parámetros. De esta forma:

$$\begin{aligned}\frac{du_1}{dh_c} &= \frac{dx_1}{dh_c} + \frac{dx_3}{dh_c} \\ \frac{du_2}{dh_c} &= \frac{dx_2}{dh_c} + \frac{dx_4}{dh_c}\end{aligned}\tag{12.32}$$

En este caso la función $f_{x,i,j}$ depende directamente de h_c y de forma implícita de e_i y x_i , por lo tanto, las correspondientes derivadas son:

$$\begin{aligned}\frac{dx_1}{dh_c}[n+1] &= \frac{df_{x,1,1}(e_1)}{dh_c}[n+1] + \frac{df_{x,1,2}(e_1)}{dh_c}[n] + \frac{df_{x,1,3}(x_1)}{dh_c}[n] \\ \frac{dx_2}{dh_c}[n+1] &= \frac{df_{x,2,1}(e_1)}{dh_c}[n+1] + \frac{df_{x,2,2}(e_1)}{dh_c}[n] + \frac{df_{x,2,3}(x_2)}{dh_c}[n] \\ \frac{dx_3}{dh_c}[n+1] &= \frac{df_{x,3,1}(e_2)}{dh_c}[n+1] + \frac{df_{x,3,2}(e_2)}{dh_c}[n] + \frac{df_{x,3,3}(x_3)}{dh_c}[n] \\ \frac{dx_4}{dh_c}[n+1] &= \frac{df_{x,4,1}(e_2)}{dh_c}[n+1] + \frac{df_{x,4,2}(e_2)}{dh_c}[n] + \frac{df_{x,4,3}(x_4)}{dh_c}[n]\end{aligned}\tag{12.33}$$

Con la finalidad de establecer las respectivas derivadas se debe tener presente que el conjunto de parámetros del controlador es $h_c = h_{xmn p} \in \{v_{xmn p}, \sigma_{xmn p}, \gamma_{xmn p}\}$. Dependiendo de la variable respecto a la cual se realiza el cálculo de la derivada se tienen diferentes casos, por lo cual, en primer lugar, si $m \neq i$ y $n \neq j$ entonces:

$$\frac{df_{xij}}{dh_{xmn p}} = \frac{df_{xij}}{d\chi} \frac{d\chi}{dh_{xmn p}} \quad (12.34)$$

Donde:

$$\frac{df_{xij}}{d\chi} = \sum_{l=1}^2 v_{xijl} (1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})})^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \quad (12.35)$$

En segundo lugar, cuando $m = i$ y $n = j$, para el parámetro $h_{xij p} = v_{xij p}$ se tiene:

$$\frac{df_{xij}}{dv_{xij p}} = \frac{df_{xij1}}{dv_{xij p}} + \frac{df_{xij2}}{dv_{xij p}} \quad (12.36)$$

En el caso donde $l = p$, la respectiva derivada se puede escribir como:

$$\begin{aligned} \frac{df_{xij p}}{dv_{xij p}} &= (1 + e^{-\sigma_{xij p}(\chi - \gamma_{xij p})})^{-1} - \\ &v_{xij p} (1 + e^{-\sigma_{xij p}(\chi - \gamma_{xij p})})^{-2} e^{-\sigma_{xij p}(\chi - \gamma_{xij p})} (-\sigma_{xij p}) \frac{d\chi}{dv_{xij p}} \end{aligned} \quad (12.37)$$

$$\begin{aligned} \frac{df_{xij p}}{dv_{xij p}} &= \left[(1 + e^{-\sigma_{xij p}(\chi - \gamma_{xij p})})^{-1} \right] + \\ &\left[v_{xij p} (1 + e^{-\sigma_{xij p}(\chi - \gamma_{xij p})})^{-2} e^{-\sigma_{xij p}(\chi - \gamma_{xij p})} \sigma_{xij p} \right] \frac{d\chi}{dv_{xij p}} \end{aligned} \quad (12.38)$$

$$\frac{df_{xij p}}{dv_{xij p}} = F v_{xij p} + K v_{xij p} \frac{d\chi}{dv_{xij p}} \quad (12.39)$$

donde:

$$F v_{xij p} = (1 + e^{-\sigma_{xij p}(\chi - \gamma_{xij p})})^{-1} \quad (12.40)$$

$$K v_{xij p} = v_{xij p} (1 + e^{-\sigma_{xij p}(\chi - \gamma_{xij p})})^{-2} e^{-\sigma_{xij p}(\chi - \gamma_{xij p})} \sigma_{xij p} \quad (12.41)$$

Por su parte, cuando $l \neq p$ entonces:

$$\frac{df_{xijl}}{dv_{xij p}} = v_{xijl} (1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})})^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \frac{d\chi}{dv_{xij p}} \quad (12.42)$$

$$\frac{df_{xijl}}{dv_{xij p}} = R v_{xijl} \frac{d\chi}{dv_{xij p}} \quad (12.43)$$

donde:

$$R v_{xijl} = v_{xijl} (1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})})^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \quad (12.44)$$

En términos generales considerando $l \neq p$ se tiene que:

$$\frac{df_{xij}}{dv_{xijp}} = Fv_{xijp} + (Rv_{xijl} + Kv_{xijp}) \frac{d\chi}{dv_{xijp}} \quad (12.45)$$

De la misma forma, ahora para el parámetro $h_{xijp} = \sigma_{xijp}$ se establece:

$$\frac{df_{xij}}{d\sigma_{xijp}} = \frac{df_{xij1}}{d\sigma_{xijp}} + \frac{df_{xij2}}{d\sigma_{xijp}} \quad (12.46)$$

En el caso donde $l = p$ se determina:

$$\frac{df_{xijp}}{d\sigma_{xijp}} = -v_{xijp} (1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})})^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \left(\gamma_{xijp} - \chi - \sigma_{xijp} \frac{d\chi}{d\sigma_{xijp}} \right) \quad (12.47)$$

$$\begin{aligned} \frac{df_{xijp}}{d\sigma_{xijp}} = & \left[v_{xijp} (1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})})^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} (\chi - \gamma_{xijp}) \right] \\ & + \left[v_{xijp} (1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})})^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \sigma_{xijp} \right] \frac{d\chi}{d\sigma_{xijp}} \end{aligned} \quad (12.48)$$

$$\frac{df_{xijp}}{d\sigma_{xijp}} = F\sigma_{xijp} + K\sigma_{xijp} \frac{d\chi}{d\sigma_{xijp}} \quad (12.49)$$

donde:

$$F\sigma_{xijp} = v_{xijp} (1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})})^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} (\chi - \gamma_{xijp}) \quad (12.50)$$

$$K\sigma_{xijp} = v_{xijp} (1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})})^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \sigma_{xijp} \quad (12.51)$$

Por su parte si $l \neq p$, entonces:

$$\frac{df_{xijl}}{d\sigma_{xijp}} = v_{xijl} (1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})})^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \frac{d\chi}{d\sigma_{xijp}} \quad (12.52)$$

$$\frac{df_{xijl}}{d\sigma_{xijp}} = R\sigma_{xijl} \frac{d\chi}{d\sigma_{xijp}} \quad (12.53)$$

donde:

$$R\sigma_{xijl} = v_{xijl} (1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})})^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \quad (12.54)$$

De forma general, considerando $l \neq p$ se tiene que:

$$\frac{df_{xij}}{d\sigma_{xijp}} = F\sigma_{xijp} + (R\sigma_{xijl} + K\sigma_{xijp}) \frac{d\chi}{d\sigma_{xijp}} \quad (12.55)$$

Finalmente para el parámetro $h_{xijp} = \gamma_{xijp}$ se establece:

$$\frac{df_{xij}}{d\gamma_{xijp}} = \frac{df_{xij1}}{d\gamma_{xijp}} + \frac{df_{xij2}}{d\gamma_{xijp}} \quad (12.56)$$

Considerando el caso donde $l = p$ se determina:

$$\frac{df_{xijp}}{d\gamma_{xijp}} = -v_{xijp} \left(1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})}\right)^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \left(\sigma_{xijp} - \sigma_{xijp} \frac{d\chi}{d\gamma_{xijp}}\right) \quad (12.57)$$

$$\begin{aligned} \frac{df_{xijp}}{d\gamma_{xijp}} = & \left[-v_{xijp} \left(1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})}\right)^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \sigma_{xijp} \right] \\ & + \left[v_{xijp} \left(1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})}\right)^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \sigma_{xijp} \right] \frac{d\chi}{d\gamma_{xijp}} \end{aligned} \quad (12.58)$$

$$\frac{df_{xijp}}{d\gamma_{xijp}} = F\gamma_{xijp} + K\gamma_{xijp} \frac{d\chi}{d\gamma_{xijp}} \quad (12.59)$$

donde:

$$F\gamma_{xijp} = -v_{xijp} \left(1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})}\right)^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \sigma_{xijp} \quad (12.60)$$

$$K\gamma_{xijp} = v_{xijp} \left(1 + e^{-\sigma_{xijp}(\chi - \gamma_{xijp})}\right)^{-2} e^{-\sigma_{xijp}(\chi - \gamma_{xijp})} \sigma_{xijp} \quad (12.61)$$

Por su parte si $l \neq p$ entonces:

$$\frac{df_{xijl}}{d\gamma_{xijp}} = v_{xijl} \left(1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})}\right)^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \frac{d\chi}{d\gamma_{xijp}} \quad (12.62)$$

$$\frac{df_{xijl}}{d\gamma_{xijp}} = R\gamma_{xijl} \frac{d\chi}{d\gamma_{xijp}} \quad (12.63)$$

donde:

$$R\gamma_{xijl} = v_{xijl} \left(1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})}\right)^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \quad (12.64)$$

De forma general considerando $l \neq p$ se tiene que:

$$\frac{df_{xij}}{d\gamma_{xijp}} = F\gamma_{xijp} + (R\gamma_{xijl} + K\gamma_{xijp}) \frac{d\chi}{d\gamma_{xijp}} \quad (12.65)$$

Finalmente las ecuaciones para la parte del controlador son:

$$\begin{aligned} \frac{dx_1}{dh_c}[n+1] = & Th_{x11} \frac{de_1}{dh_c}[n+1] + Th_{x12} \frac{de_1}{dh_c}[n] + Th_{x13} \frac{dx_1}{dh_c}[n] + \\ & Fh_{x11}(e_1[n+1]) + Fh_{x12}(e_1[n]) + Fh_{x13}(x_1[n]) \end{aligned} \quad (12.66)$$

$$\begin{aligned} \frac{dx_2}{dh_c}[n+1] = & Th_{x21} \frac{de_1}{dh_c}[n+1] + Th_{x22} \frac{de_1}{dh_c}[n] + Th_{x23} \frac{dx_2}{dh_c}[n] + \\ & Fh_{x21}(e_1[n+1]) + Fh_{x22}(e_1[n]) + Fh_{x23}(x_2[n]) \end{aligned} \quad (12.67)$$

$$\begin{aligned} \frac{dx_3}{dh_c}[n+1] = & Th_{x31} \frac{de_2}{dh_c}[n+1] + Th_{x32} \frac{de_2}{dh_c}[n] + Th_{x33} \frac{dx_3}{dh_c}[n] + \\ & Fh_{x31}(e_2[n+1]) + Fh_{x32}(e_2[n]) + Fh_{x33}(x_3[n]) \end{aligned} \quad (12.68)$$

$$\frac{dx_4}{dh_c}[n+1] = Th_{x41} \frac{de_2}{dh_c}[n+1] + Th_{x42} \frac{de_2}{dh_c}[n] + Th_{x43} \frac{dx_4}{dh_c}[n] + Fh_{x41}(e_2[n+1]) + Fh_{x42}(e_2[n]) + Fh_{x43}(x_4[n]) \quad (12.69)$$

Donde:

$$Th_{xij}(\chi) = \sum_{l=1}^2 v_{xijl} (1 + e^{-\sigma_{xijl}(\chi - \gamma_{xijl})})^{-2} e^{-\sigma_{xijl}(\chi - \gamma_{xijl})} \sigma_{xijl} \quad (12.70)$$

También es de señalar que $Fh_{xij} \in \{F\gamma_{xijp}, F\sigma_{xijp}, Fv_{xijp}\}$.

Considerando las anteriores ecuaciones los pasos del algoritmo utilizado para el entrenamiento del controlador neuro-difuso son los siguientes:

1. Establecer el modelo de la planta (neuro-difuso) y escoger la configuración inicial de parámetros del controlador.
2. Calcular la respuesta de los modelos de referencia obteniendo (y_{r1}, y_{r2}) .
3. En la k -ésima etapa de entrenamiento, $k = 0, 1, 2, \dots, K_T$, para el valor actual de n se calcula la salida del sistema de control (y_1, y_2) utilizando el modelo neuro-difuso de la planta.
4. Empleando los valores de las referencias (r_1, r_2) , las salidas (y_1, y_2) y los obtenidos para el modelo de referencia (y_{r1}, y_{r2}) se realiza el ajuste de los parámetros del controlador neuro-difuso utilizando las ecuaciones (12.23), (12.25) y subsecuentes. Es importante señalar que los parámetros que se ajustan se almacenan en variables auxiliares ya que durante este paso el controlador no utiliza estos valores.
5. Regresar a la etapa 3 para el siguiente paso de simulación $n = n + 1$ (donde se calcula la salida del sistema de control) hasta que n sea igual al valor definido N_T (tiempo de simulación).
6. En el caso de completar el tiempo de simulación, se actualizan los nuevos parámetros optimizados del controlador y se regresa al paso 3 de evaluación del sistema de control para una nueva iteración $k = k + 1$, hasta que la función de ajuste $J_c(k) = \sum_{n=1}^{N_T} J_c(k, n)$ sea menor que un ε definido, o hasta que k sea igual a un número definido K_T .

12.7. Ecuaciones de entrenamiento para la identificación de la planta

Una aproximación para el modelo del sistema consiste en estimar una estructura que permita desempeñar la misma función de la planta. Para la identificación de la planta con el sistema neuro-difuso se toman muestras de la entrada y salida de tal forma que la respuesta del sistema neuro-difuso es una función no lineal de estas señales. Con el fin de realizar la adaptación o entrenamiento de los parámetros del sistema neuro-difuso se lleva a cabo la actualización mediante la ecuación:

$$h_p(k+1) = h_p(k) - \eta \frac{dJ_p}{dh_p} \quad (12.71)$$

donde η es la tasa de aprendizaje y J_p corresponde a la función de ajuste definida como:

$$J_p = \frac{1}{2}(y_{d1} - y_1)^2 + \frac{1}{2}(y_{d2} - y_2)^2 \quad (12.72)$$

En esta ecuación y_{d1} y y_{d2} son los datos medidos de la planta y y_1 , y_2 los datos obtenidos del sistema neuro-difuso. Considerando un parámetro de ajuste h_p para implementar la ecuación (12.71), la variación de J_p en función de los parámetros de ajuste es:

$$\frac{dJ_p}{dh_p} = -(y_{d1} - y_1)\frac{dy_1}{dh_p} - (y_{d2} - y_2)\frac{dy_2}{dh_p} \quad (12.73)$$

12.7.1. Ecuaciones de entrenamiento para el módulo de salida

Para el módulo correspondiente a la salida de la planta donde los parámetros de esta parte son h_g se tiene:

$$\begin{aligned} \frac{dy_1}{dh_g} &= \frac{dg_1}{dh_g} + \frac{dg_3}{dh_g} \\ \frac{dy_2}{dh_g} &= \frac{dg_2}{dh_g} + \frac{dg_4}{dh_g} \end{aligned} \quad (12.74)$$

$$\begin{aligned} \frac{dg_1}{dh_g}[n+1] &= \frac{df_{g,1,1}(z_1)}{dh_g}[n] + \frac{df_{g,1,2}(g_1)}{dh_g}[n] \\ \frac{dg_2}{dh_g}[n+1] &= \frac{df_{g,2,1}(z_1)}{dh_g}[n] + \frac{df_{g,2,2}(g_2)}{dh_g}[n] \\ \frac{dg_3}{dh_g}[n+1] &= \frac{df_{g,3,1}(z_2)}{dh_g}[n] + \frac{df_{g,3,2}(g_3)}{dh_g}[n] \\ \frac{dg_4}{dh_g}[n+1] &= \frac{df_{g,4,1}(z_2)}{dh_g}[n] + \frac{df_{g,4,2}(g_4)}{dh_g}[n] \end{aligned} \quad (12.75)$$

Para la función f_{gij} considerando el caso donde $i \neq m$ y $j \neq n$, esta función no depende de h_{gmp} , por lo tanto:

$$\frac{df_{gij}}{dh_{gmp}} = 0 \quad (12.76)$$

Por su parte, en el caso que $i = m$ y $j = n$ se tiene que:

$$\frac{df_{gij}}{dh_{gijp}} = \frac{df_{gij1}}{dh_{gijp}} + \frac{df_{gij2}}{dh_{gijp}} \quad (12.77)$$

