

SISTEMAS NEURO DIFUSOS APLICADOS AL CONTROL AUTOMÁTICO

**BERNARDO JAVIER GARCÍA MEZA
EDWAR JAVIER VILLAMIZAR RUEDA**

**UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
CARTAGENA DE INDIAS D. T. Y C.**

2005

SISTEMAS NEURO DIFUSOS APLICADOS AL CONTROL AUTOMÁTICO

BERNARDO JAVIER GARCÍA MEZA

EDWAR JAVIER VILLAMIZAR RUEDA

**Monografía presentada para obtener el título de
Ingeniero Electrónico**

Director

EDUARDO GÓMEZ VÁSQUEZ

Magíster en Ciencias Computacionales

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR

FACULTAD DE INGENIERÍA

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CARTAGENA DE INDIAS D. T. Y C.

2005

Cartagena de Indias D. T. y C, Mayo de 2005

Señores:

Comité Evaluador

Departamento de Ingeniería Eléctrica y Electrónica

La Ciudad

Respetados Señores

Tengo el agrado de presentar a su consideración el trabajo de grado del cual me desempeño como director de la monografía titulada **“SISTEMAS NEURO DIFUSOS APLICADOS AL CONTROL AUTOMÁTICO”** desarrollada por los estudiantes BERNARDO JAVIER GARCÍA MEZA Y EDWAR JAVIER VILLAMIZAR RUEDA, como requisito para obtener el título de ingenieros electrónicos.

Atentamente

Ing. Eduardo Gómez Vásquez

Cartagena de Indias D. T. y C, mayo de 2005

Señores:

Comité Evaluador

Departamento de Ingeniería Eléctrica y Electrónica

La Ciudad

Respetados Señores

Con mucha atención nos dirigimos a ustedes para presentar la monografía titulada: **“SISTEMAS NEURO DIFUSOS APLICADOS AL CONTROL AUTOMÁTICO”** para su estudio y evaluación como requisito fundamental para obtener el título de Ingeniero Electrónico.

En espera que esta cumpla con las normas pertinentes establecidas por la institución nos despedimos

Atentamente

Bernardo Javier García Meza

Edwar Javier Villamizar Rueda

Nota de aceptación

Firma del jurado

Firma del jurado

Cartagena de Indias D. T. y C, mayo de 2005

Bernardo Javier García Meza

A Dios

A mis padres y mi hermano

*A mis maestros y todos aquellos que me
han apoyado hasta culminar esta etapa de mis estudios*

Edwar Javier Villamizar Rueda

A mis padres,

A mis hermanos,

A mis sobrinas y demás

Familiares que depositaron

Gran confianza en mi y me apoyaron

En mi formación tanto profesional como personal.

AGRADECIMIENTOS

Los autores expresan sus agradecimientos al ingeniero Eduardo Gómez Vásquez por su colaboración en la realización de la presente monografía. De igual manera a todo el equipo docente de la Facultad de Ingeniería Electrónica de la Universidad Tecnológica de Bolívar, que se han esforzado incansablemente durante estos cinco años para formarnos como profesionales y personas.

AUTORIZACIÓN

Cartagena de Indias, mayo de 2005

Yo **Edwar Javier Villamizar Rueda**, identificado con número de cédula 73184062 de la ciudad de Cartagena, autorizo a la Universidad Tecnológica de Bolívar para hacer uso de mi trabajo de grado y publicarlo en el catálogo online de la Biblioteca.

EDWAR JAVIER VILLAMIZAR RUEDA

AUTORIZACIÓN

Cartagena de Indias, mayo de 2005

Yo **Bernardo Javier García Meza**, identificado con número de cédula 73196918 de la ciudad de Cartagena, autorizo a la Universidad Tecnológica de Bolívar para hacer uso de mi trabajo de grado y publicarlo en el catálogo online de la Biblioteca.

BERNARDO JAVIER GARCÍA MEZA

ARTICULO 107

La institución se reserva el derecho de propiedad intelectual de todos los Trabajos de Grado aprobados, los cuales no pueden ser explotados comercialmente sin su autorización

TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	1
1. CONTROL INTELIGENTE	3
1.1. GENERALIDADES	3
1.1.1. Sistemas Expertos	5
1.1.2. Algoritmos Genéticos	5
1.2. ESTRATEGIAS DEL CONTROL INTELIGENTE	6
1.2.1. Control Experto	7
1.2.2. Programación de ganancias	7
1.2.3. Modelamiento en adelanto	9
1.2.4. Control inverso directo	10
1.2.5. Control con modelo de referencia	11
1.2.6. Control con modelo interno	12
1.2.7. Control predictivo	13
1.2.8. Control adaptativo usando un índice genérico (control por aprendizaje con refuerzo)	14
1.2.9. Control adaptativo lineal	15
2. REDES NEURONALES ARTIFICIALES	16
2.1. GENERALIDADES	16
2.1.1. Características de las redes neuronales	16

2.1.2.	Aplicabilidad de las redes neuronales	17
2.2.	CONCEPTOS FUNDAMENTALES	18
2.2.1.	Elementos de una neurona artificial	18
2.2.2.	Estructura de una red neuronal artificial	20
2.3.	APRENDIZAJE EN LAS REDES NEURONALES	22
2.3.1.	Aprendizaje supervisado	22
2.3.2.	Aprendizaje no supervisado	23
2.3.3.	Aprendizaje <i>off-line</i> y aprendizaje <i>on-line</i>	24
2.4.	ARQUITECTURAS DE REDES NEURONALES	25
2.4.1.	Perceptrón monocapa	25
2.4.2.	Perceptrón multicapa	26
2.4.3.	Redes de funciones de base radial	28
2.4.4.	Otros tipos de redes neuronales	29
3.	SISTEMAS DIFUSOS DE CONTROL	31
3.1.	GENERALIDADES	31
3.1.1.	Características de los sistemas basados en lógica difusa	31
3.1.2.	Ventajas y desventajas de los sistemas de control difuso	32
3.1.3.	Aplicabilidad de la lógica difusa a los sistemas de control	33
3.2.	ESTRUCTURA DEL CONTROLADOR DIFUSO	34
3.2.1.	<i>Fuzzificador</i>	34
3.2.2.	Base de reglas difusas	35

3.2.3.	Reglas de inferencia	36
3.2.4.	<i>Defuzzificador</i>	36
3.3	SISTEMAS DIFUSOS COMO MÁQUINA DE INFERENCIA	37
3.3.1.	Sistemas de inferencia difusa tipo Mamdani	38
3.3.2.	Sistemas de inferencia difusa tipo Takagi Sugeno Kang	38
4.	SISTEMAS NEURO DIFUSOS	40
4.1.	GENERALIDADES	40
4.1.1.	Estructura general de un sistema neuro difuso	40
4.1.2.	Ventajas y desventajas del uso de sistemas neuro difusos	41
4.1.3.	Importancia de los sistemas neuro difusos en el campo del control	42
4.1.4.	Pasos de desarrollo de sistemas neuro difusos	43
4.2.	APRENDIZAJE EN SISTEMAS NEURO DIFUSOS	45
4.2.1.	Generación neuronal de reglas de control difuso	46
4.2.2.	Ajuste neuronal de funciones de membresía	47
4.2.3.	Ajuste de reglas del proceso de inferencia	50
4.3.	COMPARACIÓN DE CARACTERÍSTICAS ENTRE LAS REDES NEURONALES Y LA LÓGICA DIFUSA	52
4.4.	MODELOS COMBINADOS DE REDES NEURONALES Y SISTEMAS DIFUSOS	53
4.4.1.	Modelos neuro difusos concurrentes	54

4.4.2.	Redes neuronales difusas	54
4.4.3.	Sistemas neuro difusos	55
4.4.3.1.	Sistemas neuro difusos cooperativos	56
4.4.3.2.	Sistemas neuro difusos híbridos	58
4.5.	PROTOTIPOS DE SISTEMAS NEURO DIFUSOS HÍBRIDOS	62
4.5.1.	FALCON	62
4.5.2.	ANFIS	65
4.5.3.	GARIC	66
4.5.4.	NEFCON	68
4.5.5.	SONFIN	70
4.5.6.	FUN	72
4.5.7.	EfuNN	73
4.5.8.	Diseño evolutivo de sistemas neuro difusos	75
5.	APLICACIONES ACTUALES Y PERSPECTIVAS DE LOS SISTEMAS NEURO DIFUSOS EN EL CONTROL DE PROCESOS	78
5.1.	APLICACIONES ACTUALES DE LOS SISTEMAS NEURO DIFUSOS	78
5.1.1.	Sistemas de control supervisorio jerárquico basados en sistemas neuro difusos	78