Para el respectivo cálculo se debe tener presente que las entradas z_1 y z_2 no dependen de h_g , es decir, que f_{gi1p} solamente depende de h_{gi1p} , por lo tanto, respectivamente para

cada parámetro:

$$\frac{df_{gi1p}}{dv_{gi1p}} = (1 + e^{-\sigma_{gi1p}(\chi - \gamma_{gi1p})})^{-1} \quad (12.78)$$

$$\frac{df_{gi1p}}{d\sigma_{gi1p}} = v_{gi1p} (1 + e^{-\sigma_{gi1p}(\chi - \gamma_{gi1p})})^{-2} e^{-\sigma_{gi1p}(\chi - \gamma_{gi1p})} (\chi - \gamma_{gi1p}) \quad (12.79)$$

$$\frac{df_{gi1p}}{d\gamma_{gi1p}} = v_{gi1p} (1 + e^{-\sigma_{gi1p}(\chi - \gamma_{gi1p})})^{-2} e^{-\sigma_{gi1p}(\chi - \gamma_{gi1p})} \sigma_{gi1p} \quad (12.80)$$

En el caso que $p \neq k$, se tiene:

$$\frac{df_{gi1k}}{dh_{gi1p}} = 0 \quad (12.81)$$

Considerando las señales internas g_1, \dots, g_4 , la función f_{gi2p} depende directamente como también implícitamente de h_{gi2p} , entonces:

$$\begin{aligned} \frac{df_{gi2p}}{dv_{gi2p}} &= (1 + e^{-\sigma_{gi2p}(\chi - \gamma_{gi2p})})^{-1} - \\ &v_{gi2p} (1 + e^{-\sigma_{gi2p}(\chi - \gamma_{gi2p})})^{-2} e^{-\sigma_{gi2p}(\chi - \gamma_{gi2p})} (-\sigma_{gi2p}) \frac{d\chi}{dv_{gi2p}} \end{aligned} \quad (12.82)$$

$$\frac{df_{gi2p}}{d\sigma_{gi2p}} = -v_{gi2p} (1 + e^{-\sigma_{gi2p}(\chi - \gamma_{gi2p})})^{-2} e^{-\sigma_{gi2p}(\chi - \gamma_{gi2p})} \left(\gamma_{gi2p} - \chi - \sigma_{gi2p} \frac{d\chi}{d\sigma_{gi2p}} \right) \quad (12.83)$$

$$\frac{df_{gi2p}}{d\gamma_{gi2p}} = -v_{gi2p} (1 + e^{-\sigma_{gi2p}(\chi - \gamma_{gi2p})})^{-2} e^{-\sigma_{gi2p}(\chi - \gamma_{gi2p})} \left(\sigma_{gi2p} - \sigma_{gi2p} \frac{d\chi}{d\gamma_{gi2p}} \right) \quad (12.84)$$

Adicionalmente, si $p \neq k$, entonces, f_{gi2k} depende implícitamente de h_{gi2p} por lo cual:

$$\frac{df_{gi2k}}{dh_{gi2p}} = (1 + e^{-\sigma_{gi2k}(\chi - \gamma_{gi2k})})^{-2} e^{-\sigma_{gi2k}(\chi - \gamma_{gi2k})} \sigma_{gi2k} \frac{d\chi}{dh_{gi2p}} \quad (12.85)$$

De esta forma en términos generales se tiene que:

$$\begin{aligned} \frac{dg_1}{dh_g} [n+1] &= \frac{df_{g,1,1}}{dh_g} + \frac{df_{g,1,2}}{dg_1} \frac{dg_1}{dh_g} [n] + \frac{df_{g,1,2}}{dh_g} \\ \frac{dg_2}{dh_g} [n+1] &= \frac{df_{g,2,1}}{dh_g} + \frac{df_{g,2,2}}{dg_2} \frac{dg_2}{dh_g} [n] + \frac{df_{g,2,2}}{dh_g} \\ \frac{dg_3}{dh_g} [n+1] &= \frac{df_{g,3,1}}{dh_g} + \frac{df_{g,3,2}}{dg_3} \frac{dg_3}{dh_g} [n] + \frac{df_{g,3,2}}{dh_g} \\ \frac{dg_4}{dh_g} [n+1] &= \frac{df_{g,4,1}}{dh_g} + \frac{df_{g,4,2}}{dg_4} \frac{dg_4}{dh_g} [n] + \frac{df_{g,4,2}}{dh_g} \end{aligned} \quad (12.86)$$

Donde las derivadas $df_{g,i,j}/dh_g$ corresponden a:

$$\frac{df_{gijp}}{dv_{gijp}} = (1 + e^{-\sigma_{gijp}(\chi - \gamma_{gijp})})^{-1} \quad (12.87)$$

$$\frac{df_{gijp}}{d\sigma_{gijp}} = v_{gijp} (1 + e^{-\sigma_{gijp}(\chi - \gamma_{gijp})})^{-2} e^{-\sigma_{gijp}(\chi - \gamma_{gijp})} (\chi - \gamma_{gijp}) \quad (12.88)$$

$$\frac{df_{gijp}}{d\gamma_{gijp}} = v_{gijp} (1 + e^{-\sigma_{gijp}(\chi - \gamma_{gijp})})^{-2} e^{-\sigma_{gijp}(\chi - \gamma_{gijp})} \sigma_{gijp} \quad (12.89)$$

De la misma manera, para las derivadas $df_{g,i,2}/dg_i$ se establece:

$$\frac{df_{g,i,2}}{d\chi} = \sum_{l=1}^2 v_{gi2l} (1 + e^{-\sigma_{gi2l}(\chi - \gamma_{gi2l})})^{-2} e^{-\sigma_{gi2l}(\chi - \gamma_{gi2l})} \sigma_{gi2l} \quad (12.90)$$

12.7.2. Ecuaciones de entrenamiento para el módulo de entrada

Para el módulo correspondiente a la entrada de la planta teniendo los parámetros de esta parte como h_w , entonces se tiene:

$$\begin{aligned} \frac{dy_1}{dh_w} &= \frac{dg_1}{dh_w} + \frac{dg_3}{dh_w} \\ \frac{dy_2}{dh_w} &= \frac{dg_2}{dh_w} + \frac{dg_4}{dh_w} \end{aligned} \quad (12.91)$$

$$\begin{aligned} \frac{dg_1}{dh_w}[n+1] &= \frac{df_{g,1,1}(z_1)}{dh_w}[n] + \frac{df_{g,1,2}(g_1)}{dh_w}[n] \\ \frac{dg_2}{dh_w}[n+1] &= \frac{df_{g,2,1}(z_1)}{dh_w}[n] + \frac{df_{g,2,2}(g_2)}{dh_w}[n] \\ \frac{dg_3}{dh_w}[n+1] &= \frac{df_{g,3,1}(z_2)}{dh_w}[n] + \frac{df_{g,3,2}(g_3)}{dh_w}[n] \\ \frac{dg_4}{dh_w}[n+1] &= \frac{df_{g,4,1}(z_2)}{dh_w}[n] + \frac{df_{g,4,2}(g_4)}{dh_w}[n] \end{aligned} \quad (12.92)$$

$$\begin{aligned} \frac{dz_1}{dh_w} &= w_3 \frac{dw_1}{dh_w} + w_1 \frac{dw_3}{dh_w} \\ \frac{dz_2}{dh_w} &= w_4 \frac{dw_2}{dh_w} + w_2 \frac{dw_4}{dh_w} \end{aligned} \quad (12.93)$$

$$\begin{aligned} \frac{dw_1}{dh_w}[n+1] &= \frac{df_{w,1,1}(u_1)}{dh_w}[n] + \frac{df_{w,1,2}(x_1)}{dh_p}[n] \\ \frac{dw_2}{dh_w}[n+1] &= \frac{df_{w,2,1}(u_1)}{dh_w}[n] + \frac{df_{w,3,2}(x_2)}{dh_p}[n] \\ \frac{dw_3}{dh_w}[n+1] &= \frac{df_{w,3,1}(u_2)}{dh_w}[n] + \frac{df_{w,2,2}(x_3)}{dh_p}[n] \\ \frac{dw_4}{dh_w}[n+1] &= \frac{df_{w,4,1}(u_2)}{dh_w}[n] + \frac{df_{w,4,2}(x_4)}{dh_p}[n] \end{aligned} \quad (12.94)$$

Para esta parte, dada la conexión de los bloques, la función f_{gij} depende implícitamente de los parámetros h_{wmnp} , por lo tanto:

$$\frac{df_{gij}}{dh_{wmnp}} = \sum_{l=1}^2 v_{wijnl} (1 + e^{-\sigma_{wijnl}(\chi - \gamma_{wijnl})})^{-2} e^{-\sigma_{wijnl}(\chi - \gamma_{wijnl})} \sigma_{wijnl} \frac{d\chi}{dh_{wmnp}} \quad (12.95)$$

Considerando lo anterior se establece que:

$$\begin{aligned}
\frac{dg_1}{dh_w}[n+1] &= \frac{df_{g,1,1}}{dz_1} \frac{dz_1}{dh_w}[n] + \frac{df_{g,1,2}}{dg_1} \frac{dg_1}{dh_w}[n] \\
\frac{dg_2}{dh_w}[n+1] &= \frac{df_{g,2,1}}{dz_1} \frac{dz_1}{dh_w}[n] + \frac{df_{g,2,2}}{dg_2} \frac{dg_2}{dh_w}[n] \\
\frac{dg_3}{dh_w}[n+1] &= \frac{df_{g,3,1}}{dz_2} \frac{dz_2}{dh_w}[n] + \frac{df_{g,3,2}}{dg_3} \frac{dg_3}{dh_w}[n] \\
\frac{dg_4}{dh_w}[n+1] &= \frac{df_{g,4,1}}{dz_2} \frac{dz_2}{dh_w}[n] + \frac{df_{g,4,2}}{dg_4} \frac{dg_4}{dh_w}[n]
\end{aligned} \tag{12.96}$$

Donde:

$$\frac{df_{gij}}{d\chi} = \sum_{l=1}^2 v_{wijl} (1 + e^{-\sigma_{wijl}(\chi - \gamma_{wijl})})^{-2} e^{-\sigma_{wijl}(\chi - \gamma_{wijl})} \sigma_{wijl} \tag{12.97}$$

Por su parte, para la función f_{wij} se debe observar la dependencia de los parámetros h_{wmnp} , considerando el caso donde $i \neq m$ y $j \neq n$ se establece:

$$\frac{df_{wij}}{dh_{wnml}} = 0 \tag{12.98}$$

Para el caso donde $i = m$ y $j = n$ se tiene que:

$$\frac{df_{wij}}{dh_{wijp}} = \frac{df_{wij1}}{dh_{wijp}} + \frac{df_{wij2}}{dh_{wijp}} \tag{12.99}$$

Para el respectivo cálculo se debe tener presente que las entradas u_1 y u_2 no dependen de h_w , es decir que f_{wi1p} solamente depende de h_{wi1p} , por lo tanto, respectivamente para cada parámetro:

$$\frac{df_{wi1p}}{dv_{wi1p}} = (1 + e^{-\sigma_{wi1p}(\chi - \gamma_{wi1p})})^{-1} \tag{12.100}$$

$$\frac{df_{wi1p}}{d\sigma_{wi1p}} = v_{wi1p} (1 + e^{-\sigma_{wi1p}(\chi - \gamma_{wi1p})})^{-2} e^{-\sigma_{wi1p}(\chi - \gamma_{wi1p})} (\chi - \gamma_{wi1p}) \tag{12.101}$$

$$\frac{df_{wi1p}}{d\gamma_{wi1p}} = v_{wi1p} (1 + e^{-\sigma_{wi1p}(\chi - \gamma_{wi1p})})^{-2} e^{-\sigma_{wi1p}(\chi - \gamma_{wi1p})} \sigma_{wi1p} \tag{12.102}$$

En el caso que $p \neq k$ entonces:

$$\frac{df_{wi1k}}{dh_{wi1p}} = 0 \tag{12.103}$$

Considerando las señales internas g_1, \dots, g_4 , entonces, la función f_{wi2p} depende directamente como también implícitamente de h_{wi2p} , de tal forma que:

$$\begin{aligned}
\frac{df_{wi2p}}{dv_{wi2p}} &= (1 + e^{-\sigma_{wi2p}(\chi - \gamma_{wi2p})})^{-1} - \\
&v_{wi2p} (1 + e^{-\sigma_{wi2p}(\chi - \gamma_{wi2p})})^{-2} e^{-\sigma_{wi2p}(\chi - \gamma_{wi2p})} (-\sigma_{wi2p}) \frac{d\chi}{dv_{wi2p}}
\end{aligned} \tag{12.104}$$

$$\frac{df_{wi2p}}{d\sigma_{wi2p}} = -v_{wi2p} \left(1 + e^{-\sigma_{wi2p}(\chi - \gamma_{wi2p})}\right)^{-2} e^{-\sigma_{wi2p}(\chi - \gamma_{wi2p})} \left(\gamma_{wi2p} - \chi - \sigma_{wi2p} \frac{d\chi}{d\sigma_{wi2p}}\right) \quad (12.105)$$

$$\frac{df_{wi2p}}{d\gamma_{wi2p}} = -v_{wi2p} \left(1 + e^{-\sigma_{wi2p}(\chi - \gamma_{wi2p})}\right)^{-2} e^{-\sigma_{wi2p}(\chi - \gamma_{wi2p})} \left(\sigma_{wi2p} - \sigma_{wi2p} \frac{d\chi}{d\gamma_{wi2p}}\right) \quad (12.106)$$

Adicionalmente si $p \neq k$, entonces, f_{wi2k} depende implícitamente de h_{wi2p} por lo cual:

$$\frac{df_{wi2k}}{dh_{wi2p}} = \left(1 + e^{-\sigma_{wi2k}(\chi - \gamma_{wi2k})}\right)^{-2} e^{-\sigma_{wi2k}(\chi - \gamma_{wi2k})} \sigma_{wi2k} \frac{d\chi}{dh_{wi2p}} \quad (12.107)$$

En términos generales, se tiene que:

$$\begin{aligned} \frac{dg_1}{dh_w} [n+1] &= \frac{df_{w,1,1}}{dh_w} + \frac{df_{w,1,2}}{dg_1} \frac{dg_1}{dh_w} [n] + \frac{df_{w,1,2}}{dh_w} \\ \frac{dg_2}{dh_w} [n+1] &= \frac{df_{w,2,1}}{dh_w} + \frac{df_{w,2,2}}{dg_2} \frac{dg_2}{dh_w} [n] + \frac{df_{w,2,2}}{dh_w} \\ \frac{dg_3}{dh_w} [n+1] &= \frac{df_{w,3,1}}{dh_w} + \frac{df_{w,3,2}}{dg_3} \frac{dg_3}{dh_w} [n] + \frac{df_{w,3,2}}{dh_w} \\ \frac{dg_4}{dh_w} [n+1] &= \frac{df_{w,4,1}}{dh_w} + \frac{df_{w,4,2}}{dg_4} \frac{dg_4}{dh_w} [n] + \frac{df_{w,4,2}}{dh_w} \end{aligned} \quad (12.108)$$

Donde las derivadas $df_{w,i,j}/dh_w$ corresponden a:

$$\frac{df_{wijp}}{dv_{wijp}} = \left(1 + e^{-\sigma_{wijp}(\chi - \gamma_{wijp})}\right)^{-1} \quad (12.109)$$

$$\frac{df_{wijp}}{d\sigma_{wijp}} = v_{wijp} \left(1 + e^{-\sigma_{wijp}(\chi - \gamma_{wijp})}\right)^{-2} e^{-\sigma_{wijp}(\chi - \gamma_{wijp})} (\chi - \gamma_{wijp}) \quad (12.110)$$

$$\frac{df_{wijp}}{d\gamma_{wijp}} = v_{wijp} \left(1 + e^{-\sigma_{wijp}(\chi - \gamma_{wijp})}\right)^{-2} e^{-\sigma_{wijp}(\chi - \gamma_{wijp})} \sigma_{wijp} \quad (12.111)$$

De la misma forma para las derivadas $df_{w,i,2}/dg_i$ se tiene:

$$\frac{df_{w,i,2}}{d\chi} = \sum_{l=1}^2 v_{wi2l} \left(1 + e^{-\sigma_{wi2l}(\chi - \gamma_{wi2l})}\right)^{-2} e^{-\sigma_{wi2l}(\chi - \gamma_{wi2l})} \sigma_{wi2l} \quad (12.112)$$

Como es de apreciar, se presentan realimentaciones internas en la estructura del modelo de la planta, por lo tanto, se utiliza el conjunto de ecuaciones recurrentes establecidas anteriormente. Así, los pasos del algoritmo utilizado para la identificación de la planta son los siguientes:

1. Establecer la configuración inicial de parámetros para el modelo de la planta (sistema neuro-difuso).

2. En la k -ésima etapa de entrenamiento, $k = 0, 1, 2, \dots, K_T$, para el valor actual de n y utilizando los datos de entrada (u_1, u_2) se calcula la salida del sistema (y_1, y_2) (modelo neuro-difuso de la planta).
3. Utilizando la parejas de datos de entrada-salida (u_1, y_{d1}) y (u_2, y_{d2}) , y los datos calculados del modelo (y_1, y_2) , se realiza el ajuste de los parámetros del sistema neuro-difuso utilizando las ecuaciones (12.71), (12.73) y subsecuentes. Es importante señalar que los parámetros que se ajustan se almacenan en variables auxiliares ya que durante este paso el sistema neuro-difuso no utiliza estos valores.
4. Regresar a la etapa 2 para el siguiente paso de simulación $n = n + 1$ (donde se calcula la salida del sistema neuro-difuso) hasta que n sea igual al valor definido N_T (tiempo de simulación).
5. En el caso de completar el tiempo de simulación, se actualizan los nuevos parámetros y se regresa al paso 2 para una nueva iteración $k = k + 1$, hasta que la función de ajuste $J_p(k) = \sum_{n=1}^{N_T} J_p(k, n)$ sea menor que un ε definido, o hasta que k sea igual a un valor determinado K_T .