5.1.2.	Control del torque en motores eléctricos	79
5.1.3.	Sistemas inteligentes para el control de la anestesia en los quirófanos de un hospital	81
5.1.4.	Control neuro difuso de un sistema Diesel de viento basado en biomasa	82
5.1.5.	Control neuro difuso de estructuras utilizando realimentación de aceleración	83
5.1.6.	Sistema de balanceo de columnas de fluido	85
5.2.	PERSPECTIVAS FUTURAS DE LOS SISTEMAS NEURO DIFUSOS	85
6.	CONCLUSIONES	89
7.	RECOMENDACIONES	91
	BIBLIOGRAFÍA	93
	ANEXOS	96

LISTA DE FIGURAS

	Pág.
Figura 1. Terminología de los Algoritmos Genéticos	6
Figura 2.a. Modelo genérico de modelamiento en adelante	9
Figura 2.b. Acople en paralelo de una red con una matriz	9
Figura 3. Control con modelo de referencia	12
Figura 4. Control con modelo interno	13
Figura 5. Control predictivo	14
Figura 6. Ventajas y desventajas de las redes neuronales	17
Figura 7. Esquema de una neurona artificial	19
Figura 8. Estructura de una red perceptrón monocapa	25
Figura 9. Estructura de una red perceptrón multicapa	26
Figura 10. Estructura de una red de función de base radial	28
Figura 11. Ventajas y desventajas de los sistemas de control difuso	33
Figura 12. Esquema de un controlador difuso	35
Figura 13. Sistema de inferencia difusa tipo Mamdani	38
Figura 14. Sistema de inferencia difusa tipo Takagi Sugeno Kang	39
Figura 15. Estructura general de un sistema neuro difuso	41
Figura 16. Importancia de los sistemas neuro difusos en el campo del control	43

Figura 17.	Representación de reglas para un espacio de dos entradas	47
Figura 18.	Efecto de la variación de los parámetros de una función de membresía tipo campana	49
Figura 19.	Combinaciones de modelos de redes neuronales y sistemas difusos	53
Figura 20.	Sistemas neuro difusos cooperativos	57
Figura 21.	Arquitectura de un sistema neuro difuso tipo Mamdani	59
Figura 22.	Arquitectura de un sistema neuro difuso tipo Takagi Sugeno Kang	61
Figura 23.	Arquitectura de un FALCON	63
Figura 24.	Arquitectura de un ANFIS	65
Figura 25.	Arquitectura de una ASN	67
Figura 26.	Arquitectura de un NEFCON	68
Figura 27.	Arquitectura de un SONFIN	71
Figura 28.	Arquitectura de un FUN	72
Figura 29.	Arquitectura de un EfuNN	75
Figura 30.a	Arquitectura de un diseño evolutivo de sistemas neuro-difusos	76
Figura 30.b	Estructura computacional de un diseño evolutivo de sistemas neuro difusos	77
Figura 30.	Sistema neuro difuso para evaluación de características	79
Figura 31.	Esquema de compensación para el control de rizado en el torque	80
Figura 32.	Control de anestesia en un quirófano	81
Figura 33.	Sistema de potencia Diesel de viento	83

Figura 34.	Fluidos magnetoreológicos	84
Figura 35.	Perspectivas de los sistemas neuro difusos	86

LISTA DE TABLAS

	Pág.
Tabla 1. Cuadro comparativo entre el Control inteligente y el Control Clásico	4
Tabla 2. Fortalezas y debilidades del control inverso directo	11
Tabla 3. Tipos comunes de funciones de transferencia	20
Tabla 4. Tipos de aprendizaje para redes neuronales	23
Tabla 5. Aplicaciones de las redes neuronales tipo perceptrón multicapa	27
Tabla 6. Otras arquitecturas de redes neuronales	29
Tabla 7. Tabla de verdad para la función lógica implicación y sus equivalentes	50
Tabla 8. Definición de algunas normas y conormas triangulares	51
Tabla 9. Comparación entre Redes Neuronales y Sistemas de Lógica Difusa	52

LISTA DE ANEXOS

	Pág.
ANEXO A. Características principales de las técnicas de control	97
ANEXO B. Glosario de términos	99
ANEXO C. Índice alfabético de materias	103

INTRODUCCIÓN

En los últimos años el análisis de los problemas de la Ingeniería de Control ha generado en los investigadores un interés hacia técnicas no convencionales (variable compleja, análisis en el espacio de estados) que permitan resolver problemas que presentan los sistemas del mundo real, esto es principalmente, las no linealidades y la variancia en el tiempo. En ayuda de las técnicas clásicas aparecieron los sistemas apoyados en Inteligencia Artificial para poder dotar a los controladores de habilidades casi humanas como la generalización y el aprendizaje, pero muchas veces al costo de grandes plataformas computacionales, como las de los Sistemas Expertos tan conocidos por la Ingeniería de Computación. Hoy en día nos enfrentamos a una nueva revolución en las Técnicas de Control Inteligente con el desarrollo de estructuras similares a la naturaleza misma del pensamiento humano. Así, se puede decir que las Redes Neuronales son la abstracción matemática del *hardware* del cerebro humano, con una asombrosa capacidad de aprendizaje pero a la vez extremadamente complejas. En este aspecto aparece el otro gran protagonista de esta revolución, capaz de procesar de una forma relativamente sencilla la información del mundo real, añadiéndole algo que durante siglos se consideró rígido e inflexible: el valor de verdad. En efecto, los sistemas basados en Lógica Difusa, considerados por algunos como el *software* del cerebro, procesan la información asumiendo que en el mundo real los límites entre conjuntos o clases no están tan claramente

definidos como lo están en el mundo de las matemáticas. Esta suposición tan sencilla como traída de los cabellos les significa a estos sistemas el ser capaces de manejar sistemas tan enormes como los Sistemas Expertos pero a un costo computacionalmente mas económico, y además, de una manera tan “sencilla” que puede ser interpretada por el lenguaje natural con el que nos comunicamos las personas.

Juntar estos dos esquemas de computación ha venido a originar lo que ya algunos consideran es lo más cercano a un cerebro humano sin serlo. Sin ir mas lejos en este aspecto, podemos asegurar a ciencia cierta que los sistemas neuro difusos (los hijos de estos dos esquemas) son una herramienta valiosa y que no se puede obviar ni desconocer en el ámbito de la investigación y desarrollo. Probablemente no sean el último escalón en esta búsqueda por un esquema de computación eficiente pero si son un referente que puede sorprender a muchos tanto en la Ingeniería del Control como en el Procesamiento de Señales.

Nuestra tarea comienza y termina justo en las orillas de este inmenso mar de conocimiento, pero pretende despertar a unos en el conocimiento de estas técnicas y motivar a otros en el estudio y desarrollo de aplicaciones de sistemas neuro difusos.

1. CONTROL INTELIGENTE

1.1. GENERALIDADES

El control clásico consta esencialmente de tratamiento de información en un nivel numérico y actuación sobre el mundo, no apareciendo en él actividades cognitivas de alto nivel de abstracción. Es por ello que en muchos casos se debe añadir a este tipo de controles la supervisión de un ser humano. El **Control Inteligente** se puede definir como aquel mecanismo que es capaz de razonar, comprender y aprender de la tarea de control que realiza, mejorando permanentemente su desempeño. Dentro de este tipo de problemas son de especial importancia los derivados de los comportamientos impredecibles de los sistemas a controlar.

El control inteligente comprende una serie de técnicas, tomadas fundamentalmente de la Inteligencia Artificial, con las que se pretenden resolver problemas de control inabordables por los métodos clásicos¹

La tabla 1 contiene una serie de características que sirven para comparar los esquemas de control clásico, tales como el análisis en frecuencia compleja o en el espacio de los estados, y las técnicas de control inteligente.

¹SANZ, R. *Arquitectura de Control Inteligente de Procesos (1990)*. Tesis Doctoral. Departamento De Automática, Ing. Electrónica e Informática Industrial.

CONTROL INTELIGENTE	CONTROL CLÁSICO
Manejo de datos de entrada imprecisos o ruidosos	Sensibilidad ante perturbaciones y datos de entrada poco confiables
Comportamiento cambiante con el conocimiento adquirido	Comportamiento altamente determinístico
Adaptación a partir de las condiciones del sistema	Muy poca capacidad de adaptación
Enfocado hacia la lógica y algorítmica	Enfocado hacia el tratamiento numérico de la información
Requisitos de cómputo altos	Requisitos moderados de cómputo
Respuesta variable e imprecisa	Respuesta altamente determinística

Tabla 1. Cuadro comparativo entre el Control inteligente y el Control Clásico

Entre las técnicas utilizadas por el control inteligente, sin duda las más sobresalientes y aplicadas son:

- Sistemas Expertos
- Algoritmos Genéticos
- Redes Neuronales
- Lógica Difusa

A continuación incluimos una breve descripción acerca de cada uno de estas técnicas de control inteligente. Profundizaremos con mayor detalle en las Redes Neuronales y los sistemas de Lógica Difusa en capítulos posteriores.