12.8. Resultados experimentales

Con el fin de observar el comportamiento que tiene el sistema de control adaptativo, se realiza un conjunto de pruebas experimentales con la planta MIMO real, de tal forma que se pueda apreciar los ajustes iterativos para que la salida de la planta llegue al valor de referencia deseado.

El índice de desempeño considerado para evaluar el sistema corresponde al mostrado en la ecuación (12.113), siendo r_1, r_2 los valores de referencia deseados, y_1, y_2 las respuestas del sistema de control y N_T el número total de datos.

$$J = \frac{1}{N_T} \sum_{n=1}^{N_T} [(r_1[n] - y_1[n])^2 + (r_2[n] - y_2[n])^2] \quad (12.113)$$

Escribiéndolo de otra forma:

$$J = \frac{1}{N_T} \sum_{n=1}^{N_T} (r_1[n] - y_1[n])^2 + \frac{1}{N_T} \sum_{n=1}^{N_T} (r_2[n] - y_2[n])^2 = J_1 + J_2 \quad (12.114)$$

Donde:

$$J_1 = \frac{1}{N_T} \sum_{n=1}^{N_T} (r_1[n] - y_1[n])^2 \quad (12.115)$$

$$J_2 = \frac{1}{N_T} \sum_{n=1}^{N_T} (r_2[n] - y_2[n])^2 \quad (12.116)$$

Considerando los datos obtenidos en lazo abierto de la planta (figura 12.4) y los resultados reportados en [267], el comportamiento deseado corresponde a sistemas de primer

orden con tiempo de establecimiento de 60 segundos, de tal forma que los modelos de referencia son:

$$G_1(s) = \frac{1}{15s + 1} \quad (12.117)$$

$$G_2(s) = \frac{1}{15s + 1} \quad (12.118)$$

Tomando los valores de las referencias de la tabla 12.4, la respectiva respuesta de los modelos de referencia se puede observar en la figura 12.19.

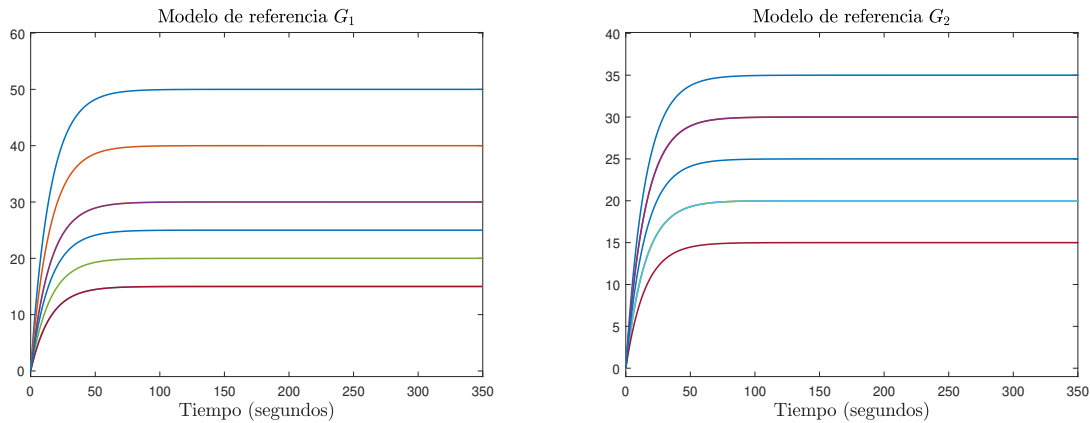


Figura 12.19: Simulación de los modelos de referencia.

Para llevar a cabo las pruebas experimentales se consideran tres posibles configuraciones del controlador neuro-difuso mostradas en la tabla 12.3, donde se plantea la presencia o ausencia de las funciones f_j que se tienen en la figura 12.13.

- CO1: se propone como la equivalencia con una función de transferencia con un polo y ganancia.

$$C_i(z) = \frac{b_1}{1 - a_0 z^{-1}} \quad (12.119)$$

La ecuación para implementar los subsistemas BBR es $w[n] = f_1(e[n]) + f_3(w[n-1])$.

- CO2: se considera una analogía de una función de transferencia con cero, polo y ganancia.

$$C_i(z) = \frac{b_1 + b_0 z^{-1}}{1 - a_0 z^{-1}} \quad (12.120)$$

La ecuación para la implementación de los subsistemas BBR es $w[n] = f_1(e[n]) + f_2(e[n-1]) + f_3(w[n-1])$.

- CO3: es una analogía con una función de transferencia con polo, ganancia y un cero en el origen.

$$C_i(z) = \frac{b_0 z^{-1}}{1 - a_0 z^{-1}} \quad (12.121)$$

La ecuación para implementar los subsistemas BBR corresponde a $w[n] = f_2(e[n-1]) + f_3(w[n-1])$.

Configuración	f_1	f_2	f_3
CO1	Si	No	Si
CO2	Si	Si	Si
CO3	No	Si	Si

Tabla 12.3: Resumen de las configuraciones considerando f_i de la figura 12.13.

Para establecer la configuración de los parámetros los subsistemas BBR, tal como se mencionó anteriormente, se consideran los conjuntos difusos de la figura 12.10 para modelar valores negativos y positivos; de esta forma, cada función f_j de la figura 12.20 se puede configurar para tener un comportamiento como el mostrado en la figura 12.21. Este enfoque permite establecer los parámetros de cada subsistema BBR mediante una analogía con el sistema lineal considerado. Teniendo presente el rango de valores de las variables de entrada, en este caso $[-100, 100]$, se configura el subsistema BBR para tener un comportamiento similar al lineal en este rango.

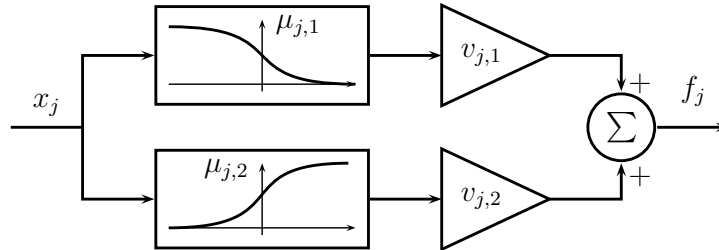


Figura 12.20: Configuración para la salida parcial f_j .

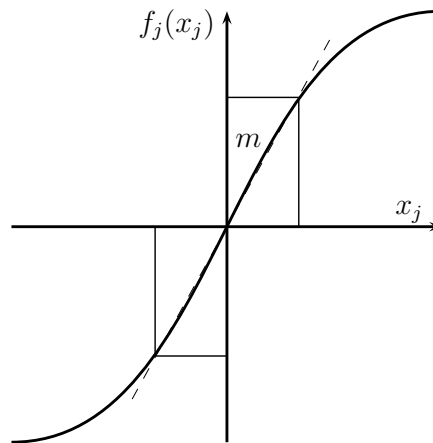


Figura 12.21: Configuración para la aproximación con pendiente positiva.

Por su parte, para la inicialización de la planta se consideran los valores obtenidos de la identificación de los parámetros de la sección 12.2, de esta forma, la configuración establecida en la figura 12.17 se propone para tener una equivalencia con el diagrama de la figura 12.3, por lo cual, la configuración inicial considera que $g_2 = 0$ y $g_3 = 0$. Por otra parte, la salida intermedia $z_1 = w_1 w_3$ se hace equivalente a $u_1(100 - u_2)$, y la otra salida

parcial $z_2 = w_2w_4$ se hace equivalente a u_1u_2 . La función de $(100 - u_2)$ se logra mediante la configuración que permite tener f_j de la figura 12.22.

La configuración de $BBR_{2,1}$ y $BBR_{2,4}$ permite comportamientos equivalentes a sistemas de primer orden con un tiempo de establecimiento de 100 segundos similar al presente en los depósitos de la planta real. Los parámetros de las funciones de pertenencia se configuran para tener una relación de f_j como la mostrada en las figuras 12.21 y 12.22, donde se presentan funciones con pendientes positivas y negativas.

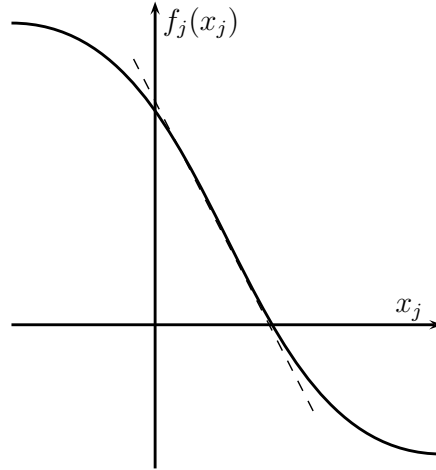


Figura 12.22: Configuración para la aproximación con pendiente negativa.

Para las pruebas se consideran los diferentes valores de referencia mostrados en la tabla 12.4. Los resultados para la salida del sistema se puede apreciar en la figura 12.23, donde también se puede observar la acción de control asociada. En la figura 12.24 se muestran resultados para las referencias R_1 , R_3 y R_7 . Por su parte, la figura 12.25 presenta los resultados para las referencias R_2 , R_6 y R_8 . Finalmente, en la figura 12.25 se tienen los resultados para las referencias R_4 y R_5 . En estos resultados se aprecia que el sistema presenta una mayor variación para valores altos de la referencia, también se puede observar que cada 100 segundos se realiza un proceso de identificación y adaptación del controlador siendo el primer proceso de adaptación donde se presenta la mayor variación. Luego, en la medida que se realizan más procesos de adaptación, las salidas se aproximan a las referencias. Es de notar en estos resultados que las configuraciones CO1 y CO3 son las que presentan la mayor variación en la mayoría de las referencias tomadas.

Además, en estos resultados se observa que la salida y_1 presenta más ruido en comparación con y_2 , lo cual está asociado con la ubicación del tanque $D1$ en la parte inferior del montaje, como se observa en la figura 12.1. A pesar de lo anterior, el sistema de control logra alcanzar los valores de referencia para ambas salidas y_1 y y_2 .

Salidas	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
y_1	50	40	30	30	20	15	15	25
y_2	35	30	20	30	20	20	15	25

Tabla 12.4: Referencias utilizadas.

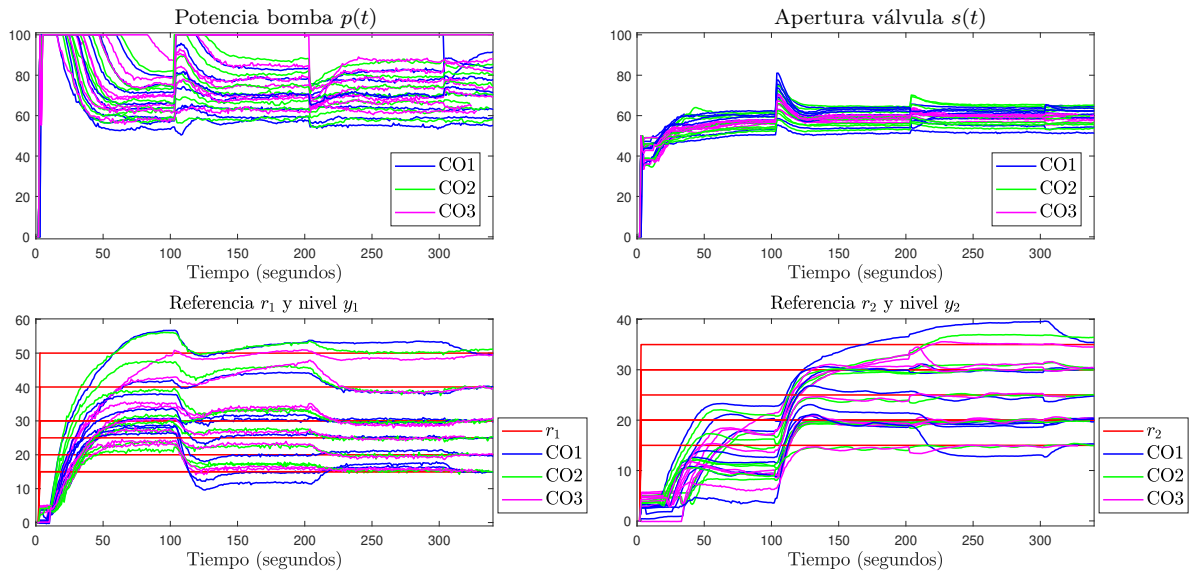


Figura 12.23: Resultados del sistema de control adaptativo.

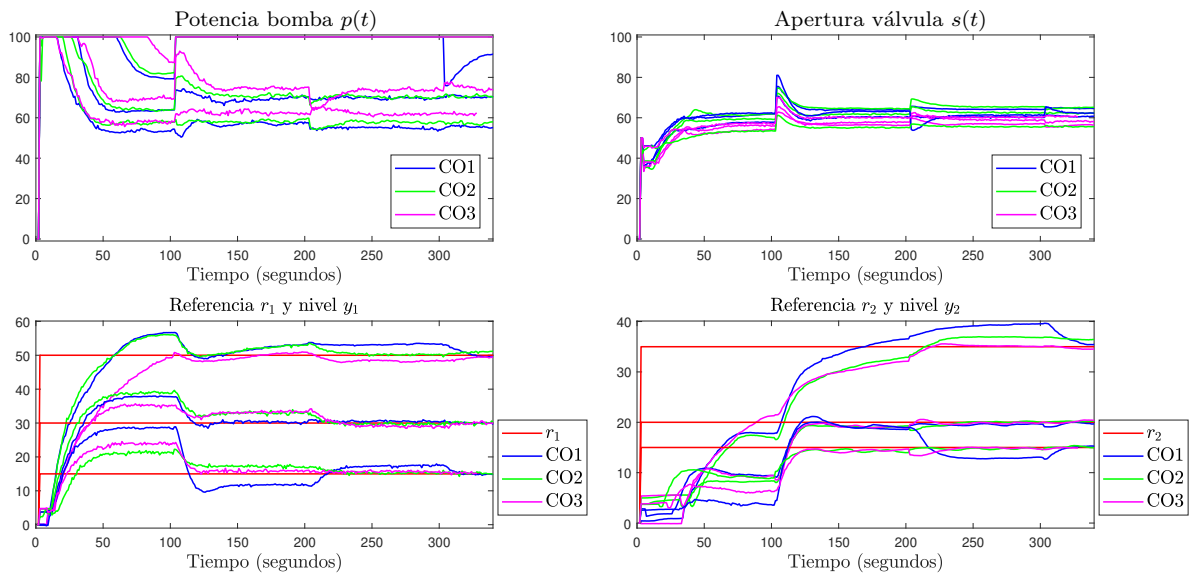


Figura 12.24: Resultados del sistema de control adaptativo, referencias R_1 , R_3 y R_7 .

Realizando el respectivo cálculo del error para cada referencia, en la tabla 12.5 se muestran los valores para J_1 y en la tabla 12.6 los resultados para J_2 , de esta forma, en la tabla 12.7 se pueden apreciar los valores de J para los diferentes sistemas y referencias. En estos resultados se puede apreciar que el mejor valor de J_1 , J_2 y J se obtiene para la configuración CO2. Considerando los resultados de la tabla 12.7 y tomando como referencia el mejor valor total obtenido para J , la diferencia de la configuración CO2 en relación al CO1 es 11,5 % y frente a CO3 es 11,7 %. Además, en la tabla 12.7 se aprecia que CO1 presenta los mejores resultados para R_2 y R_4 ; la configuración de CO2 obtiene el mejor resultado para R_1 , R_5 , R_6 , R_7 y R_8 , mientras que con CO3 el mejor resultado se obtiene para R_3 .