1.1.1. Sistemas Expertos

Como su nombre lo indica, el propósito de estos sistemas es el de emular el comportamiento de los expertos humanos², o en su defecto, del operador del proceso. Estos sistemas hacen uso de herramientas de la Inteligencia Artificial para inferir las acciones a realizar a partir de dos tipos de reglas: reglas referentes a la interpretación del estado del proceso (valores de las entradas) y reglas para la determinación de las actuaciones propiamente dichas.

Por lo general, estos sistemas son bastante voluminosos en cuanto a la información que deben manejar y no son tan flexibles como los sistemas basados en lógica difusa.

1.1.2. Algoritmos Genéticos

Esta técnica fue desarrollada inicialmente por John Holland³ y David Goldberg en el MIT (*Massachusetts Institute of Technology*) hacia las décadas de 1970 y 1980. Los algoritmos genéticos son modelos de adaptación de las reglas de un sistema basado en conocimiento experto, que tienen un esquema evolutivo similar al de los seres vivos. De esta manera, cuando se produce una variación en el entorno,

² ALAMÁN, X. *Introducción a la I.A. para la asistencia al control en Tiempo Real (1992)*. Ensayo publicado en el libro "La Inteligencia Artificial y el control en Tiempo Real". Editorial Repsol Colección

³ HOLLAND, J.H et al. *INDUCTION. Processes of Inference, Learning and Discovery (1989)*. MIT Press

estas reglas son afectadas por operadores como cruce (*crossover*) o mutación para dar origen a unas nuevas, con las que se esperan mejores resultados, aunque no siempre sea así. De igual forma que en la evolución de las especies, estos conjuntos de reglas codificados en **cromosomas**, realizan entre sí una competencia para sobrevivir o ser eliminados en virtud de su adaptación a las variaciones.

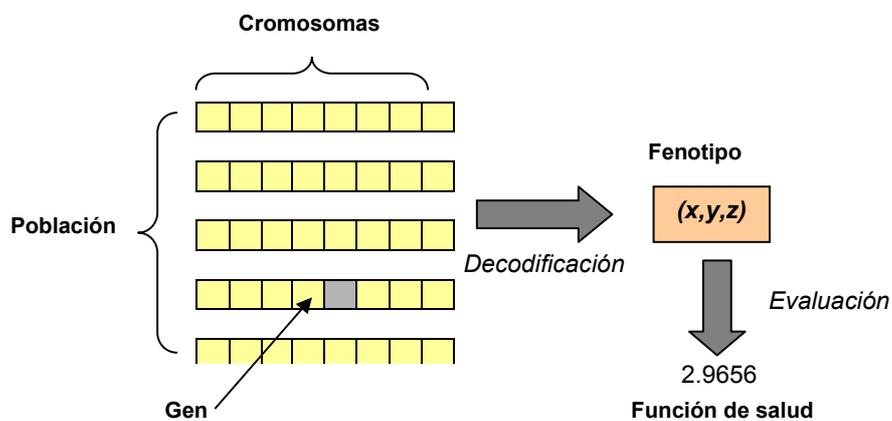


Figura 1. Terminología de los Algoritmos Genéticos

1.2. ESTRATEGIAS DEL CONTROL INTELIGENTE

Las estrategias del control inteligente son la manera como se realiza la tarea de control utilizando las técnicas mencionadas anteriormente. A continuación presentamos algunas de ellas⁴.

⁴ JANTZEN, Jan. *Neural and Neurofuzzy Control* (2003). Technical University of Denmark. Disponible en la Web: <http://fuzzy.iaue.dtu.dk/download/nnfcon.pdf>

1.2.1. Control Experto

El propósito original del control experto⁵ era imitar el comportamiento de un operador humano. Es una estrategia de control que permite modelar diferentes acciones complicadas y riesgosas de un operador en una determinada planta. En este caso, la entrada del sistema corresponde a la información del sensor recibida por el operador. Las **salidas deseadas** usadas para el entrenamiento corresponden a la entrada a la planta dada por el humano. Los **algoritmos de entrenamiento** pueden aprovechar los datos numéricos y refinar las funciones de membresía de manera sistemática. Las **reglas del operador** definen la estructura del controlador, y los datos medidos son usados para ajustar los parámetros.

Es característico que los controladores difusos incluyan información lingüística. Este tipo de controlador responde rápidamente y presenta un comportamiento consistente, en contraste con los operadores humanos, los cuales tienen estilos individuales de operación.

1.2.2. Programación de ganancias

⁵ MACDONALD, Bruce y ANDREAE, John H. *Expert Control For A Robot Body*. Department of Computer Science, University of Calgary. 1987. Disponible en la Web: http://pharos.cpsc.ucalgary.ca/Dienst/Repository/2.0/Body/ncstrl.ucalgary_cs/1987-286-34/pdf

Originalmente, esta estrategia de control se basaba en modificar las ganancias (proporcional, integral y derivativa) de un regulador PID (de ahí su nombre) cuya función de transferencia tiene la forma de la Ecuación (1)

$$G(s) = \frac{K_D s^2 + K_P s + K_I}{s} \quad (1)$$

Un controlador con programación de ganancias⁶ contiene un regulador lineal cuyos parámetros son cambiados en función del punto de operación de la planta, siguiendo una forma preprogramada. Un sistema difuso puede relacionar los parámetros del controlador con el estado de la planta de manera que su salida corresponda a los parámetros del controlador.

Específicamente, las entradas a un controlador con programación de ganancias comprenden dos tipos de variables: **variables programadoras** y **variables de estado**. Las variables programadoras se usan en la parte de la premisa (condiciones, antecedentes) para determinar que modo o características tiene la planta. Una vez que se han determinado el modo o características de la planta, se obtiene la regla correspondiente con una salida igual a las variables de estado multiplicadas por unas ganancias apropiadas de realimentación de estados.

Una desventaja de la programación de ganancias es que es una compensación de lazo abierto, es decir, no hay realimentación para compensar una programación

⁶ CORRIGA, Giorgio *et al.* *An Implicit Gain-Scheduling Controller for Cranes*. IEEE Transactions on Control Systems Technology, vol. 6, no. 1, January 1998. Disponible en la Web: <http://www.diee.unica.it/~giua/PAPERS/JOUR/98tcst.pdf>

incorrecta. Una ventaja es que los parámetros del controlador cambian tan pronto como cambian los parámetros de las variables programadoras, mientras que los métodos basados en modelos tienden a introducir una constante de tiempo o un retardo en el camino de actualización.

1.2.3. Modelamiento en adelanto

El modelamiento en adelanto es un procedimiento de entrenamiento de una red neuronal para representar la dinámica de una planta de acuerdo con la figura 2a. La red neuronal se coloca en paralelo con la planta y el error entre el sistema y las salidas de la red es usado como señal de entrenamiento.

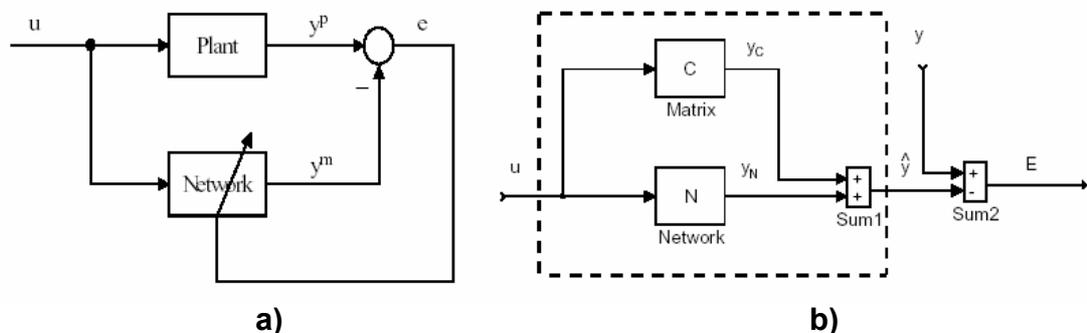


Figura 2. Modelamiento en adelanto

a) Modelo genérico; b) Acople en paralelo de una red con una matriz

Una posible forma de tener en cuenta la naturaleza dinámica de la planta es introducir dinámicas dentro de la red. Esto puede hacerse usando realimentación interna en la red (una red recurrente) o introduciendo funciones de transferencia

en las neuronas. Una aproximación directa es aumentar la entrada de la red con señales correspondientes a entradas y salidas pasadas.

Si se sabe que una planta no lineal tiene una parte lineal y una parte no lineal, una red puede ser entrenada para modelar solamente el comportamiento no lineal de la planta. La red es acoplada en paralelo con una matriz de ganancias constantes C , y ambos son alimentados con los datos de entrada a la planta, como se ilustra en la figura 2b.

La estructura paralela combinada tiene una salida total igual a la suma de las salidas de la red no lineal y la matriz lineal, y la matriz combinada de ganancias es la suma de las dos matrices de ganancia individuales. La configuración dentro del cuadro punteado de la figura 2b se comporta como una red ordinaria, y puede ser entrenada como una red.

1.2.4. Control inverso directo

El control inverso directo⁷ asume la existencia de un **modelo inverso de la planta**, aunque esto no es generalmente válido y resulta difícil de comprobar en muchas ocasiones. Lo que es más, la minimización del error de la red no garantiza la minimización del error del sistema completo.

⁷ HUSSAIN, Mohamed *et al.* *Implementation of Neural-Network-Based Inverse-Model Control Strategies on an Exothermic Reactor.* *ScienceAsia* vol 27. 2001. Disponible en la Web: http://scienceasia.tiac.or.th/PDF/vol27/v27_041_050.pdf

La tabla 2 presenta un esquema con las ventajas y desventajas de este esquema de control inteligente.

VENTAJAS	DESVENTAJAS
No se necesita conocimiento previo de la dinámica de la planta	Es posible que no se disponga con anterioridad del valor de salida deseada del siguiente intervalo
Es posible activar el aprendizaje <i>on-line</i> para trabajar con plantas variantes en el tiempo	Debe verificarse que la planta tenga un modelo inverso único
	Los datos de entrenamiento pueden estar distribuidos de forma inadecuada en el espacio de entradas

Tabla 2. Ventajas y desventajas del control inverso directo

1.2.5. Control con modelo de referencia

En esta estrategia de control⁸, el desempeño deseado del sistema en lazo cerrado se especifica a través de un **modelo de referencia estable**, definido por sus pares entrada-salida. El sistema de control intenta hacer que la salida de la planta tienda asintóticamente a la salida del modelo de referencia. En esta estructura el error (definido como la salida del modelo de referencia menos la salida de la planta) se usa para entrenar el controlador.

⁸ BOUZOUBAA, Mounir *et al.* *Model reference control of a laser beam steering system for laser communication applications*. Proc. SPIE Vol. 4272. 2001. Disponible en la Web: http://adsabs.harvard.edu/cgi-bin/nph-ib_query?bibcode=2001SPIE.4272...93B&db_key=INST

Claramente, esta aproximación se relaciona con el entrenamiento del control inverso directo, mencionado anteriormente. En el caso en el que el modelo de referencia corresponda a un mapeo identidad (entradas iguales a las salidas), las dos aproximaciones coinciden.

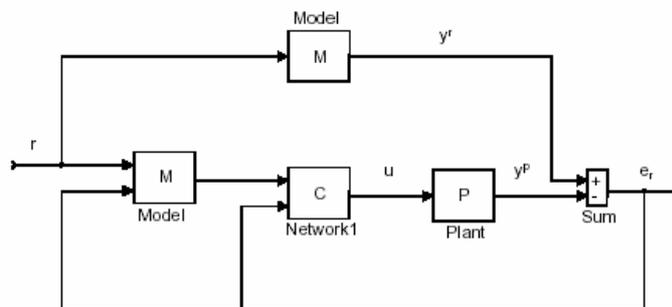


Figura 3. Control con modelo de referencia

1.2.6. Control con modelo interno

Aquí se enfatiza el papel del modelo en adelante y el modelo inverso. En esta estructura⁹ ambos son usados directamente como elementos dentro del lazo de realimentación. Un modelo en adelante de la planta se ubica en paralelo con la planta real. La diferencia entre las salidas de la planta y el modelo es usada para la realimentación, y esta es procesada por un subsistema controlador. El control con modelo interno dicta que esta parte del controlador debe estar relacionado con la inversa de la planta.

⁹ COLEMAN, Brosilow y BABU, Joseph. *One-Degree of Freedom Internal Model Control. Tomado del texto Techniques of Model-Based Control.* Prentice Hall. 2004.

Dados los modelos de redes para la dinámica en adelante e inversa de la planta es sencillo implementar el control con modelo interno; el modelo de la planta M y el controlador C (el modelo inverso) son realizados usando los modelos de redes neuronales como se muestra en la figura 4. El subsistema F es normalmente un filtro lineal.

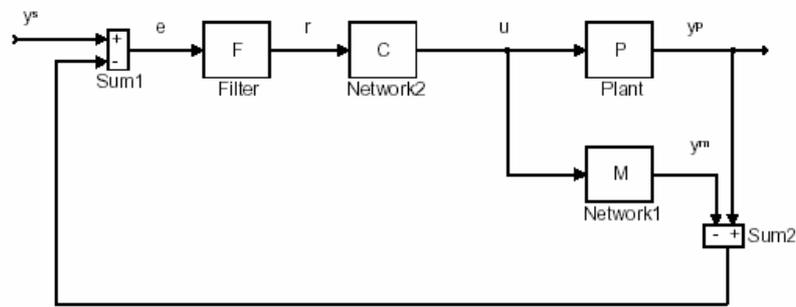


Figura 4. Control con modelo interno

Debe anotarse que esta estrategia de control está limitada a plantas estables en lazo abierto.

1.2.7. Control predictivo

En el método de control predictivo¹⁰ una red neuronal proporciona una **predicción de la respuesta futura** de la planta para un horizonte de predicción específico. Luego, estas predicciones pasan a una rutina de optimización numérica que calcula la

Artículo 1.
2001

¹⁰ MACIEJOWSKI, J. M. Predictive Control With Constraints. Pearson Education.

señal de control apropiada. A su vez, esta señal es escogida por el criterio de los **mínimos cuadrados**.

También es posible entrenar otra red para imitar la acción de la rutina de optimización. Esta red controladora se entrena para producir la misma salida de control (para una salida dada de la planta) que la rutina de optimización. Una ventaja es que el lazo exterior, que contiene el modelo de la planta y la rutina de optimización, ya no se necesita cuando el entrenamiento está completo.

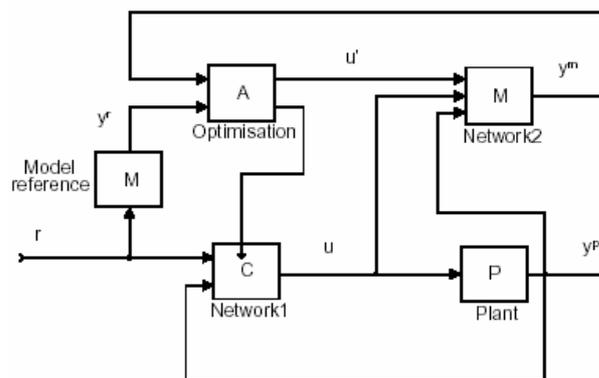


Figura 5. Control predictivo

1.2.8. Control adaptativo usando un índice genérico (control por aprendizaje con refuerzo)

Este tipo de estrategia¹¹ usa una señal de realimentación de baja calidad. Mientras que la medida del desempeño para un sistema con aprendizaje supervisado se define en términos de un conjunto de salidas deseadas con un error conocido, el aprendizaje con refuerzo soluciona el problema usando cualquier medida cuyo valor pueda ser proporcionado al algoritmo de aprendizaje. En lugar de intentar determinar las salidas deseadas del controlador a partir de las respuestas esperadas de la planta, se intenta determinar las salidas deseadas del controlador que lleven a un incremento en un **indicador de desempeño** de la planta, pero que no necesariamente se define en términos de las respuestas deseadas de la planta. Un bloque crítico es capaz de evaluar el rendimiento de la planta y generar una **señal de evaluación** que es usada por el algoritmo de aprendizaje con refuerzo.

Esta aproximación es apropiada cuando hay una falta del conocimiento requerido para aplicar métodos aprendizaje más especializado. Un ejemplo es el problema del péndulo invertido donde el refuerzo es cero a lo largo de un ensayo y se vuelve -1 cuando ocurre una falla. En un controlador con conocimiento más intensivo se necesita una trayectoria de referencia para controlar la planta.

1.2.9. Control adaptativo lineal

¹¹ TU, Jilin. *Continuous Reinforcement Learning for Feedback Control Systems*. Tesis de Maestría. Colorado State University. 2001. Disponible en la Web: <http://www.cs.colostate.edu/~anderson/res/rl/jilin-tu.ps.gz>

En este caso¹² la red está incluida en el camino de adaptación y es usada para minimizar simultáneamente las tasas de error cuadrático de los estados. La salida de la red representa los **parámetros de un modelo lineal** para la planta. Esta aproximación ha mostrado ser útil para la estimación de los mínimos cuadrados para plantas variantes e invariantes en el tiempo.