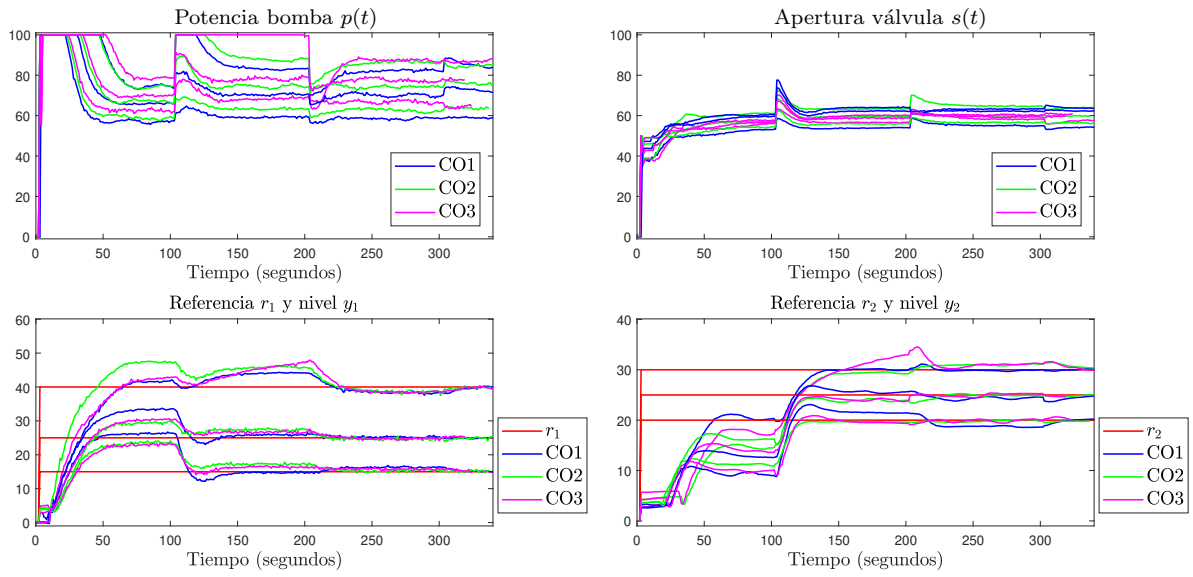


Figura 12.25: Resultados del sistema de control adaptativo, referencias R_2 , R_6 y R_8 .

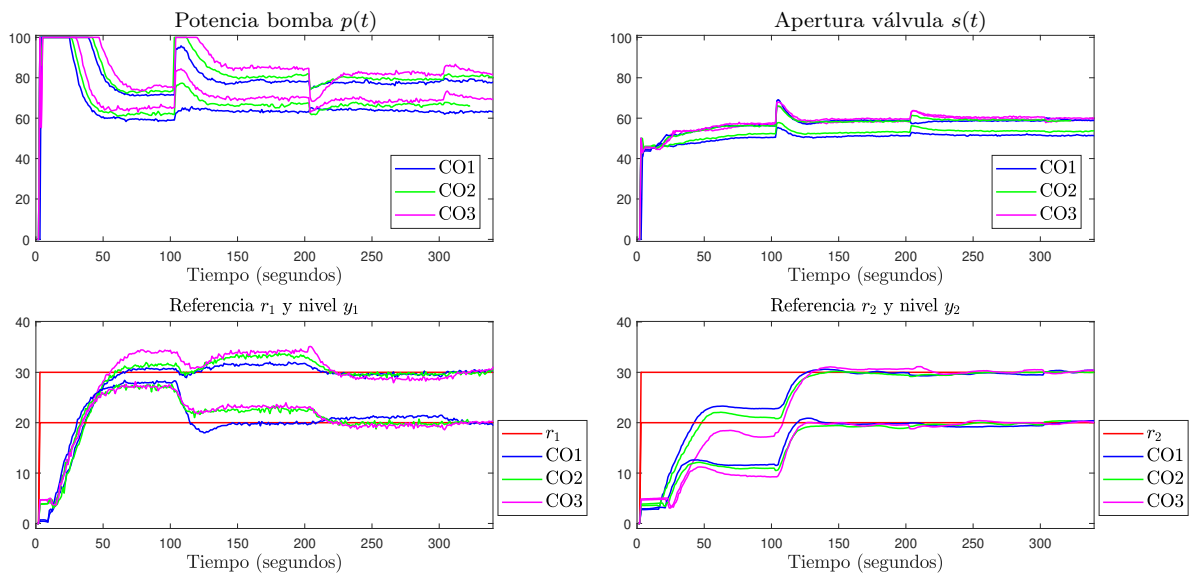


Figura 12.26: Resultados del sistema de control adaptativo, referencias R_4 y R_5 .

Configuración	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	Total
CO1	4.0317	3.0330	1.5658	1.8523	0.9244	1.1357	1.6300	1.2699	15.443
CO2	3.4896	2.5275	1.5308	1.7981	0.8279	0.7300	0.5484	1.0653	12.518
CO3	5.6787	2.8151	1.3731	1.6940	0.8448	0.6466	0.7030	1.1250	14.880

Tabla 12.5: Resultados obtenidos para J_1 en escala de 1×10^4 .

Configuración	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	Total
CO1	7.3540	3.1550	1.7838	2.2425	1.2314	1.6137	1.5872	2.4866	21.454
CO2	6.8454	4.3796	1.9408	2.4945	1.3056	1.2068	0.5229	1.8525	20.548
CO3	7.2573	3.6674	1.6123	3.4419	1.5170	1.4623	0.8969	2.2196	22.075

Tabla 12.6: Resultados obtenidos para J_2 en escala de 1×10^4 .

Configuración	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	Total
CO1	1.1386	0.6188	0.3350	0.4095	0.2156	0.2749	0.3217	0.3757	3.6897
CO2	1.0335	0.6907	0.3472	0.4293	0.2133	0.1937	0.1071	0.2918	3.3066
CO3	1.2936	0.6482	0.2985	0.5136	0.2362	0.2109	0.1600	0.3345	3.6955

Tabla 12.7: Resultados obtenidos para J en escala de 1×10^5 .

Finalmente, es de notar que el esquema de control neuro-difuso MIMO propuesto permite el ajuste iterativo del controlador para llegar a los valores de referencia. Para esto también se realizan ajustes progresivos de la identificación de la planta.

Capítulo 13

Conclusiones, aportes originales y trabajos futuros

13.1. Conclusiones

- Considerando las diferentes codificaciones de una tabla de verdad y criterios de diseño, se establecen los sistemas neuro-difusos compactos basados en relaciones booleanas utilizados para identificación y control. También en varios casos se deducen las ecuaciones de entrenamiento para estos sistemas.
- El uso de álgebra de Kleene fue un aspecto importante para lograr la equivalencia con las tablas de verdad de Boole y de esta forma poder establecer casos donde se puede tener la simplificación de términos, lo cual es el principio para el diseño de los sistemas neuro-difusos compactos basados en relaciones booleanas.
- Una de las estructuras compactas permite realizar la analogía con un sistema dinámico de tiempo discreto, lo que facilita la configuración de sus parámetros.
- En la aplicación del sistema de distribución de energía, el esquema de control neuro-difuso propuesto permite el ajuste del controlador después que ha ocurrido una variación de la carga. Se aprecia que el sistema opera satisfactoriamente para tres generadores, siendo esto de importancia en los sistemas de generación distribuida.
- En el análisis estadístico permite evidenciar que se logra de forma satisfactoria el proceso de adaptación en la mayoría de las configuraciones utilizadas. También se evalúan diferentes configuraciones, dependiendo de los retardos de la entrada y salida del sistema.
- Para la simulación con tres generadores, es de notar que se realizan varios ajustes para que los valores de salida lleguen a la referencia cuando existe una variación de la carga. El proceso adaptativo de cada generador se realiza por separado siendo esto un enfoque distribuido, es decir, que no existe un algoritmo principal que realice la optimización de los tres controladores, lo cual permite la conexión y desconexión de generadores.

- Para la identificación de la planta cuando existe la variación de la carga se dispone de una cantidad limitada de datos, lo cual influye en el ajuste del modelo de la planta.
- La identificación de la planta es un aspecto de gran importancia para el funcionamiento del sistema de control adaptativo, por lo cual, se realiza un ajuste progresivo para la identificación de la planta y con esta la optimización del controlador.
- La aplicación con los sistemas hidráulicos SISO permite observar que el sistema de control logra el seguimiento de la referencia tal como fue presentado con el tanque cónico. También permite la regulación de la salida cuando existe variación de parámetros, como lo mostrado con los tanques esféricos acoplados.
- En el caso del sistema SISO de tanques esféricos se utilizan valores P y Q para ponderar la señal de error y la acción de control en la función objetivo. El análisis estadístico realizado evidencia el efecto que tienen estos parámetros.
- Con la aplicación MIMO se muestra el caso de un sistema hidráulico real con varias no linealidades. Se realiza el diseño del sistema y deducen las ecuaciones de entrenamiento que permiten implementar el sistema de control el cual logra regular el nivel de los tanques para diferentes valores de referencia.
- Para el caso MIMO, mediante los resultados experimentales se logra comprobar que el esquema adaptativo propuesto permite llegar a los valores de referencia deseados. Se observa que en el primer proceso de adaptación se presenta la mayor variación.
- Se realizó la prueba experimental considerando varias configuraciones del controlador, compuesto por subsistemas BBR. En trabajos futuros se pueden considerar otras posibles configuraciones de estos sub-sistemas con los cuales se implementa el controlador.
- En relación al entrenamiento preliminar, para la aplicación del sistema de distribución se efectúa un entrenamiento inicial (fuera de línea) del controlador y del modelo neuro-difuso de la planta, en la aplicación del tanque cónico se lleva a cabo un entrenamiento preliminar del modelo neuro-difuso de la planta; mientras tanto, para el sistema SISO de tanques esféricos y para el sistema MIMO no se realizan entrenamientos preliminares antes de poner en funcionamiento el sistema de control adaptativo.

13.2. Aportes originales

- Se propone una mejora en el diseño de los sistemas de inferencia difusa basados en relaciones booleanas. Se emplea una extensión a una tabla de tres elementos considerando la condición de regularidad y se proponen varios criterios de diseño.
- Se proponen los sistemas difusos compactos basados en relaciones booleanas. Considerando la extensión a una tabla de tres elementos, se encuentran ecuaciones generales para la implementación de sistemas difusos basados en relaciones booleanas que presentan una codificación simplificada.

- El sistema de control neuro-difuso basado en relaciones booleanas se implementa de forma adaptativa y distribuida para el control en un sistema de distribución de energía cuando se tienen varios generadores.
- Se proponen diferentes configuraciones de los sistemas neuro-difusos compactos para la implementación del esquema de control adaptativo en un sistema de distribución de energía, como también en sistemas hidráulicos SISO y MIMO.

13.3. Trabajos futuros

- Estudiar otras posibles arquitecturas del sistema neuro-difuso como puede ser un controlador PID difuso.
- Análisis y diseño del sistema con valores propios y análisis de Liapunov para FIS-BBR.
- Plantear estrategias de control de modo deslizante utilizando FIS-BBR.
- Estudiar estrategias de control basado en pasividad mediante FIS-BBR.

13.4. Productos académicos

En esta sección se muestran los diferentes productos académicos que se obtuvieron en el desarrollo de este proyecto. Estos se organizan según su relación con el desarrollo realizado.

Pruebas preliminares con aplicaciones de energía:

- Helbert Eduardo Espitia, Guzmán Díaz González, *Identification of a Permanent Magnet Synchronous Generator using neuronal networks*, Workshop on Engineering Applications International Congress on Engineering (WEA 2015), Bogotá, Colombia, October 28-30, 2015.
<https://doi.org/10.1109/WEA.2015.7370143>
- Helbert Espitia, Guzmán Díaz, *Comparison between neuronal networks and ANFIS for wind speed-energy forecasting*, Workshop on Engineering Applications (WEA 2016), Bogotá, Colombia, September 21-23, 2016.
https://doi.org/10.1007/978-3-319-50880-1_9
- Helbert Eduardo Espitia, Guzmán Díaz González, Susana Irene Díaz, *Optimización de controladores PI discretos utilizando algoritmos PSO para la regulación de voltaje en una red de distribución*, Séptimo Congreso Internacional de Computación México-Colombia y XVI Jornada Académica en Inteligencia Artificial (CICOM 2017), Bogotá, Colombia, Septiembre 28-30, 2017.
<https://es.scribd.com/document/377605168/Memoria-Sci-Com-2017>

- Helbert Espitia, Iván Machón, Hilario López, *Control of a Microturbine Using Neural Networks*, Communications in Computer and Information Science 1052, 6th Workshop on Engineering Applications (WEA 2019), Santa Marta, Colombia, October 16-18, 2019.
https://doi.org/10.1007/978-3-030-31019-6_18
- Helbert Eduardo Espitia, Iván Machón-González, Hilario López-García, *Review of Control Techniques in Distributed Power Generation Systems*, TECCIENCIA, vol. 14, no. 27, 2019.
<https://tecciencia.ecci.edu.co/index.php/TECCIENCIA/article/view/105>
<https://dialnet.unirioja.es/servlet/articulo?codigo=7390332>

Pruebas preliminares de sistemas difusos:

- Helbert Eduardo Espitia, Guzmán Díaz González, *Optimization of linear and fuzzy controllers for cylindrical tank*, International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE 2015), Cuernavaca, México, November 24-27, 2015.
<https://doi.org/10.1109/ICMEAE.2015.30>
- Helbert Eduardo Espitia, Guzmán Díaz González, Susana Irene Díaz, *Adaptive model for a variable load in a distribution network using a neuro-fuzzy system*, IEEE Workshop on Power Electronics and Power Quality Applications (PEPQA 2017), Bogotá, Colombia, May 31 - June 2, 2017.
<https://doi.org/10.1109/PEPQA.2017.7981643>
- Helbert Eduardo Espitia, Iván Machón-González Hilario López-García, *Optimization of a Takagi-Sugeno fuzzy controller for voltaje regulation of a DC-AC single-phase inverter*, IEEE Workshop on Power Electronics and Power Quality Applications (PEPQA 2019), Manizales, Colombia, May 30-31, 2019.
<https://doi.org/10.1109/PEPQA.2019.8851572>
- Helbert Eduardo Espitia-Cuchango, Iván Machón-González, Hilario López-García, *Optimization of compact fuzzy controllers for temperature regulation*, Visión Electrónica, vol. 14, no. 1, 2020.
<https://revistas.udistrital.edu.co/index.php/visele/article/view/16012>
- Helbert Eduardo Espitia Cuchango, Iván José Machón González, Hilario López García, *Optimización de controladores difusos para la regulación de voltaje en una red de distribución*, XIII Congreso Internacional de Electrónica Control y Telecomunicaciones (CIECT 2018), Bogotá, Colombia, Noviembre 6-8, 2018.
<https://comunidad.udistrital.edu.co/ciect13/call-for-papers/>
https://www.dropbox.com/s/dcuoccm4yhha5b/Memorias%20CIECT_2018.pdf?dl=0
- Helbert Eduardo Espitia Cuchango, Iván Machón González, Hilario López García, Domingo Guzmán Díaz González, *Optimization of fuzzy controllers for a radial distribution network*, Visión Electrónica, vol. 13, no. 1, 2019.
<https://doi.org/10.14483/22484728.14681>

Pruebas preliminares del algoritmo de entrenamiento:

- Helbert Eduardo Espitia, Iván Machón-González, Hilario López-García, *Optimización de un controlador para un motor DC empleando Dynamic Backpropagation*, XIII Congreso Internacional de Electrónica Control y Telecomunicaciones (CIECT 2018), Bogotá, Colombia, Noviembre 6-8, 2018.
<https://comunidad.udistrital.edu.co/ciect13/call-for-papers/>
https://www.dropbox.com/s/dcuuoccm4yhha5b/Memorias%20_CIECT_2018.pdf?dl=0
- Helbert Eduardo Espitia-Cuchango, Iván Machón-González, Hilario López-García, *Optimization of a linear controller using dynamic back-propagation*, Visión Electrónica, vol. 13, no. 2, 2019.
<https://doi.org/10.14483/22484728.15183>
- Helbert Espitia, Iván Machón, Hilario López, *Optimization of a Fuzzy Automatic Voltage Controller Using Real-Time Recurrent Learning*, Processes, vol. 9, no. 6, paper 947, 2021.
<https://doi.org/10.3390/pr9060947>
- Helbert Espitia, Guzmán Díaz, Susana Díaz, *Control of a Permanent Magnet Synchronous Generator Using a Neuro-Fuzzy System*, Applied Computer Sciences in Engineering Communications in Computer and Information Science 915, 5th Workshop on Engineering Applications (WEA 2018), Medellín, Colombia, October 17-19, 2018.
https://doi.org/10.1007/978-3-030-00350-0_8