¹² CONRADIE, A.v.E et al. *Adaptive Control utilising Neural Swarming*: Proceedings of the Genetic and Evolutionary Computation Conference. 2002. Disponible en la Web: <http://www.chemeng.sun.ac.za/Content/gecco2002.pdf>

2. REDES NEURONALES ARTIFICIALES

2.1. GENERALIDADES

Una **red neuronal artificial** (RNA)¹³ es una estructura de procesamiento de datos basada en cálculos aritméticos simples ejecutados por elementos de procesamiento llamados comúnmente **neuronas** (por analogía a las neuronas biológicas) y conexiones entre estas representadas matemáticamente por factores de multiplicación también llamados **pesos**. Desde luego la analogía con las neuronas biológicas (unidades funcionales del cerebro) se relaciona directamente con el propósito original de emular las capacidades de memorizar, aprender y generalizar del cerebro humano emulando su modelo de funcionamiento.

2.1.1. Características de las redes neuronales

Aunque en justicia se debe reconocer que los modelos artificiales de redes neuronales han realizado variantes que no existen en las redes neuronales biológicas, su desarrollo ha conseguido potencializar enormemente algunas de las características que les dan el lugar preponderante que tienen en todo el universo

¹³ HILERA, José Ramón y MARTÍNEZ, Víctor José. *Redes neuronales artificiales: fundamentos, modelos y aplicaciones*. México D. F : Alfaomega : Ra-ma, 2000

de la inteligencia artificial. Probablemente, de ahí surge que las ventajas de las redes neuronales se asocien a las funciones que comparte con el cerebro humano. De igual forma, las redes neuronales presentan características no deseables en algunas aplicaciones y que desaconsejan su uso en estos casos.

La figura 6 nos muestra un esquema de las principales ventajas y desventajas de las redes neuronales.

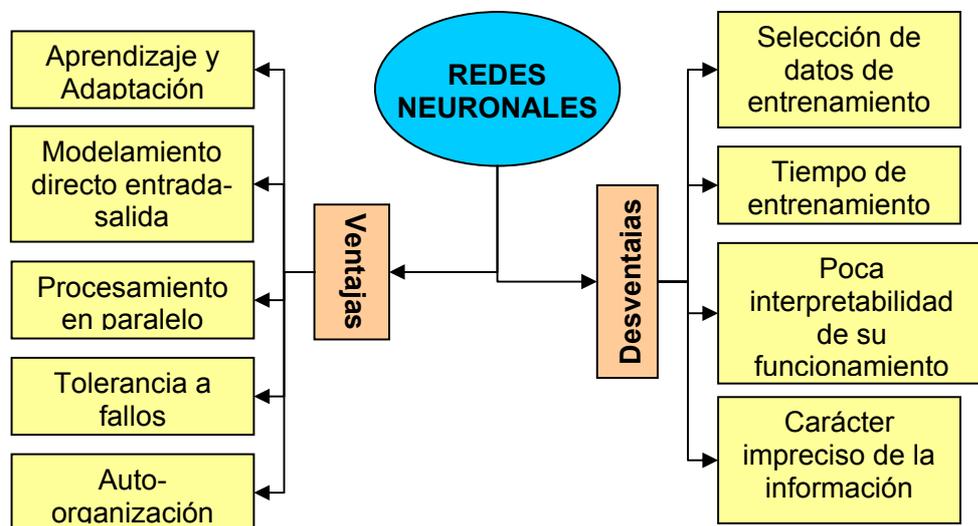


Figura 6. Ventajas y desventajas de las redes neuronales

2.1.2. Aplicabilidad de las redes neuronales

Si bien existe una enorme diversidad de situaciones en las que resulta adecuado el empleo de las redes neuronales, y se pudiera pensar en una aplicabilidad ilimitada de las redes neuronales, se pueden destacar en general, algunas

características que favorecen el uso de de las redes neuronales para abordar un problema específico. Estas son:

- Carencia de un modelo previo o de conocimientos expertos relacionados con el problema
- Gran cantidad de datos o bien datos ruidosos, contradictorios e incompletos
- Adaptabilidad al entorno, por evolución en el tiempo o variación de parámetros que requieran la reconfiguración parcial o total.
- Tiempos de respuesta críticos
- Evidencia de una dinámica significativamente no lineal.

2.2. CONCEPTOS FUNDAMENTALES

2.2.1. Elementos de una neurona artificial

Una neurona artificial es un elemento de procesamiento similar a una neurona biológica, esto es, capaz de **recibir los estímulos** presentes en sus múltiples entradas, **ponderarlos** de acuerdo con su importancia relativa, y **producir una respuesta** que represente la relación, típicamente no lineal, entre estímulos y respuesta (entradas y salida). Un esquema de una neurona artificial se puede observar en la figura 7.

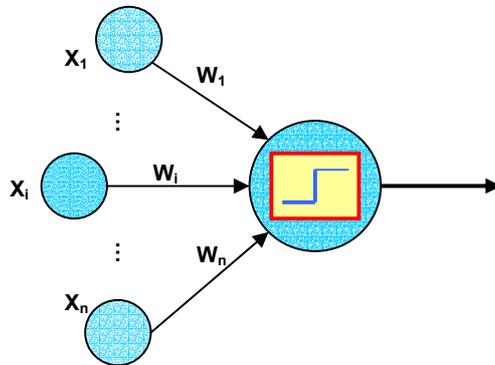


Figura 7. Esquema de una neurona artificial

Con base en estas tareas podemos definir tres elementos principales que conforman la neurona artificial:

- **Pesos relativos de cada conexión:** son valores típicamente en el intervalo $[-1,1]$ que representan la importancia relativa de un estímulo (entrada), multiplicando su valor actual para luego sumarlas
- **Estado de activación:** indica el valor actual de las señales que recibe la neurona, como la suma ponderada de estas y sus pesos asociados. Comúnmente se asocian unos umbrales de activación que indican un valor preestablecido que debe superar la suma neta de entradas para poder activar la neurona.
- **Función de salida o de transferencia:** es quien determina el valor que toma la salida de la neurona dado un valor en su estado de activación. Típicamente es una función no lineal que representa matemáticamente la relación entrada-salida. Las formas más comunes son las de tipo escalón, tipo lineal y tipo sigmoideal.

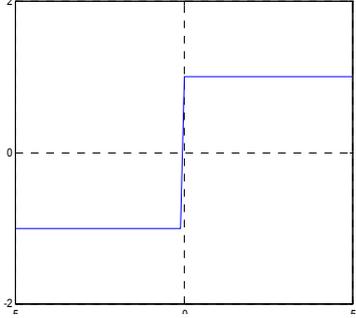
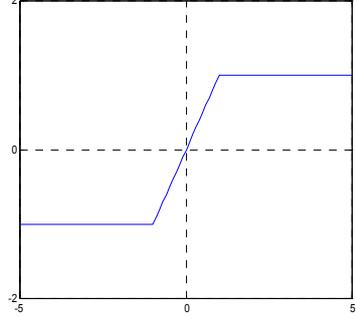
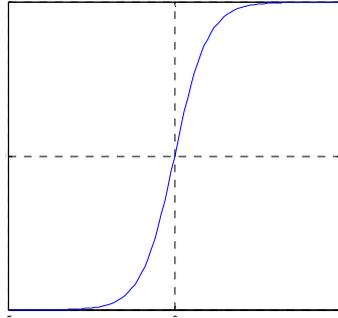
 <p style="text-align: center;">a)</p>	 <p style="text-align: center;">b)</p>	 <p style="text-align: center;">c)</p>
$f(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases}$ $f(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ -1 & \text{si } x < 0 \end{cases}$	$f(x) = \begin{cases} 0 & \text{si } x < -c \\ 1 & \text{si } x > c \\ a & \text{en otro caso} \end{cases}$ $f(x) = \begin{cases} 0 & \text{si } x < -c \\ 1 & \text{si } x > c \\ \frac{x}{2c} + \frac{1}{2} & \text{en otro caso} \end{cases}$	$f(x) = \frac{1}{1 + e^{-ax}}$

Tabla 3. Tipos comunes de funciones de transferencia

a) Tipo Escalón; b) Tipo Lineal; c) Tipo Sigmoidal

2.2.2. Estructura de una red neuronal artificial

Una red neuronal artificial es la unión de varias neuronas artificiales con una determinada estructura y tipo de enlaces. En líneas generales, los elementos que definen una arquitectura particular de red neuronal como tal son:

- **Número de procesamiento (neuronas):** aumenta o disminuye la complejidad de la red y su capacidad para manejar ciertos tipos de información

- **Número de niveles o capas de neuronas:** una capa se define como el conjunto de neuronas cuyas entradas provienen de la misma etapa de procesamiento de los datos y cuyas salidas se dirigen hacia otra etapa de procesamiento común. Las capas pueden ser de entrada, de salida u ocultas cuando no toman ni arrojan datos desde ni hacia el exterior
- **Patrón de conectividad entre neuronas:** se refiere a la manera como están dispuestas y conectadas entre sí las neuronas dentro de la red. Una neurona puede tener conexiones hacia delante (hacia una etapa de procesamiento posterior), hacia atrás (hacia una etapa de procesamiento anterior), laterales (hacia neuronas en su misma capa), o incluso auto-recurrentes (su salida se realimenta a su entrada).
- **Sentido de propagación:** este indica la forma como se va generando la información en la red, bien sea desde las entradas (*feedforward*) o desde las salidas (*feedback*)
- **Tipo de funciones de transferencia:** este componente de cada neurona individual es también importante para la red ya que determina ciertas características de la red como su precisión, manejo de datos analógicos o incluso el proceso de aprendizaje
- **Regla de aprendizaje:** es el algoritmo que sigue la red para ajustar sus pesos a medida que recibe la información de entrada.

2.3. APRENDIZAJE EN LAS REDES NEURONALES

Se puede definir el aprendizaje en una red neuronal como el proceso de modificación de sus parámetros (principalmente pesos de conexión y umbrales de activación) para alcanzar la dinámica deseado. Al modificar estos valores, se pretende que, al propagarse los patrones de entrada, se obtengan en las salidas otros patrones esperados o congruentes con el propósito de la red.

Podemos hablar de dos clases generales de aprendizaje:

- Aprendizaje supervisado
- Aprendizaje no supervisado

2.3.1. Aprendizaje supervisado

Este se caracteriza por la presencia de un tutor (*master*) que asocia los patrones de entradas con patrones de salida predefinidos. Para cualquiera de estos patrones, la red debe modificarse una y otra vez hasta que produzca la salida deseada, o en su defecto, el error se encuentre dentro de un límite aceptable definido por el diseñador. Una red con aprendizaje supervisado puede utilizarse para aproximar cualquier función no lineal (del tipo caja negra) siempre y cuando tenga el suficiente número de neuronas ocultas. Hay tres tipos de aprendizajes supervisados: aprendizaje por corrección de error, por refuerzo, o estocástico

2.3.2. Aprendizaje no supervisado

En este esquema de aprendizaje la red no tiene un tutor que asocie las entradas con las salidas, sino que por su estructura misma y algoritmo de adaptación de pesos, ella logra asociar las entradas con un conjunto de salidas correspondientes a clases o categorías que representan rasgos comunes o relaciones subyacentes entre ciertos conjuntos de entradas. Las redes que emplean este tipo de aprendizaje pueden utilizarse para agrupar un conjunto de patrones de entrada dentro de ciertas categorías (*clusters*). Estos algoritmos pueden trabajar tanto off-line u on-line. Se tienen dos tipos: aprendizaje hebbiano, y aprendizaje competitivo o cooperativo.

SUPERVISADO	Corrección de error OFF-LINE	Basado en la evaluación de una señal de error (salida actual menos salida esperada)
	Refuerzo ON-LINE	Basado en una función que evalúa la efectividad de los cambios realizados y realimenta un estímulo positivo o negativo
	Estocástico OFF-LINE	Pueden basarse en corrección de error o refuerzo pero buscan los pesos de una forma no determinística
NO SUPERVISADO	Hebbiano OFF-LINE / ON-LINE	Basado en la familiaridad “natural” entre los datos de entrada
	Competitivo/ cooperativo OFF-LINE / ON-LINE	Basado en conexiones laterales entre neuronas y compiten o cooperan para activarse

Tabla 4. Tipos de aprendizaje para redes neuronales

2.3.3. Aprendizaje *off-line* y aprendizaje *on-line*

Otro criterio para clasificar los algoritmos de aprendizaje de las redes neuronales es la forma en que se usa la información de entrenamiento (patrones). Según esto podemos hablar de:

- **Aprendizaje Off-line (batch):** la adaptación de parámetros de la red se realiza tomando el conjunto total de patrones de entrenamiento **antes** de su etapa de operación. Cada iteración del algoritmo de aprendizaje usualmente recibe el nombre de época (*epoch*). Aunque proporciona menores desviaciones en el comportamiento deseado de la red durante su etapa de operación, usualmente se torna problemático con muchos datos.
- **Aprendizaje On-line (recursivo):** la adaptación de parámetros de la red se realiza a la misma vez que la red los va procesando **durante** su etapa de operación. Con este tipo de aprendizaje la red puede aprender de mejor forma al enfrentarse con los datos del mundo real, pero al ir cambiando sus pesos puede presentar variaciones no deseadas que le restan fiabilidad al sistema.
- **Métodos intermedios:** aunque no se menciona frecuentemente, existe un tipo de aprendizaje en el cual los pesos no se modifican off-line pero tampoco lo hacen constantemente, sino cada vez que pasan por la red cierto número de patrones (*epoch size*).

2.4. ARQUITECTURAS DE REDES NEURONALES

El estudio e investigación acerca de las redes neuronales artificiales ha originado una enorme diversidad de arquitecturas de redes neuronales cuya mera descripción resultaría demasiado extensa para nuestro objeto de estudio. En este apartado nos limitaremos a exponer aquellas de interés para la comprensión de los sistemas neuro difusos.

2.4.1. Perceptrón monocapa

Esta red neuronal más simple solo contiene una neurona de procesamiento. A pesar de su sencillez es capaz de clasificar correctamente unos patrones determinados en una de varias clases, y además responde de manera congruente ante patrones aleatorios. La regla de decisión es responder +1 si el patrón presentado pertenece a la clase A, o -1 si el patrón pertenece a la clase B, la salida depende de la entrada neta.

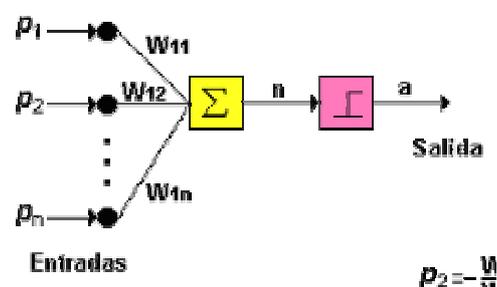


Figura 8. Diagrama de una red perceptrón monocapa

En operación, la única neurona del perceptrón calcula la suma neta de todas las señales de entrada ponderadas, y luego compara este valor con un umbral preestablecido para determinar la salida de la única neurona de salida del perceptrón.

La principal desventaja de la red perceptrón monocapa es su incapacidad para solucionar problemas que no sean linealmente separables

2.4.2. Perceptrón multicapa

El perceptrón multicapa es una estructura jerárquica que consiste en varias capas de neuronas totalmente interconectadas con alimentación hacia delante, que admiten como entradas las salidas de los elementos de proceso (neuronas) de la capa anterior. Esta red permite regiones de decisión mucho más complejas que las de los dos semiplanos como lo hace el perceptrón de una sola capa.

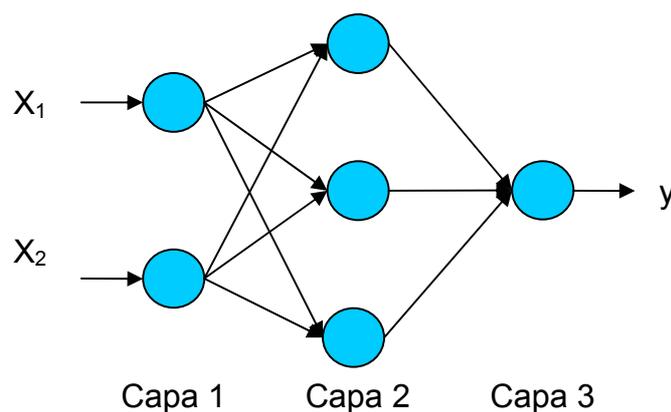


Figura 9. Estructura de una red perceptrón multicapa.

En las redes perceptrón multicapa se distinguen tres tipos de capas como se ilustra en la figura 9:

- **Capa de entrada:** La capa esta formada por el número de entradas externas que se limitan a distribuir las señales de entrada a la capa siguiente.
- **Capas ocultas:** Las capas ocultas están formadas por neuronas que no tienen contacto físico con el exterior.
- **Capa de salida:** Está formada por el número de salidas externas, cuyas salidas constituyen el vector de salidas externas del perceptrón multicapa

Estas redes tienen capacidad de representación funcional universal. Además genera buenas representaciones internas de las características de los datos de entrada, y es ampliamente estudiada y aplicada en la práctica. Sin embargo, tienen la desventaja de requerir un elevado tiempo de aprendizaje para estructuras complejas. La tabla 5 presenta algunas de las aplicaciones mas importantes de las redes perceptrón multicapa.