Sistemas difusos basados en relaciones booleanas:

- Helbert Espitia, José Soriano, Iván Machón, Hilario López, *Design Methodology for the Implementation of Fuzzy Inference Systems Based on Boolean Relations*, Electronics, vol. 8, no. 11, paper 1243, 2019.
<https://doi.org/10.3390/electronics8111243>
- Helbert Espitia, José Soriano, Iván Machón, Hilario López, *Compact Fuzzy Systems Based on Boolean Relations*, Applied Sciences, vol. 11, no. 4, paper 1793, 2021.
<https://doi.org/10.3390/app11041793>

Desarrollo de aplicaciones:

- Helbert Eduardo Espitia, Iván Machón-González, Hilario López-García, Guzmán Díaz, *Proposal of an Adaptive Neurofuzzy System to Control Flow Power in Distributed Generation Systems*, Complexity, vol. 2019, paper 1610898, 2019.
<https://doi.org/10.1155/2019/1610898>
- Helbert Espitia, Iván Machón, Hilario López, *Control of a MIMO coupled plant using a neuro-fuzzy adaptive system based on Boolean relations*, IEEE Access, vol. 9, pp. 59987-60009, 2021.
<https://doi.org/10.1109/ACCESS.2021.3073067>

Jornadas internacionales de doctorado Universidad de Oviedo:

- Espitia Cuchango, Helbert Eduardo, *SISTEMA NEURO-DIFUSO PARA EL CONTROL DE SISTEMAS DE GENERACIÓN DISTRIBUIDA*, VII Jornadas Internacionales de Doctorado de la Universidad de Oviedo, UNIVERSIDAD, INVESTIGACIÓN Y CONOCIMIENTO: Comprensión e intervención en una sociedad compleja, Editorial Universidad de Oviedo, 2018.

https://cei.uniovi.es/c/document_library/get_file?uuid=b1a452e9-ba7e-4d3a-b9ad-70821a900d2e&groupId=49472

- Espitia-Cuchango Helbert Eduardo, Machón-González Iván, López-García Hilario, *SISTEMA DE CONTROL NEURO-DIFUSO ADAPTATIVO BASADO EN RELACIONES BOOLEANAS*, VIII Jornadas Internacionales de Doctorado de la Universidad de Oviedo, UNIVERSIDAD, INVESTIGACIÓN Y CONOCIMIENTO: LA TRANSVERSALIDAD COMO REFERENTE DEL DESARROLLO PROFESIONAL Y SOCIAL, Editorial Universidad de Oviedo, 2019.

https://cei.uniovi.es/c/document_library/get_file?uuid=166e53a6-4f33-47b1-a5ad-bac9e66995dc&groupId=49472

Bibliografía

- [1] Stuart RUSSELL, Peter NORVIG, *Artificial Intelligence: A Modern Approach*, Prentice Hall, Ed. 3, 2009.
- [2] Alexander KRAMER, Fernando MORGADO-DIAS, *Artificial intelligence in process control applications and energy saving: a review and outlook*, Greenhouse Gases: Science and Technology, vol. 10, no. 6, pp. 1133-1150, 2020.
- [3] José SORIANO, *Propuesta de congresor basado en relaciones booleanas*, Revista de Ingeniería, vol. 6, no. 1, pp. 44-50, 2001.
- [4] Camilo GÁLVEZ, David VARGAS, *Identificación de personas por medio de diometría en la planta de la mano usando técnicas digitales de procesamiento de imágenes y congresor difuso basado en relaciones booleanas*, Proyecto de fin de grado, Ingeniería Electrónica, Universidad Distrital Francisco José de Caldas, 2005.
- [5] Detlef D. NAUCK, Andreas NÜRNBERGER, *Neuro-fuzzy Systems: A Short Historical Review*, Computational Intelligence in Intelligent Data Analysis. Studies in Computational Intelligence, vol 445, 2013.
- [6] Abeer ALSHEJARI, *Neuro-Fuzzy Based Intelligent Approaches to Nonlinear System Identification and Forecasting*, PhD Thesis, University of Westminster, London, United Kingdom, 2018.
- [7] Varun OJHA, Ajith ABRAHAM, Václav SNÁŠEL, *Heuristic design of fuzzy inference systems: A review of three decades of research*, Engineering Applications of Artificial Intelligence, vol. 85, pp. 845-864, 2019.
- [8] Bonifacio MARTÍN DEL BRIO, Alfredo SANZ, *Redes Neuronales y Sistemas Borrosos*, Alfaomega - Ra-ma, 3er edición, 2008.
- [9] Nazmul SIDDIQUE, Hojjat ADELI, *Computational Intelligence: Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing*, John Wiley & Sons, 2013.
- [10] Madjid TAVANA, Vahid HAJIPOUR, *A practical review and taxonomy of fuzzy expert systems: methods and applications*, Benchmarking: An International Journal, vol. 27, no. 1, pp. 81-136, 2019.

- [11] Hung T. NGUYEN, Nadipuram R. PRASAD, Carol L. WALKER, Elbert A. WALKER, *A First Course in Fuzzy and Neural Control*, Chapman and Hall/CRC, 1er edición, 2003.
- [12] Radim BELOHLAVEK, Joseph W. DAUBEN, George J. KLIR, *Fuzzy Logic and Mathematics: A Historical Perspective*, Oxford University Press, 2017.
- [13] Tsung-Chih LIN, Chia-Hao KUO, Tun-Yuan LEE, Valentina E. BALAS, *Adaptive fuzzy H^∞ tracking design of SISO uncertain nonlinear fractional order time-delay systems*, *Nonlinear Dynamics*, vol. 69, pp. 1639-1650, 2012.
- [14] Alvin SILVERSTEIN, Virginia SILVERSTEIN, Laura SILVERSTEIN, *Adaptation*, Twenty-First Century Books, 2007.
- [15] Robert J. STERNBERG, *Adaptive Intelligence: Surviving and Thriving in Times of Uncertainty*, Cambridge University Press, 2021.
- [16] Kevin PASSINO, *Biomimicry of bacterial foraging for distributed optimization and control*, *IEEE Control Systems Magazine*, vol. 22, no. 3, 2002.
- [17] Kevin PASSINO, *Biomimicry for optimization, control, and automation*, Springer-Verlag, London, UK, 2005.
- [18] Shankar SASTRY, Marc BODSON, *Adaptive Control: Stability, Convergence and Robustness*, Dover Books on Electrical Engineering Series, Dover Publications, 2011.
- [19] Nhan T. NGUYEN, *Model-Reference Adaptive Control: A Primer*, *Advanced Textbooks in Control and Signal Processing*, Springer International Publishing, 2018.
- [20] Ioan Doré LANDAU, Rogelio LOZANO, Mohammed M'SAAD, Alireza KARIMI, *Adaptive Control, Algorithms, Analysis and Applications*, Springer, 2nd edition, 2011.
- [21] Petros A. IOANNOU, Jing SUN, *Robust Adaptive Control*, Dover Publications, 2012.
- [22] Harsh S. DHIMAN, Dipankar DEB, Vlad MURESAN, Valentina E. BALAS, *Wake Management in Wind Farms: An Adaptive Control Approach*, *Energies*, vol. 12, no. 7, 2019.
- [23] Yanjun ZHANG, Gang TAO, Mou CHEN, Liyan WEN, Zhengqiang ZHANG, *A matrix decomposition based adaptive control scheme for a class of MIMO non-canonical approximation systems*, *Automatica*, vol. 103 pp. 490-502, 2019.
- [24] H.R. MAIER, S. RAZAVI, Z. KAPELAN, L.S. MATOTT, J. KASPRZYK, B.A. TOLSON, *Introductory overview: Optimization using evolutionary algorithms and other metaheuristics*, *Environmental Modelling & Software*, vol. 114, pp. 195-213, 2019.
- [25] Rajesh Kumar ARORA, *Optimization: Algorithms and Applications*, Chapman and Hall/CRC, 2015.
- [26] Peter A.N. BOSMAN, *On Gradients and Hybrid Evolutionary Algorithms for Real-Valued Multi-Objective Optimization*, *IEEE Transactions On Evolutionary Computation*, vol. 16, no. 1, 2012.

- [27] Mykel J. KOCHENDERFER, Tim A. WHEELER, *Algorithms for Optimization*, MIT Press, 2019.
- [28] Fadzil AHMAD, Nor Ashidi MAT ISA, Muhammad KHUSAIRI OSMAN, Zakaria HUSSAIN, *Performance comparison of gradient descent and Genetic Algorithm based Artificial Neural Networks training*, 10th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 604-609, 2010.
- [29] James C.R. WHITTINGTON, Rafal BOGACZ, *Theories of Error Back-Propagation in the Brain*, Trends in Cognitive Sciences, vol. 23, no. 3, pp. 235-250, 2019,
- [30] Jorge CERVANTES, Wen YU, Sergio SALAZAR, Isaac CHAIREZ, *Takagi-Sugeno Dynamic Neuro-Fuzzy Controller of Uncertain Nonlinear Systems*, IEEE Transactions on Fuzzy Systems, vol. 25, no. 6, pp. 1601-1615, 2017.
- [31] Qian SHI, Hak-Keung LAM, Chengbin XUAN, Ming CHEN, *Adaptive neuro-fuzzy PID controller based on twin delayed deep deterministic policy gradient algorithm*, Neurocomputing, vol. 402, pp. 183-194, 2020.
- [32] Robert BABUŠKA, *Fuzzy Modeling for Control*, Volumen 12 de International Series in Intelligent Technologies, Springer Science & Business Media, 2012.
- [33] Gonzalo PAJARES, Matilde SANTOS, *Inteligencia Artificial e Ingeniería del Conocimiento*, Alfaomega Ra-Ma, 2006.
- [34] Alexander BALLÉN, Cesar RODRIGUEZ, *Diseño e implementación de un controlador difuso autosintonizado sobre microcontroladores, aplicado al control del péndulo invertido*, Proyecto de fin de grado, Ingeniería Electrónica, Universidad Distrital Francisco José de Caldas, 2003.
- [35] Riccardo ROVATTI, Roberto GUERRIERI, Giorgio BACCARANI, *An Enhanced Two-Level Boolean Synthesis Methodology for Fuzzy Rules Minimization*, IEEE Transactions On Fuzzy Systems, vol. 3, no. 3, 1995.
- [36] José SORIANO, Andrés OLARTE, Miguel MELGAREJO, *Fuzzy Controller for MIMO Systems using Defuzzification based on Boolean Relations (DBR)*, The 14th IEEE International Conference on Fuzzy Systems, 2005.
- [37] José SORIANO, Juan FIGUEROA, *A Comparison of ANFIS, ANN and DBR systems on volatile Time Series Identification*, NAFIPS '07, North American Fuzzy Information Processing Society, 2007.
- [38] Lina Morales, Helbert Espítia, José Soriano, *Sistema Neuro-difuso aplicado a series de tiempo mackey-glass*, Revista Visión Electrónica: Algo Más Que Un Estado sólido, vol. 3, no. 2, 2009.
- [39] Lina Morales, Helbert Espítia, José Soriano, *Predicción de la serie de tiempo Lorenz usando un sistema neuro-DBR*, Revista Ciencia e Ingeniería Neogranadina, vol. 20-2, 2010.
- [40] Helbert ESPITIA, *Aplicación del DBR a sistemas difusos tipo II*, Tesis de Maestría, Ingeniería Industrial, Universidad Distrital Francisco José De Caldas, 2008.

- [41] Helbert ESPITIA, José Soriano, *Sistema de inferencia difusa basado en relaciones Booleanas*, Ingeniería, vol. 15, no. 2, pp. 52-66, 2010.
- [42] Helbert ESPITIA, Harold CHAMORRO, José SORIANO, *Fuzzy controller design using concretion based on boolean relations (CBR)*, 12th UK Workshop on Computational Intelligence (UKCI), 2012.
- [43] Milton GANTIVA, *Estudio comparativo entre los métodos de estimación de estado mediante filtro de kalman extendido y filtro de kalman “unscented”, y los métodos de control de modo deslizante y difuso CBR para la determinación y control de orientación de un satélite cubesat*, Proyecto de fin de grado, Ingeniería Electrónica, Universidad Distrital Francisco José de Caldas, 2014.
- [44] Ángela DUQUINO, Christian ROJAS, José SORIANO, Diana OVALLE, *Design and Implementation of a CBR Interval Type-2 Fuzzy Controller for Stabilizing an Inverted Pendulum*, IEEE Workshop on Engineering Applications International Congress on Engineering (WEA), 2015.
- [45] Erika VANEGAS, Carlos CHOLO, José SORIANO, *Comparison Between Interval Type-2 and Three Valued Algebra Fuzzy Logic Controllers*, Communications in Computer and Information Science vol. 657, Third Workshop on Engineering Applications WEA, pp. 235-246, 2016.
- [46] Erika ZUTTA, Jhonattan GANTIVA, José SORIANO, *Generalization of Fuzzy Inference System Based on Boolean and Kleenean Relations FIS-BKR for Modelling and Control*, Communications in Computer and Information Science vol. 742, 4th Workshop on Engineering Applications WEA, pp. 351-364, 2017.
- [47] Omar SALAZAR, Jairo SORIANO, Humberto SERRANO, *From binary logic functions to fuzzy logic functions*, Applied Mathematical Sciences, vol. 7, pp. 5129-5138, 2013.
- [48] Omar SALAZAR, Jairo SORIANO, *Método de simplificación de fórmulas por medio de álgebras finitas*, Editorial UD; Universidad Distrital Francisco José de Caldas: Bogotá, Colombia, 2018.
- [49] Jairo SORIANO, *Álgebra abstracta aplicada en ingeniería. Casos de aplicación en sistemas difusos tipo 1 y tipo 2*, Editorial UD; Universidad Distrital Francisco José de Caldas: Bogotá, Colombia, 2019.
- [50] Thomas WEISE, *Global optimization algorithms - theory and application*, Self-Published Thomas Weise, 2009.
- [51] Pouria TOORANJIPOUR, Ramin VATANKHAH, Adaptive critic-based quaternion neuro-fuzzy controller design with application to chaos control, Applied Soft Computing, vol. 70, pp. 622-632, 2018.
- [52] Rachel W. SOARES, Luciana R. BARROSO, Omar A.S. AL-FAHDAWI, Response attenuation of cable-stayed bridge subjected to central US earthquakes using neuro-fuzzy and simple adaptive control, Engineering Structures, vol. 203, pp. 109874, 2020.

- [53] Ememobog ESSIEN, Hussameldin IBRAHIM, Mehran MEHRANDEZH, Raphael IDEM, *Adaptive neuro-fuzzy inference system (ANFIS) - based model predictive control (MPC) for carbon dioxide reforming of methane (CDRM) in a plug flow tubular reactor for hydrogen production*, Thermal Science and Engineering Progress, vol. 9, pp. 148-161, 2019.
- [54] Mohammad Javad KHODAEI, M. Hadi BALAGHI I., Amin MEHRVARZ, Nader JALILI, *An Adaptive Multi-critic Neuro-fuzzy Control Framework for Intravenous Anesthesia Administration*, IFAC Papers On Line, vol. 51-34, pp. 202-207, 2019.
- [55] Fayez G. AREED, Amira Y. HAIKAL, Reham H. MOHAMMED, *Adaptive neuro-fuzzy control of an induction motor*, Ain Shams Engineering Journal, vol. 1, pp. 71-78, 2010.
- [56] Elkhatib KAMAL, Lounis ADOUANE, Rustem ABDRAKHMANOV, Nadir OUDDAH, *Hierarchical and Adaptive Neuro-Fuzzy Control for Intelligent Energy Management in Hybrid Electric Vehicles*, IFAC PapersOnLine, vol. 50-1, pp. 3014-3021, 2017.
- [57] Srinivasan ALAVANDAR, M.J. NIGAM, *New hybrid adaptive neuro-fuzzy algorithms for manipulator control with uncertainties-Comparative study*, ISA Transactions, vol. 48, pp. 497-502, 2009.
- [58] Doaa M. ATIA, Hanaa T. EL-MADANY, *Analysis and design of greenhouse temperature control using adaptive neuro-fuzzy inference system*, Journal of Electrical Systems and Information Technology, vol. 4, pp. 34-48, 2017.
- [59] S. MOHAMED, I.A. HAMEED, *A GA-Based Adaptive Neuro-Fuzzy Controller for Greenhouse Climate Control System*, Alexandria Engineering Journal, vol. 57, no. 2, pp. 773-779, 2018.
- [60] Shuai SONG, Baoyong ZHANG, Xiaona SONG, Yijun ZHANG, Zhengqiang ZHANG, Wenjie LI, *Fractional-order adaptive neuro-fuzzy sliding mode H_∞ control for fuzzy singularly perturbed systems*, Journal of the Franklin Institute, vol. 356, pp. 5027-5048, 2019.
- [61] Ahmad BAGHERI, Jalal Javadi MOGHADDAM, *Decoupled adaptive neuro-fuzzy (DANF) sliding mode control system for a Lorenz chaotic problem*, Expert Systems with Applications, vol. 36, pp. 6062-6068, 2009.
- [62] Saleh MASUMPOOR, Hamid YAGHOBI, Mojtaba Ahmadi KHANESAR, *Adaptive sliding-mode type-2 neuro-fuzzy control of an induction motor*, Expert Systems with Applications, vol. 42, pp. 6635-6647, 2015.
- [63] S.P. MOUSTAKIDIS, G. A. ROVITHAKIS, J. B. THEOCHARIS, *An adaptive neuro-fuzzy tracking control for multi-input nonlinear dynamic systems*, Automatica, vol. 44, pp. 1418-1425, 2008.
- [64] Pouria SARHADI, Behrooz REZAIE, Zahra RAHMANI, *Adaptive predictive control based on adaptive neuro-fuzzy inference system for a class of nonlinear industrial processes*, Journal of the Taiwan Institute of Chemical Engineers, vol. 61, pp. 132-137, 2016.