APLICACIONES DE LAS REDES NEURONALES TIPO PERCEPTRÓN MULTICAPA	
✓ Compresión de datos	✓ Aproximación de funciones
✓ Control adaptable	✓ Reconocimiento de patrones
✓ Segmentación de imágenes y señales	✓ Filtrado de señales
	✓ Eliminación de ruido

Tabla 5. Aplicaciones de las redes neuronales tipo Perceptrón multicapa

2.4.3. Redes de funciones de base radiales

Las redes de funciones de base radiales RBFN (*Radial Basis Function Network*) están constituidas por dos capas. La capa oculta esta compuesta por n unidades radiales totalmente conectadas al vector de entrada. Las funciones de transferencia de la capa oculta son similares a una función de densidad gaussiana.

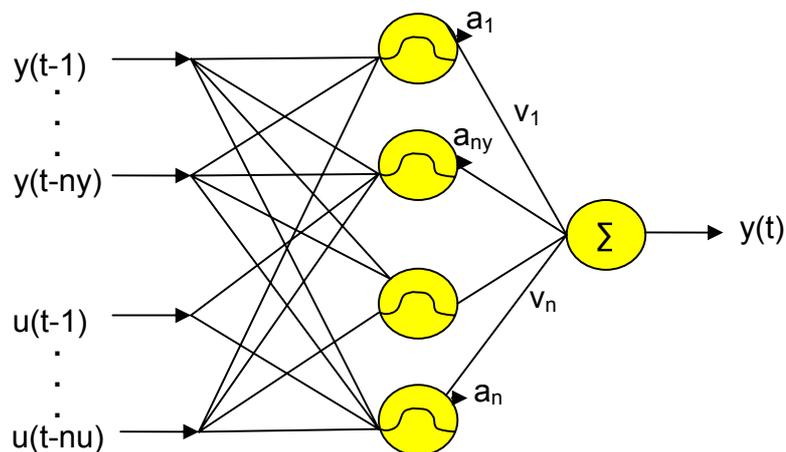


Figura 10. Estructura de una red de función de base radial

Estas redes sobresalen por su capacidad universal de representación funcional. Además, la estructura de esta red tiene interpretación directa, lo que permite realizar una buena inicialización de los pesos de la red, y extraer conocimiento de las estructuras ajustadas. A su vez, la buena inicialización de los pesos acelera el proceso de aprendizaje. La contraparte de esta red es que el procesamiento realizado es algo más complejo que en el caso del perceptrón multicapa.

Esta red tiene aplicabilidad especialmente en la aproximación de funciones y en el reconocimiento de patrones, y en consecuencia es uno de los modelos mas presentes en los sistemas neuro difusos, especialmente para los procesos de fuzzificacion y defuzzificacion (para mas detalles ver capitulo 3).

2.4.4. Otros tipos de redes neuronales

A manera de complemento de esta sección se ha incluido la tabla 6 que muestra algunas otras redes neuronales bastante comunes.¹⁴.

Nombre	Creadores	Ventajas	Desventajas	Aplicaciones
ADALINE/ MADALINE	Bernard Widrow; 1960	Implementación rápida y fácil con circuitos analógicos	Solo es posible clasificar espacios linealmente separados	Filtrado de señales, ecualizadores adaptativos, módems
Avalancha	Stephen Grossberg; 1967	Bastante sencilla comparada con las otras que pueden realizar las mismas	No es fácil alterar la velocidad o interpolar el movimiento	Reconocimiento de habla continua, control de brazos de robot.
Back Propagation	Paul Werbos, David Parker y David Rumelhar; 1974	Facilidad de aprendizaje potente, el aprendizaje y la arquitectura son muy simples	Aprendizaje y arquitectura simples	Síntesis de voz desde texto, control de robots, predicción y reconocimiento de patrones.
Brain-	James	Mejora la	Realización	Extracción de

¹⁴ MARTÍN del Brio, Bonifacio y SANZ Molina, Alfredo. *Redes neuronales y sistemas difusos*. México D. F : Alfaomega, 2002

State-in-a-Box	Anderson; 1977	realización del tipo de redes Hopfield.	y potenciales aplicaciones no estudiadas totalmente	conocimiento en bases de datos
Self-Organizing Map SOM	Teuvo Kohonen; 1984	Realiza mapas de características comunes de los datos aprendidos	Requiere mucho entrenamiento para que pueda aprender	Reconocimiento de patrones, codificación de datos, optimización.
Máquinas de Boltzman y Cauchy	Jeffrey Hinton, Terry Sejnowski, Harold Szu; 1986	Simples con gran capacidad de representación óptima de patrones	Necesitan un tiempo muy largo para su aprendizaje	Reconocimiento de patrones (sonar, radar)
Teoría Resonancia Adaptativa ART	Gail Carpenter, Stephen Grossberg; 1986	Sofisticada	Muy sensible a la translación y distorsión.	Reconocimiento de patrones (radar, sonar).

Tabla 6. Otras arquitecturas de redes neuronales

3. SISTEMAS DIFUSOS DE CONTROL

3.1. GENERALIDADES

El propósito de los sistemas de control difusos es realizar procesos que requieran control humano. Los sistemas de control difuso permiten describir el conjunto de reglas que utilizaría un ser humano que controlase el proceso, con toda la imprecisión que poseen los lenguajes naturales y, sólo a partir de esas reglas, generan las acciones que realiza el control. Por esta razón, también se les denomina controladores lingüísticos. Inclusive, el control difuso puede ser usado para reemplazar el pensamiento y la experiencia humana ya que puede ejecutar un razonamiento similar, realizando de esta forma un control inteligente. Para realizar un control difuso, se necesita establecer relaciones SI-ENTONCES. Por medio de estas reglas se puede aprovechar el conocimiento de los operadores, y después de establecidas se puede realizar el control por razonamiento con lógica difusa.

3.1.1. Características de los sistemas basados en lógica difusa

La lógica difusa es un planteamiento que permite evaluar cuantitativamente el valor de verdad/falsedad de una afirmación, en contraste con la descripción cualitativa bi-estable de la lógica clásica. Este manejo cuantitativo también incluye

la agrupación de los valores de una variable dentro de categorías lingüísticas, asociándolas a valores representativos de su grado de pertenencia a una categoría mediante **funciones de membresía**. De una forma sencilla podemos describir las características de la lógica difusa así:

- La lógica difusa está basada en variables lingüísticas, no numéricas
- Es posible y común que existan variaciones no lineales
- El manejo de la certeza de una expresión no es digital (cierto-falso) sino mas bien análogo

Por brevedad hemos omitido en este documento un análisis matemático de la lógica difusa, pero pueden encontrarse análisis mas completos en español en textos como el de Bonifacio Martín de Río¹⁵

3.1.2. Ventajas y desventajas de los sistemas de control difuso

Las características de los sistemas de control difuso pueden resultar en ventajas o desventajas para los problemas a resolver, pero el estudio e implementación extensiva de ellos ha permitido analizarlos con sus ventajas y desventajas.

La figura 11 contiene algunas de las ventajas y desventajas mas notorias de los sistemas difusos de control.

¹⁵ *Ibid* p. 29

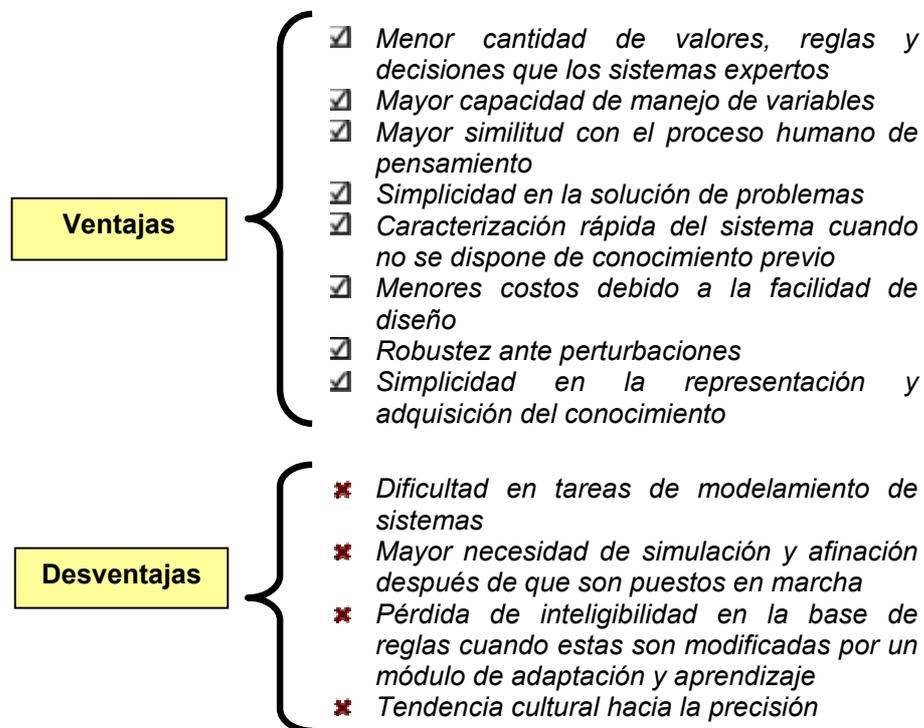


Figura 11. Ventajas y desventajas de los sistemas de control difuso

3.1.3. Aplicabilidad de la lógica difusa a los sistemas de control

La lógica difusa comenzó a utilizarse en controladores por su simplicidad, ya que no requiere de constructores matemáticos complejos, permitiendo en cambio diseñar mediante la descripción del funcionamiento con lenguaje natural y facilitando las tareas de prueba y mantenimiento del sistema.