- [65] Shuai SONG, Baoyong ZHANG, Xiaona SONG, Zhengqiang ZHANG, *Adaptive neuro-fuzzy backstepping dynamic surface control for uncertain fractional-order nonlinear systems*, Neurocomputing, vol. 360, pp. 172-184, 2019.
- [66] Mohammad ABDELRAHMAN, Sang-Young PARK, *Spacecraft attitude control via a combined state-dependent Riccati equation and adaptive neuro-fuzzy approach*, Aerospace Science and Technology, vol. 26, pp. 16-28, 2013.
- [67] M. Hadi BALAGHI E., Ramin VATANKHAH, Mehrdad BROUSHAKI, Aria ALASTY, *Adaptive optimal multi-critic based neuro-fuzzy control of MIMO human musculoskeletal arm model*, Neurocomputing, vol. 173, pp. 1529-1537, 2016.
- [68] Mahmoud M. SAAFAN, Mohamed M. ABDELSALAM, Mohamed S. ELKSAS, Sabry F. SARAYA, Fayez F.G. AREED, *An adaptive neuro-fuzzy sliding mode controller for MIMO systems with disturbance*, Chinese Journal of Chemical Engineering, vol. 25, pp. 463-476, 2017.
- [69] Nguyen Quoc DINH, Nitin V. AFZULPURKAR, *Neuro-fuzzy MIMO nonlinear control for ceramic roller kiln*, Simulation Modelling Practice and Theory, vol. 15, pp. 1239-1258, 2007.
- [70] V. NEKOUKAR, A. ERFANIAN, *Adaptive fuzzy terminal sliding mode control for a class of MIMO uncertain nonlinear systems*, Fuzzy Sets and Systems, vol. 179, pp. 34-49, 2011.
- [71] Tan Luy NGUYEN, *Adaptive dynamic programming-based design of integrated neural network structure for cooperative control of multiple MIMO nonlinear systems*, Neurocomputing, vol. 237 pp. 12-24, 2017.
- [72] Jin ZHANG, Shuling DAI, *Neural network control of discrete-time non-affine mimo systems with disturbances*, IFAC Proceedings, vol. 38, no. 1, pp. 931-936, 2005.
- [73] Qian YE, Xuyang LOU, Li SHENG, *Generalized predictive control of a class of MIMO models via a projection neural network*, Neurocomputing, vol. 234, 2017, pp. 192-197.
- [74] L. ENDER, R. SCHEFFER, R. Maciel FILHO, *Computer design of a new predictive adaptive controller coupling neural networks and Kalman filter applied to siso and mimo control*, Computer Aided Chemical Engineering, vol. 10, pp. 481-486, 2002.
- [75] Kun JIANG, Xiaomei WANG, Ben NIU, Zhenhua WANG, Junqing LI, Peiyong DUANA, Dong YANG, *Finite-time adaptive neural control and almost disturbance decoupling for disturbed MIMO non-strict-feedback nonlinear systems*, Journal of the Franklin Institute, vol. 357, no. 16, pp. 11750-11772, 2020.
- [76] Xinjun WANG, Xinghui YIN, Qinghui WU, Fanqi MENG, *Disturbance observer based adaptive neural control of uncertain MIMO nonlinear systems with unmodeled dynamics*, Neurocomputing, vol. 313, pp. 247-258, 2018.
- [77] Juan LI, Shihua LI, Xisong CHEN, Jun YANG, *RBFNDOB-based neural network inverse control for non-minimum phase MIMO system with disturbances*, ISA Transactions, vol. 53, pp. 983-993, 2014.

- [78] Yiyong GOU, Hongbo LI, Xinmin DONG, Zongcheng LIU, *Constrained adaptive neural network control of an MIMO aeroelastic system with input nonlinearities*, Chinese Journal of Aeronautics, vol. 30, no. 2, pp. 796-806, 2017.
- [79] Yongming LI, Tieshan LI, Shaocheng TONG, *Adaptive neural networks output feedback dynamic surface control design for MIMO pure-feedback nonlinear systems with hysteresis*, Neurocomputing, vol. 198, pp. 58-68, 2016.
- [80] Salem ZERKAOUI, Fabrice DRUAUX, Edouard LECLERCQ, Dimitri LEFEBVRE, *Stable adaptive control with recurrent neural networks for square MIMO non-linear systems*, Engineering Applications of Artificial Intelligence, vol. 22, no. 4-5, pp 702-717, 2009.
- [81] Ci CHEN, Zhi LIU, Kan XIE, Yun ZHANG, C.L. Philip CHEN, *Adaptive neural control of MIMO stochastic systems with unknown high-frequency gains*, Information Sciences, vol. 418-419, pp. 513-530, 2017.
- [82] Qinmin YANG, Zaiyue YANG, Youxian SUN, *Universal Neural Network Control of MIMO Uncertain Nonlinear Systems*, IEEE Transactions on Neural Networks and Learning Systems, vol. 23, no. 7, 2012.
- [83] Heqing LIU, Tianping ZHANG, Xiaonan XIA, *Adaptive neural dynamic surface control of MIMO pure-feedback nonlinear systems with output constraints*, Neurocomputing, vol. 333, pp. 101-109, 2019.
- [84] Yan WEI, Ping-fang ZHOU, Yue-ying WANG, Deng-ping DUAN, Weixiang ZHOU, *Adaptive neural dynamic surface control of MIMO uncertain nonlinear systems with time-varying full state constraints and disturbances*, Neurocomputing, vol. 364, pp. 16-31, 2019.
- [85] Sabrine SLAMA, Ayachi ERRACHDI, Mohamed BENREJEB, *Neural Adaptive PID and Neural Indirect Adaptive Control Switch Controller for Nonlinear MIMO Systems*, Mathematical Problems in Engineering, vol. 2019.
- [86] Sabrine SLAMA, Ayachi ERRACHDI, Mohamed BENREJEB, *Adaptive PID controller based on neural networks for mimo nonlinear systems*, Journal of Theoretical and Applied Information Technology JATIT, vol. 97, no 2, 2019.
- [87] Jeen LIN, Ruey-Jing LIAN, *Hybrid fuzzy-logic and neural-network controller for MIMO systems*, Mechatronics, vol. 19, pp. 972-986, 2009.
- [88] Patel HIMANSHUKUMAR, Shah VIPUL, *Decentralized stable and robust fault-tolerant PI plus fuzzy control of MIMO systems: a quadruple tank case study*, International Journal on Smart Sensing and Intelligent Systems, vol. 12, no. 1, 2019.
- [89] K. KHADEM, M. BASU, M.F. CONLON, *Parallel operation of inverters and active power filters in distributed generation*, Renewable and Sustainable Energy Reviews, vol. 15, pp. 5155-5168, 2011.
- [90] Zheng ZENG, Huan YANG, Rongxiang ZHAO, Chong CHENG, *Topologies and control strategies of multi-functional grid-connected inverters for power quality enhancement: A comprehensive review*, Renewable and Sustainable Energy Reviews, vol. 24, pp. 223-270, 2013.

- [91] Shashi Kant PANDEY, Soumya R. MOHANTY, Nand KISHOR, *A literature survey on load-frequency control for conventional and distribution generation power systems*, Renewable and Sustainable Energy Reviews, vol. 25, pp. 318-334, 2013.
- [92] Anurag CHAUHAN, R.P. SAINI, *A review on Integrated Renewable Energy System based power generation for stand-alone applications: Configurations, storage options, sizing methodologies and control*, Renewable and Sustainable Energy Reviews, vol. 38, pp. 99-120, 2014.
- [93] Allal M. BOUZID, Josep M. GUERRERO, Ahmed CHERITI, Mohamed BOUHAMIDA, Pierre SICARD, Mustapha BENGHANEM, *A survey on control of electric power distributed generation systems for microgrid applications*, Renewable and Sustainable Energy Reviews, vol. 44, pp. 751-766, 2015.
- [94] Omid PALIZBAN, Kimmo KAUHANIEMI, *Hierarchical control structure in microgrids with distributed generation: Island and grid-connected mode*, Renewable and Sustainable Energy Reviews, vol 44, pp. 797-813, 2015.
- [95] Nasif MAHMUD, A. ZAHEDI, *Review of control strategies for voltage regulation of the smart distribution network with high penetration of renewable distributed generation*, Renewable and Sustainable Energy Reviews vol. 64, pp. 582-595, 2016.
- [96] J.A. LAGHARI, H. MOKHLIS, M. KARIMI, A.H.A. BAKAR, Hasmaini MOHAMAD, *Computational Intelligence based techniques for islanding detection of distributed generation in distribution network: A review*, Energy Conversion and Management, vol. 88, pp. 139-152, 2014.
- [97] Ahmad Rezaee JORDEHI, *Allocation of distributed generation units in electric power systems: A review*, Renewable and Sustainable Energy Reviews, vol. 56, pp. 893-905, 2016.
- [98] Min DAI, Mohammad N. MARWALI, Jin-Woo JUNG, Ali KEYHANI, *Power Flow Control of a Single Distributed Generation Unit with Nonlinear Local Load*, IEEE PES Power Systems Conference and Exposition, pp. 398-403, 2004.
- [99] Min DAI, Mohammad Nanda MARWALI, Jin-Woo JUNG, Ali KEYHANI, *Power Flow Control of a Single Distributed Generation Unit*, IEEE Transactions on Power Electronics, vol. 23, no. 1, pp. 343-352, 2008.
- [100] Yong XUE, Jiamei DENG, Shuangbao MA, *Power Flow Control of a Distributed Generation Unit in Micro-grid*, IEEE 6th International Power Electronics and Motion Control Conference (IPEMC), pp. 2122-2125, 2009.
- [101] Yun Wei LI, Ching-Nan KAO, *An Accurate Power Control Strategy for Power-Electronics-Interfaced Distributed Generation Units Operating in a Low-Voltage Multibus Microgrid*, IEEE Transactions on Power Electronics, vol. 24, no. 12, pp. 2977-2988, 2009.
- [102] R. BOJOI, L. R. LIMONGI, D. ROIU A. TENCONI, *Enhanced Power Quality Control Strategy for Single-Phase Inverters in Distributed Generation Systems*, IEEE Transactions on Power Electronics, vol. 26, no. 3, pp. 798-806, 2011.

- [103] M. PRODANOVIC, K. DE BRABANDERE, J. VAN DEN KEYBUS, T. GREEN, J. DRIESEN, *Harmonic and reactive power compensation as ancillary services in inverter-based distributed generation*, Generation, Transmission & Distribution (IET), vol. 1, no. 3, pp. 432-438, 2007.
- [104] Pedro M.S. CARVALHO, Pedro F. CORREIA, Luís A.F.M. FERREIRA, *Distributed Reactive Power Generation Control for Voltage Rise Mitigation in Distribution Networks*, IEEE Transactions on Power Systems, vol. 23, no. 2, pp. 766-772, 2008.
- [105] Ferry A. VIAWAN, Daniel KARLSSON, *Coordinated Voltage and Reactive Power Control in the Presence of Distributed Generation* IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, pp. 1-6, 2008.
- [106] Fei WANG, Jorge L. DUARTE, Marcel A.M. HENDRIX, *Active and Reactive Power Control Schemes for Distributed Generation Systems Under Voltage Dips*, IEEE Energy Conversion Congress and Exposition (ECCE), pp. 3564-3571, 2009.
- [107] Gao YANG, Liu LI, Xu AO-RAN, Zhao YI, Sun WEN-YAO, Wng GANG, *Research on the Reactive Power Control of Distributed Generation System*, China International Conference on Electricity Distribution (CICED), pp. 1-5, 2012.
- [108] M.G. MOLINA, P.E. MERCADO, *Stabilization and control of tie-line power flow of microgrid including wind generation by distributed energy storage*, International Journal of Hydrogen Energy, vol 35, no. 11, pp. 5827-5833, 2010.
- [109] Maria Stefania CARMELIA, Francesco CASTELLI-DEZZA, Marco MAURI, Gabriele MARCHEGANI, Daniele ROSATI, *Control strategies and configurations of hybrid distributed generation systems*, Renewable Energy, vol. 41, pp. 294-305, 2012.
- [110] Shinichi KUSAW, Eiaske MASDA, Jimpei BABA, Masaaki OHSHIMA, Istvan NAGY, *Coordinate control of distributed generations with power converters in a micro grid*, European Conference on Power Electronics and Applications, 2005.
- [111] X. JIN, J. ZHAO, H.F. WANG, *On-Line Stability Control of Power Systems Integrated with Distributed Generation Systems*, Proceedings of the 41st International Universities Power Engineering Conference (UPEC), pp. 472-476, 2006.
- [112] E. ORTJOHANN, W. SINSUKTHAVORN, A. MOHD, M. LINGEMANN, N. HAMSIC, A. SCHMELTER, D. MORTON, *Control Methodology of Distributed Generation in Interconnected Grids* IEEE/PES Power Systems Conference and Exposition (PSCE), pp. 1-6, 2009.
- [113] Qiong TAO, Deshun WANG, Bo YANG, Huan LIU, Shengchi YAN, *Voltage Control of Distribution Network with distributed generation based on Voltage Sensitivity Matrix*, IEEE International Conference on Energy Internet, pp. 298-302, 2018.
- [114] Irvin J. BALAGUER, Qin LEI, Shuitao YANG, Uthane SUPATTI, Fang Zheng PENG, *Control for Grid-Connected and Intentional Islanding Operations of Distributed Power Generation*, IEEE Transactions on Industrial Electronics, vol. 58, no. 1, pp. 147-157, 2011.

- [115] Hua GENG, Dewei (David) XU, Bin WU, Geng YANG, *Active Islanding Detection for Inverter-Based Distributed Generation Systems With Power Control Interface*, IEEE Transactions on Energy Conversion, vol. 26, no. 4, pp. 1063-1072, 2011.
- [116] Xiaolong CHEN, Yongli LI, *An Islanding Detection Algorithm for Inverter-Based Distributed Generation Based on Reactive Power Control*, IEEE Transactions On Power Electronics, vol. 29, no. 9, pp. 4672-4683, 2014.
- [117] Majid MEHRASA, Edris POURESMAEIL, Hasan MEHRJERDI, Bo Nørregaard JØRGENSENB, João P.S. CATALÃO, *Control technique for enhancing the stable operation of distributed generation units within a microgrid*, Energy Conversion and Management, vol. 97, pp. 362-373, 2015.
- [118] Maurizio CIRRINCIONE, Marcello PUCCI, Gianpaolo VITALE, *Direct power control of three-phase VSIs for the minimization of common-mode emissions in distributed generation systems*, Electric Power Systems Research, vol. 81, pp. 830-839, 2011.
- [119] Yunqing PEI, Guibin JIANG, Xu YANG, Zhaoan WANG, *Auto-master-slave control technique of parallel inverters in distributed AC power systems and UPS*, IEEE 35th Annual Power Electronics Specialists Conference (PESC), vol. 3, pp. 2050-2053, 2004.
- [120] Milan PRODANOVIĆ, Timothy C. GREEN, *High-Quality Power Generation Thorough Distributed Control of a Power Park Microgrid* IEEE Transactions on Industrial Electronics, vol. 53, no. 5, pp. 1471-1482, 2006.
- [121] Adrian TIMBUS, Marco LISERRE, Remus TEODORESCU, Pedro RODRIGUEZ, Frede BLAABJERG, *Evaluation of Current Controllers for Distributed Power Generation Systems*, IEEE Transactions on Power Electronics, vol. 24, no. 3, pp. 654-664, 2009.
- [122] Petr HUSEK, *PID controller design for hydraulic turbine based on sensitivity margin specifications*, Electrical Power and Energy Systems, vol. 55, pp. 460-466, 2014.
- [123] Geng LIANG, Wen LI, Zhijun LI, *Control of superheated steam temperature in large-capacity generation units based on active disturbance rejection method and distributed control system*, Control Engineering Practice, vol. 21, no. 3, pp. 268-285, 2013.
- [124] Bongani MALINGA, John E. SNECKENBERGER, Ali FELIACHI, *Modeling and Control of a Wind Turbine as a Distributed Resource*, Proceedings of the 35th Southeastern Symposium on System Theory, pp. 108-112, 2003.
- [125] Peng LI, Ling ZHANG, Van WANG, Yin-Be SHENG, *Research on The Control of The Single-Stage Photovoltaic System in Microgrid*, China International Conference on Electricity Distribution (CICED), pp. 1-7, 2008.
- [126] Puja DASH, Lalit Chandra SAIKIA, Nidul SINHA, *Automatic generation control of multi area thermal system using Bat algorithm optimized PD-PID cascade controller*, Electrical Power and Energy Systems, vol. 68, pp. 364-372, 2015.
- [127] Swati SONDHI, Yogesh V. HOTE, *Fractional order PID controller for load frequency control*, Energy Conversion and Management, vol. 85, pp. 343-353, 2014.

- [128] Ashok Mohan JADHAV, K. VADIRAJACHARYA, *Performance Verification of PID Controller in an Interconnected Power System Using Particle Swarm Optimization*, Energy Procedia, 2nd International Conference on Advances in Energy Engineering (ICAEE), vol. 14, pp. 2075-2080, 2012.
- [129] G. MALLESHAM, S. MISHRA, A.N. JHA, *Maiden Application of Ziegler-Nichols Method to AGC of Distributed Generation System*, IEEE/PES Power Systems Conference and Exposition, pp. 1-7, 2009.
- [130] Xiao LI, Xiaohong JIAO, Luhao WANG *Coordinated power control of wind-PV-fuel cell for hybrid distributed generation systems*, Proceedings of SICE Annual Conference (SICE), pp. 150-155, 2013.
- [131] Chandra Shekar THELUKUNTLA, Mummadi VEERACHARY, *Resonant controller based single-phase Z-source inverter with LCL-filter*, Joint International Conference on Power Electronics, Drives and Energy Systems (PEDES), pp. 1-6, 2010.
- [132] J.M. PACAS, M.G. MOLINA, E.C. DOS SANTOS JR, *Design of a robust and efficient power electronic interface for the grid integration of solar photovoltaic generation systems*, International Journal of hydrogen energy, vol. 37, pp. 10076-10082, 2012.
- [133] S. SINAN, A. ELNADY, *Optimized PID Controller Based Voltage Oriented Control of the 7-level Diode Clamped Inverter for Distributed Generation System*, International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp. 1-5, 2017.
- [134] Farshid HABIBI, Ali Hesami NAGHSHBANDY, Hassan BEVRANI, *Robust voltage controller design for an isolated Microgrid using Kharitonov's theorem and D-stability concept*, Electrical Power and Energy Systems, vol. 44, pp. 656-665, 2013.
- [135] Mojtaba SHIROEI, Mohammad Reza TOULABI, Ali Mohammad RANJBAR, *Robust multivariable predictive based load frequency control considering generation rate constraint*, Electrical Power and Energy Systems, vol. 46, pp. 405-413, 2013.
- [136] Shashi Kant PANDEY, Soumya R. MOHANTY, Nand KISHOR, João P.S. CATALÃO, *Frequency regulation in hybrid power systems using particle swarm optimization and linear matrix inequalities based robust controller design*, Electrical Power and Energy Systems, vol. 63, pp. 887-900, 2014.
- [137] Yi HAN, Peter Michael YOUNG, Abhishek JAIN, Daniel ZIMMERLE, *Robust Control for Microgrid Frequency Deviation Reduction With Attached Storage System*, IEEE Transactions on Smart Grid, vol. 6, no. 2, pp. 557-565, 2015.
- [138] Mohammad Mahdi REZAEI, Jafar SOLTANI, *A robust control strategy for a grid-connected multi-bus microgrid under unbalanced load conditions*, Electrical Power and Energy Systems, vol. 71, pp. 68-76, 2015.
- [139] Vijay P. SINGH, Soumya R. MOHANTY, Nand KISHOR, Prakash K. RAY, *Robust H-infinity load frequency control in hybrid distributed generation system*, Electrical Power and Energy Systems, vol. 46, pp. 294-305, 2013.

- [140] Soumya R. MOHANTY, Nand KISHOR, Prakash K. RAY, *Robust H-infinite loop shaping controller based on hybrid PSO and harmonic search for frequency regulation in hybrid distributed generation system*, Electrical Power and Energy Systems, vol. 60, pp. 302-316, 2014.
- [141] Zhipeng WU, Fuwen YAN, Qing-Long HANG, *Distributed H_∞ Equalization Control for Multi-Node Power Generation Systems*, Proceeding of the 11 th World Congress on Intelligent Control and Automation, pp. 5276-5281, 2014.
- [142] J.R. FISCHER, S.A. GONZÁLEZ, I. CARUGATI, M.G. JUDEWICZ, D.O. CARRICA, *Control Directo de Potencia Predictivo Robusto con Sincronismo Intrínseco*, Revista Iberoamericana de Automática e Informática industrial, vol. 12, pp. 199-207, 2015.
- [143] Yasser Abdel-Rady Ibrahim MOHAMED, Hatem H. ZEINELDIN, M.M.A. SALAMA, R. SEETHAPATHY, *Seamless Formation and Robust Control of Distributed Generation Microgrids via Direct Voltage Control and Optimized Dynamic Power Sharing*, IEEE Transactions on Power Electronics, vol. 27, no. 3, pp. 1283-1294, 2012.
- [144] Majid MEHRASA, Edris POURESMAEIL, João P.S. CATALÃO, *Direct Lyapunov Control Technique for the Stable Operation of Multilevel Converter-Based Distributed Generation in Power Grid*, IEEE Journal of Emerging and Selected Topics in Power Electronics, vol. 2, no. 4, pp. 931-941, 2014.
- [145] Yung-Tsai WENG, Yuan-Yih HSU, *Sliding mode regulator for maximum power tracking and copper loss minimisation of a doubly fed induction generator*, IET Renewable Power Generation, vol. 9, no. 4, pp. 297-305, 2015.
- [146] Jun CHAI, Ricardo G. SANFELICE, *A robust hybrid control algorithm for a single-phase DC/AC inverter with variable input voltage*, American Control Conference (ACC), pp. 1420-1425, 2014.
- [147] V. YARAMASU, M. RIVERA, M. NARIMANI, B. WU, J. RODRIGUEZ, *High performance operation for a four-leg NPC inverter with two-sample-ahead predictive control strategy*, Electric Power Systems Research, vol. 123, pp. 31-39, 2015.
- [148] Alejandro J. del REAL, Alicia ARCE, Carlos BORDONS, *Combined environmental and economic dispatch of smart grids using distributed model predictive control*, Electrical Power and Energy Systems, vol. 54, pp. 65-76, 2014.
- [149] Da WANG, Mevludin GLAVIC, Louis WEHENKEL, *Comparison of centralized, distributed and hierarchical model predictive control schemes for electromechanical oscillations damping in large-scale power systems*, Electrical Power and Energy Systems, vol. 58, pp. 32-41, 2014.
- [150] M. SHIROEI, A.M. RANJBAR, *Supervisory predictive control of power system load frequency control*, Electrical Power and Energy Systems, vol. 61, pp. 70-80, 2014.
- [151] Miaomiao MA, Hong CHEN, Xiangjie LIU, Frank ALLGÖWER, *Distributed model predictive load frequency control of multi-area interconnected power system*, Electrical Power and Energy Systems, vol. 62, pp. 289-298, 2014.

- [152] G. BRUNI, S. CORDINER, V. MULONE, V. ROCCO, F. SPAGNOLO, *A study on the energy management in domestic micro-grids based on Model Predictive Control strategies*, Energy Procedia, International Conference on Applied Energy (ICAE), vol. 61, pp. 1012-1016, 2014.
- [153] Peng KOU, Feng GAO, Xiaohong GUAN, *Stochastic predictive control of battery energy storage for wind farm dispatching: Using probabilistic wind power forecasts*, Renewable Energy, vol. 80, pp. 286-300, 2015.
- [154] Zhuoyu GUO, Haixiao JIANG, Yi ZHENG, Shaoyuan LI, *Distributed Model Predictive Control for Efficient Operation of Islanded Microgrid*, Chinese Automation Congress (CAC), pp. 6253-6258, 2017.
- [155] Xiao-Feng LI, Jian SUN, Hui-Yan WU, Wei-Dong ZONG, *Application of The Fuzzy-PID to The Power Plant*, Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS), pp. 210-215, 2007.
- [156] R. GARDUNO-RAMIREZ, K.Y. LEE, *Fuzzy Gain-Scheduling PID+Decoupling Control for Power Plant Wide-Range Operation*, International Conference on Intelligent Systems Applications to Power Systems (ISAP), pp. 1-6, 2007.
- [157] Guillaume RAMI, Tuan TRAN-UOC, Nouredine HADJSAID, *Fuzzy logic supervision and control of distributed generators*, 18th International Conference and Exhibition on Electricity Distribution (CIRED), pp. 1-5, 2005.
- [158] H. SHAYEGHI, H.A. SHAYANFAR, A. JALILI, *Multi-stage fuzzy PID power system automatic generation controller in deregulated environments*, Energy Conversion and Management, vol. 47, pp. 2829-2845, 2006.
- [159] Zhitong GUO, Kwang Y. LEE, *A self-adaptive fuzzy PI controller of power conditioning system for hybrid fuel-cell/turbine power plant*, North American Power Symposium (NAPS), pp. 1-6, 2011.
- [160] Rabindra Kumar SAHU, G.T. Chandra SEKHAR, Sidhartha PANDA, *DE optimized fuzzy PID controller with derivative filter for LFC of multi source power system in deregulated environment*, Ain Shams Engineering Journal, vol. 6, pp. 511-530, 2015.
- [161] Engin YESIL, *Interval type-2 fuzzy PID load frequency controller using BigBang.Big Crunch optimization*, Applied Soft Computing, vol. 15, pp. 100-112, 2014.
- [162] R.K. PATNAIK, P.K. DASH, *Impact of wind farms on disturbance detection and classification in distributed generation using modified Adaline network and an adaptive neuro-fuzzy information system*, Applied Soft Computing, vol. 30, pp. 549-566, 2015.
- [163] Matheus Ferreira ZAMBRONI DE SOUZA, Yuri REIS, Adriano Batista ALMEIDA, Isaias LIMA, Antonio Carlos ZAMBRONI DE SOUZA, *Load margin assessment of systems with distributed generation with the help of a neuro-fuzzy method*, IET Renewable Power Generation, vol. 9, no. 4, pp. 331-339, 2015.
- [164] Inamanamelluri KUMARSWAMY, Tara Kalyani SANDIPAMU, Venkata PRASANTH, *Analysis of Islanding Detection in Distributed Generation using Fuzzy Logic Technique*, 7th Asia Modelling Symposium (AMS), pp. 3-7, 2013.

- [165] P.K. DASH, Malhar PADHEE, T.K. PANIGRAHI, *A hybrid time-frequency approach based fuzzy logic system for power island detection in grid connected distributed generation*, *Electrical Power and Energy Systems*, vol. 42, pp. 453-464, 2012.
- [166] H. SHAYEGHI, B. SOBHANI, *Zero NDZ assessment for anti-islanding protection using wavelet analysis and neuro-fuzzy system in inverter based distributed generation*, *Energy Conversion and Management*, vol. 79, pp. 616-625, 2014.
- [167] Aihua WANG, *Design of fuzzy power system for distributed generation system*, 2nd International Conference on Intelligent Control and Information Processing (ICICIP), vol. 2, pp. 881-884, 2011.
- [168] Aihua WANG, *Optimization of Fuzzy Power System for Distributed Generation System Using Genetic Algorithm*, International Conference on Computer Distributed Control and Intelligent Environmental Monitoring (CDCIEM), pp. 537-540, 2011.
- [169] Sudipta Chakraborty, Marcelo G. Simões, *Fuzzy ARTMAP Based Forecast of Renewable Generation for a High Frequency AC Microgrid*, 31st Annual Conference of IEEE Industrial Electronics Society (IECON), 2005.
- [170] Hassan YOUSEF, *Adaptive fuzzy logic load frequency control of multi-area power system*, *Electrical Power and Energy Systems*, vol. 68, pp. 384-395, 2015.
- [171] Helmut WEISS, Jim XIAO, *Fuzzy system control for combined wind and solar power distributed generation unit*, IEEE International Conference on Industrial Technology, vol. 2, pp. 1160-1165, 2003.
- [172] T. Praveen KUMAR, N. SUBRAHMANYAM, M. SYDULU, *Control Strategies of a Fuzzy Controlled Solid Oxide Fuel Cell/Battery Distributed Generation System for Power Quality Enhancement*, International Conference on Circuit, Power and Computing Technologies (ICCPCT), pp. 64-69, 2014.
- [173] T. Praveen KUMAR, Y. CHANDRASHEKAR, N. SUBRAHMANYAM, M. SYDULU, *Control Strategies of a Fuzzy Controlled Grid Connected Hybrid PV/PEMFC/Battery Distributed Generation System*, IEEE Power and Energy Conference at Illinois (PECI), 2015.
- [174] Indranil PAN, Saptarshi DAS, *Fractional order fuzzy control of hybrid power system with renewable generation using chaotic PSO*, ISATransactions, pp. 1-11, 2015.
- [175] Tarkeshwar V. MUKHERJEE, *A novel quasi-oppositional harmony search algorithm and fuzzy logic controller for frequency stabilization of an isolated hybrid power system*, *Electrical Power and Energy Systems*, vol. 66, pp. 247-261, 2015.
- [176] Ismit MADO, Adi SOEPRIJANTO, SUHARTONO, *Design of Robust-Fuzzy Controller for SMIB Based on Power-Load Cluster Model with Time Series Analysis*, *Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, pp. 8-15, 2014.
- [177] S. Sajjad SEYEDALIPOUR, Habib Allah AALAMI, Amin BARZEGAR, *A Lyapunov Function-Based Adaptive Fuzzy Control for Integration of Distributed Generation Resources to Electrical Networks*, 5th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), pp. 184-189, 2017.

- [178] Amin HAJIZADEH, Masoud Aliakbar GOLKAR, *Intelligent power management strategy of hybrid distributed generation system*, Electrical Power and Energy Systems, vol. 29, pp. 783-795, 2007.
- [179] A. HAJIZADEH, M.A. GOLKAR, *Fuzzy neural control of a hybrid fuel cell/battery distributed power generation system*, IET Renewable Power Generation, vol. 3, no. 4, pp. 402-414, 2009.
- [180] Amin HAJIZADEH, Masoud Aliakbar GOLKAR, *Intelligent robust control of hybrid distributed generation system under voltage sag*, Expert Systems with Applications, vol. 37, pp. 7627-7638, 2010.
- [181] Sanjeev K. NAYAK, D.N. GAONKAR, R. SHIVARUDRASWAMY, *Fuzzy Logic Controlled Microturbine Generation System for Distributed Generation*, 2nd International Conference on Advances in Energy Engineering (ICAEE2011), Energy Procedia, vol. 14, pp. 1213-1219, 2011.
- [182] Abhishek R. SAKHARE, Asad DAVAN, Ali FELIACHI, *Control of Stand Alone Solid Oxide Fuel Cell using Fuzzy Logic*, Proceedings of the 35th Southeastern Symposium on System Theory, pp. 473-476, 2003.
- [183] Donghua PAN, Fengxiang WANG, *Modeling and Simulation of Fuzzy Control System for Dual Stator Winding Induction Generator*, Proceeding of International Conference on Electrical Machines and Systems, 2007.
- [184] H.T. NGUYEN, E.A. WALKER, *A first course in fuzzy logic*, 3a ed. Chapman and Hall/CRC. 2006.
- [185] Mai GEHRKE, Carol WALKER, Elbert WALKER, *A mathematical setting for fuzzy logic*, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 5, no. 3, pp. 223-238, 1997.
- [186] Mai GEHRKE, Carol WALKER, Elbert WALKER, *Some comments on fuzzy normal forms*, Proceedings of the ninth IEEE international conference on fuzzy systems, vol. 2, pp. 593-598, 2000.
- [187] Mai GEHRKE, Carol WALKER, Elbert WALKER, *Normal forms and truth tables for fuzzy logics*, Fuzzy Sets And Systems, vol. 138, no. 1, pp. 25-51, 2003.
- [188] R.L. GOODSTEIN, *Boolean Algebra*, Dover Publications, 2007.
- [189] José MUÑOZ, *Introducción a la teoría de conjuntos*, Facultad de Ciencias, Universidad Nacional de Colombia, 2002.
- [190] Robert BABUŠKA, *Fuzzy and Neural Control*, DISC Course Lecture Notes, Delft University of Technology, Delft, The Netherlands, 2001.
- [191] R. ZHANG, Y. PHILLIS, V. KOUIKOGLU, *Fuzzy Control of Queuing Systems*, Springer-Verlag London, 2005.
- [192] Li-Xin WANG, *A course in Fuzzy Systems and Control*, Prentice Hall, 1997.
- [193] L. TROIANO, L.J. RODRÍGUEZ, P. MARINARO, I. DÍAZ, *Statistical analysis of parametric t-norms*, Information Sciences, vol. 257, pp. 138-162, 2014.

- [194] E. KLEMENT, R. MESIAR, E. PAP, *Triangular Norms*, Kluwer Academic Publishers, 2000.
- [195] Jerry MENDEL, *Uncertain Rule-Based Fuzzy Logic Systems, Introduction and new directions*, Prentice Hall PTR, Upper Saddle River, NJ, 2001.
- [196] James ANDERSON, *Redes Neuronales*, Alfaomega, 2007.
- [197] Stanislaw H. ZAK, *Systems and Control*, Oxford University Press, 2003.
- [198] Daniel GRAUPE, *Principles Of Artificial Neural Networks: Basic Designs To Deep Learning*, World Scientific, 4th edition, 2019.
- [199] Alejandro MEJÍA, *Caso de control neuronal adaptable para sistemas multivariados*, Proyecto de fin de grado, Ingeniería de Control, Universidad Nacional de Colombia, Sede Medellín, 2009.
- [200] Jagannathan SARANGAPANI, *Neural Network Control of Nonlinear Discrete-Time Systems*, CRC Press, 2018.
- [201] Howard DEMUTH, Mark BEALE, *Neural Network Toolbox, User's Guide*, The MathWorks, Inc. Version 4, 2002.
- [202] Józef KORBICZ, Jan M. KOSCIELNY, Zdzislaw KOWALCZUK, Wojciech CHOLEWA, *Fault Diagnosis: Models, Artificial Intelligence, Applications*, Springer Science & Business Media, 2012.
- [203] Maciej LAWRYŃCZUK, *Computationally Efficient Model Predictive Control Algorithms: A Neural Network Approach*, Studies in Systems, Decision and Control, Springer Science & Business Media, 2014.
- [204] Sergey Edward LYSHEVSKI, *Mechatronics and Control of Electromechanical Systems*, CRC Press, 2017.
- [205] Hamid ASGARI, Xiao Qi CHEN, *Gas Turbines Modeling, Simulation, and Control: Using Artificial Neural Networks*, CRC Press, 2015.
- [206] Anne Katrine DUUN-HENRIKSEN, Signe SCHMIDT, Rikke Meldgaard RØGE, Jonas Bech MØLLER, Kirsten NØRGAARD, John Bagterp JØRGENSEN, Henrik MADSEN, *Model Identification Using Stochastic Differential Equation Grey-Box Models in Diabetes*, Journal of diabetes science and technology, vol. 7, no. 2, pp. 431-440, 2013.
- [207] Ruiyun QI, Gang TAO, Bin JIANG, *Fuzzy System Identification and Adaptive Control*, Communications and Control Engineering, Springer International Publishing, Ed. 1, 2019.
- [208] Manu SINGH, Isha SINGH, Alka VERMA, *Identification on Non Linear Series-Parallel Model Using Neural Network*, International Journal of Electrical and Instrumentation Engineering, vol. 3, no. 1, pp. 21-23, 2013.
- [209] Krzysztof PATAN, *Robust and Fault-Tolerant Control: Neural-Network-Based Solutions*, Volumen 197 de Studies in Systems, Decision and Control, Springer, 2019.

- [210] J. MENDEL, *Rule Based Fuzzy Logic Systems*, Prentice Hall, Upper Saddle River N.J., 2001.
- [211] Timothy J. ROSS, *Fuzzy Logic with Engineering Applications*, John Wiley & Sons, 2017.
- [212] Guanrong CHEN, Trung Tat PHAM, *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*, CRC Press, Boca Raton, 2000.
- [213] J. Eldon WHITESITT, *Boolean Algebra and Its Applications*, Dover Books on Computer Science, Dover Publications, 2012.
- [214] , J. Acedo SÁNCHEZ, *Instrumentación y control básico de procesos*, Editorial Díaz de Santos, S.A. 2013.
- [215] Patrick O.J. KALTJOB, *Control of Mechatronic Systems: Model-Driven Design and Implementation Guidelines*, Wiley, 2021.
- [216] Harold CHAMORRO, N.L. DIAZ, José SORIANO, Helbert ESPITIA, *Active and Reactive Power Flow Fuzzy Controller for VSC HVDC using DBR and DBR Type 2*, IEEE Annual Meeting of the North American Fuzzy Information Processing Society, 2011.
- [217] Helbert ESPITIA, *Aplicación del congresor basado en relaciones booleanas para sistemas de lógica difusa tipo dos*, Tesis de maestría, Ingeniería Industrial, Universidad Distrital Francisco José de Caldas, 2009.
- [218] Mai GEHRKE, Carol WALKER, Elbert WALKER, *Some Comments on Interval Valued Fuzzy Sets*, International Journal of Intelligent Systems, vol. 11. no. 10, pp. 751-759, 1996.
- [219] Erich Peter KLEMENT, Radko MESIAR, Endre PAP, *Triangular norms. Position paper I: basic analytical and algebraic properties*, Fuzzy Sets and Systems, vol. 143, no. 1, pp. 2-26, 2004.
- [220] M. MUKAIDONO, *Regular ternary logic function-Ternary logic functions suitable for treating ambiguity*, IEEE Transactions on Computers, vol. 35, pp. 179-183, 1986.
- [221] Y. YAMAMOTO, M. MUKAIDONO, *P-Functions-Ternary Logic Functions Capable of Correcting Input Failures and Suitable for Treating Ambiguities*, IEEE Transactions on Computers, vol. 41, no. 1, pp. 28-35, 1992.
- [222] R. MANI, T. THYAGARAJAN, *Design of Decentralized Fuzzy Controllers for Quadruple tank Process*, International Journal of Computer Science and Network Security, vol. 8, no. 11, pp. 163-168, 2008.
- [223] S. TUNYASRIRUT, S. WANGNIPPARNTO, *Level Control in Horizontal Tank by Fuzzy-PID Cascade Controller*, International Journal of Applied Science, Engineering and Technology, vol 4, no. 1, 2007.
- [224] Q. XIAO, D. ZOU, P. WEI, *Fuzzy Adaptive PID Control Tank Level*, International Conference on Multimedia Communications, pp. 149-152, 2010.

- [225] E. DOEBELIN, *Control System Principles and Design*, McGraw Hill Education; 4 edition, 2012.
- [226] Yue WU, Hui WANG, Biaobiao ZHANG, K.-L. DU, *Using Radial Basis Function Networks for Function Approximation and Classification*, International Scholarly Research Notices, paper 324194, vol. 2012.
- [227] Ravi PATEL, Dipankar DEB, Rajeeb DEY, Valentina E. BALAS, *Introduction to Adaptive Control*, Adaptive and Intelligent Control of Microbial Fuel Cells. Intelligent Systems Reference Library, vol. 161, Springer, Cham. 2020.
- [228] Gang TAO, Ge SONG, *Higher Order Tracking Properties of Model Reference Adaptive Control Systems*, IEEE Transactions on Automatic Control, vol. 63, no. 11, pp. 3912-3918, 2018.
- [229] O. De JESÚS, M.T. HAGAN, *Backpropagation Algorithms for a Broad Class of Dynamic Networks*, IEEE Transactions on Neural Networks, vol. 18, no. 1, pp. 14-27, 2007.
- [230] Ke-Lin DU, M.N.S. SWAMY, *Neural Networks and Statistical Learning*, Springer London, 2013.
- [231] Bindeshwar SINGH, Janmejaya SHARMA, *A review on distributed generation planning*, Renewable and Sustainable Energy Reviews, vol. 76, pp. 529-544, 2017.
- [232] JiaPeng TONG, Tao YU, *Nonlinear PID control design for improving stability of micro-turbine systems*, Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies DRPT, April 6-9, pp. 2515-2518, 2008.
- [233] H. JIAYI, J. CHUANWEN, X. RONG, *A review on distributed energy resources and microgrid*, Renew. Sustain Energy Rev. vol. 12, 2008.
- [234] Abbas KHORSHIDI, Mahdi ZOLFAGHARI, Maryam Akhavan HEJAZI, *Dynamic Modeling and Simulation of Microturbine Generating System for Stability Analysis in Microgrid Networks*, International Journal of Basic Sciences & Applied Research, vol. 3, no. 9, pp. 663-670, 2014.
- [235] P. PIAGI, R.H. LASSETER, *Microgrid: a conceptual solution*, Proc Power Electronics Specialists Conf. 6, pp. 4285-4290, 2004.
- [236] A.G. TSIKALAKIS, N.D. HATZIARGYRIOU, *Centralized control for optimizing microgrids operation*, IEEE Trans Energy Convers, pp. 241-248, 2008.
- [237] C. SREENIVASULU, G. Madhusudhana RAO, B.V. Sanker RAM, *Reliable Load Flow Solution for Controlling Power Network by FACTS Devices*, International Journal of Emerging Technology and Advanced Engineering, vol. 3, no. 1, 2013.
- [238] K. Krushna MURTHY, S.V. Jaya Ram KUMAR, *Three-Phase Unbalanced Radial Distribution Load Flow Method*, International Refereed Journal of Engineering and Science (IRJES), vol. 1, no. 1, 2012.

- [239] Hongwei LI, Yong JIN, Anan ZHANG, Xia SHEN, Chao LI, Bing KONG, *An improved hybrid load flow calculation algorithm for weakly-meshed power distribution system*, International Journal of Electrical Power & Energy Systems, vol. 74, pp. 437-445, 2016.
- [240] Wei YAN, Fang LIU, Guanjie WANG, Guoyu XU, Shanglian HUANG, *Layer-by-Layer Back/Forward Sweep Method for Radial Distribution Load Flow*, Proceedings of the Chinese Society for Electrical Engineering, vol. 23, no. 8, pp. 76-80, 2003.
- [241] Yubin YAO, Dan WANG, Zhiliang WU, Yong CHEN, *Convergence Analysis on the Power Flow Methods for Distribution Networks with Small Impedance Branches*, Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies DRPT, 2008.
- [242] Ming ZHANG, Pengxiang BI, Jian LIU, Fang XIE, *Research on Convergency of Branch Current Method Based Power Flow Calculation for Radial Distribution Network*, Proceedings of the Chinese Society for Electrical Engineering, vol. 23, no. 6, pp. 41-44, 2003.
- [243] Karim L. ANAYA, Michael G. POLLITT, *Distributed Generation: Opportunities for Distribution Network Operators, Wider Society and Generators*, Cambridge Working Paper in Economics CWPE-1505, 2015.
- [244] Maher A. AZZOUZ, Hany E. FARAG, E.F. EL-SAADANY, *Fuzzy-Based Control of On-Load Tap Changers under High Penetration of Distributed Generators*, International Conference on Electric Power and Energy Conversion Systems, Yildiz Technical University, Istanbul, Turkey, 2013.
- [245] S.N. SALIH, P. CHEN, *On coordinated control of OLTC and reactive power compensation for voltage regulation in distribution systems with wind power*, IEEE Transactions on Power Systems, Transactions on Power Systems, vol. PP(99), pp. 1-10, 2015.
- [246] Q. LIU, T. MOHAMED, T. KERDPHOL, Y. MITANI, *PID-MPC Based Automatic Voltage Regulator Design in Wide-Area Interconnected Power System*, International Journal of Emerging Technology and Advanced Engineering, vol. 4, no. 8, pp. 412-417, 2014.
- [247] P. ALCARIA, S.F. PINTO, J.F. SILVA, *Active Voltage Regulators for Low Voltage Distribution Grids: the Matrix Converter Solution*, 4th International Conference on Power Engineering, Energy and Electrical Drives Istanbul, Turkey, May 13-17, 2013.
- [248] Sujit KUMAR SINGH, Suman KUMAR, Sitaram GIRI, Er.L. RAMESH, *A Review of Voltage Control Technique of Grid Connected Distributed Generation*, International Journal of Innovative Research in Science, Engineering and Technology, vol. 3, no. 1, pp. 1565-1571, 2014.
- [249] Puthireddy Umapathi REDDY, Sirigiri SIVANAGARAJU, Prabandhamkam SANGAMESWARARAJU, *Impact of voltage regulators in unbalanced radial distribution systems using Particle Swarm Optimization*, International Journal of Advances in Engineering & Technology, vol. 2, no. 1, pp. 129-138, 2012.

- [250] H. Wayne BEATY, Donald G. FINK, *Standard Handbook for Electrical Engineers*, McGraw Hill Professional, 16th edition, 2012.
- [251] Ali, EHSAN, Qiang YANG, *Optimal integration and planning of renewable distributed generation in the power distribution networks: A review of analytical techniques*, Applied Energy, vol. 210, pp. 44-59, 2018.
- [252] Yuan SHUN, Han SHUI, *Distribution Grid Reactive Power Optimization and Compensation*, Chinese Electric Power Press, 2003.
- [253] Chi YONGNING, Guan HONGLIANG, Wang WEISHENG, *Enhancement of transient voltage stability of induction generator based wind farm by SVC and pitch control*, Automation of Electric Power Systems, vol. 31, no. 3, 2007.
- [254] R. MURALEKRISHNEN, P. SIVAKUMAR, *Improving the Power Quality Performance for Distributed Power Generation*, International Conference on Computing, Electronics and Electrical Technologies (ICCEET), 2012.
- [255] Yanhua LIU, Xu ZHANG, Dongmei ZHAO, Min MA, *Research on Wind Farm Reactive Power Compensation Capacity and Control Target*, Asia-Pacific Power and Energy Engineering Conference (APPEEC), 2011.
- [256] Jonathan NYE, Mobolaji BELLO, *Local Reactive Power Droop Control Modification for Distributed Generators*, IEEE PES T&D Conference and Exposition, 2014.
- [257] Guzmán DÍAZ, Javier GÓMEZ-ALEIXANDRE, José COTO, *Direct backward/forward sweep algorithm for solving load power flows in AC droop-regulated microgrids*, IEEE Transactions on Smart Grid, vol. 7, no. 5, pp. 2208-2217, 2016.
- [258] John H. SKILLINGS, *A First Course in the Design of Experiments: A Linear Models Approach*, Routledge, 2018.
- [259] A.J. ARRIAZA, F.M. FERNÁNDEZ, A. LÓPEZ, M. MUÑOZ, S. PÉREZ, A. SÁNCHEZ, *Estadística básica con R y R-Commander*, Servicio de Publicaciones de la Universidad de Cádiz, 2008.
- [260] Janez DEMSAR, *Statistical comparisons of classifiers over multiple data sets*, Journal of Machine Learning Research, vol. 7, 2006.
- [261] Salvador GARCÍA, Francisco HERRERA, *An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons*, Journal of Machine Learning Research, vol. 9, 2008.
- [262] Frank BRETZ, Torsten HOTHORN, Peter WESTFALL, *Multiple Comparisons Using R*, CRC Press, 2016.
- [263] Oscar O. MELO, Luis A. LÓPEZ, Sandra E. MELO, *Diseño de Experimentos Métodos y Aplicaciones*, Universidad Nacional de Colombia, Facultad de Ciencias, 2020.
- [264] Sateesh Kumar VAVILALA, Vinopraba THIRUMAVALAVAN, Chandrasekaran K, *Level control of a conical tank using the fractional order controller*, Computers and Electrical Engineering, vol. 87, 2020.

- [265] C. SREEPRADHA, P. DEEPA, RAMES C. PANDA, M. MANAMALI, R. SHIVAKUMAR, *Synthesis of fuzzy sliding mode controller for liquid level control in spherical tank*, Cogent Engineering, vol.3, no. 1, pp.1-16, 2016.
- [266] K. Suresh MANIC, V. RAJINIKANTH, S. ANANTHASIVAM, U. SURESH, *Design of Controller in Double Feedback Control Loop - An Analysis with Heuristic Algorithms*, Chemical Product and Process Modeling, vol. 10, no. 4, pp. 253-262, 2015.
- [267] Iván MACHÓN-GONZÁLEZ, Hilario LÓPEZ-GARCÍA, Ignacio BOCOS-BARRANCO, *Dynamics identification and control of nonlinear MIMO coupled plant using supervised neural gas and comparison with recurrent neural controller*, Neural Computing and Applications, vol. 32, pp. 18123-18142, 2020.