En líneas generales, los sistemas cuyas características hacen que los sistemas difusos resulten necesarios o beneficiosos son:

- Sistemas complejos donde es difícil o imposible el modelamiento

- Sistemas controlados por el conocimiento de operadores humanos expertos
- Sistemas con entradas y salidas complejas y continuas
- Sistemas que requieren el uso de la observación humana de las entradas o se basan en reglas
- Sistemas con naturaleza indefinida, tal como los que involucren reglas sociales y de comportamiento.

Los sistemas difusos pueden ser usados para estimación, toma de decisiones, y sistemas de control mecánico tal como en sistemas de aire acondicionado, control automovilístico y aún casas inteligentes, como también en el control de procesos industriales y patrón en otras aplicaciones.

3.2. ESTRUCTURA DEL CONTROLADOR DIFUSO

El esquema fundamental de un controlador difuso es el que se observa en la figura 9, donde cada bloque es el encargado de realizar cada uno de los cuatro pasos importantes del proceso de inferencia difusa.

3.2.1. Fuzzificador

Este bloque es el encargado de realizar la normalización y traducción de los valores físicos de las entradas en valores lingüísticos que formen parte del

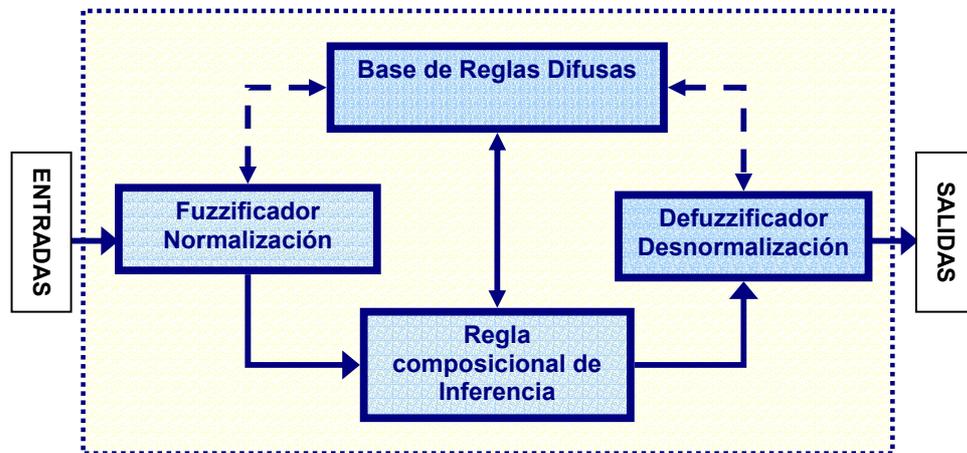


Figura 12. Esquema de un controlador difuso

universo de discurso normalizado. Entre las formas de fuzzificación podemos destacar

- Inferencia basada en una composición
- Inferencia basada en la activación individual de las reglas

3.2.2. Base de reglas difusas

Es el bloque encargado de almacenar el conocimiento en forma de una base de datos y una base de reglas. Su función es representar de manera estructurada los criterios del control experto. En esencia, una base de regla difusa consiste en un conjunto de reglas difusas SI-ENTONCES, es el corazón del sistema difuso en el sentido de que todos los otros componentes son usados para implantar estas reglas de una manera rápida y eficiente.

En los modelos difusos adaptativos, esta base de reglas se modifica periódicamente a través del aprendizaje y adaptación a los patrones de entrada y salida del sistema.

3.2.3. Reglas de inferencia

Su trabajo consiste en evaluar el estado de las condiciones a la entrada del sistema e inferir las acciones que deban tener lugar a la salida del sistema, basado en las contribuciones individuales de cada regla en la base de reglas.

3.2.4. Defuzzificador

El bloque *defuzzificador* convierte el conjunto de valores de salida expresados como valores difusos en un solo valor puntual, y adicionalmente realiza la desnormalización de la salida para que pueda ser aplicable al sistema real. Existen diversos métodos de *fuzzificación*, entre los más comunes tenemos el método del centro de gravedad y el método del centro de las sumas.

- **Método del centro de gravedad o del centroide:** consiste en crear para la salida del sistema una función de pertenencia a un nuevo conjunto obtenido como unión de aquellos a los que pertenece parcialmente el valor de salida. Esta nueva función puede calcularse mediante la suma de las funciones de pertenencia de estos conjuntos, pero multiplicadas aritméticamente por el

grado de pertenencia de la salida al subconjunto que fue calculado en la fase de evaluación de las reglas de control. Matemáticamente se puede expresar:

$$y^o = \frac{\int B^*(y) \cdot y \cdot dy}{\int B^*(y) \cdot dy} \quad (2)$$

Donde $B^* = w_1 B_1 \cup w_2 B_2 \cup w_3 B_3$ es un producto que representa la inferencia difusa gráficamente, esto es, la unión de los productos entre los pesos de las reglas w_i y los conjuntos difusos B_i .

- **Método del centro de las sumas:** La acción neta de control se obtiene mediante la *defuzzificación*, que es una combinación adecuada de las acciones individuales que se derivan de cada regla, el método de *defuzzificación* de la media ponderada, se puede enunciar de la siguiente manera:

$$y^o = \frac{w_1 y_1^o + w_2 y_2^o + w_3 y_3^o}{w_1 + w_2 + w_3} \quad (3)$$

Donde w_i es el peso de cada regla y y_i^o es el centroide de cada conjunto difuso.

3.3. SISTEMAS DIFUSOS COMO MÁQUINA DE INFERENCIA

Presentamos dos sistemas de inferencia difusa que han sido usados extensamente en varias aplicaciones. La diferencia entre estos sistemas de inferencia difusa radican en los consecuentes de sus reglas difusas, y por lo tanto sus procedimientos de agregación y defuzzificación difieren.

3.3.1. Sistema de inferencia difusa tipo Mamdani

La mayoría de los sistemas de inferencia difusa emplean el método de inferencia propuesto por Mamdani en el que el consecuente de la regla esta definido por conjuntos difusos y tiene la siguiente estructura:

$$\text{SI } (x \text{ es } A_1) \text{ Y } (y \text{ es } B_1) \text{ ENTONCES } (z_1 = C) \quad (4)$$

donde A_1 y B_1 son los conjuntos difusos de entrada y C es el conjunto difuso de salida.

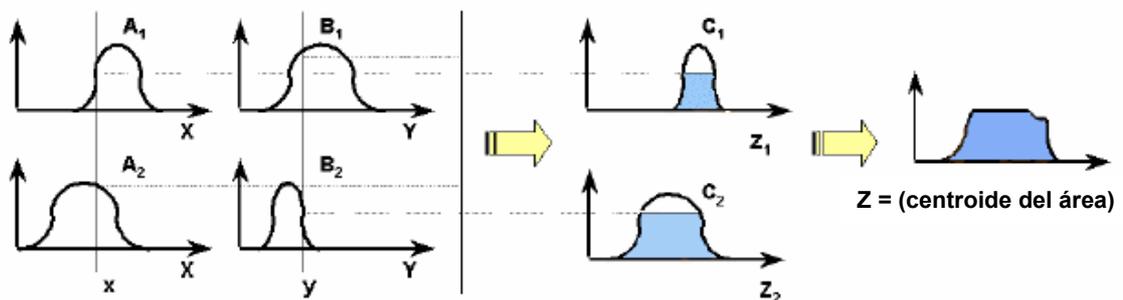


Figura 13. Sistema de inferencia difusa tipo Mamdani

3.3.2. Sistema de inferencia difusa tipo Takagi Sugeno Kang

Takagi, Sugeno y Kang (TSK) propusieron un esquema de inferencia en el que la conclusión (consecuente) de una regla difusa esta constituida por una combinación lineal ponderada de las entradas, en vez de ser un conjunto difuso. Esto se puede expresar de la siguiente forma

$$\text{SI (x es } A_1) \text{ Y (y es } B_1) \text{ ENTONCES (f = } p_1x + q_1y+r) \quad (5)$$

Donde A_1 y B_1 son los conjuntos difusos de entrada, y p_1 , q_1 y r son parámetros lineales que representan la salida en términos de las variables de entrada.

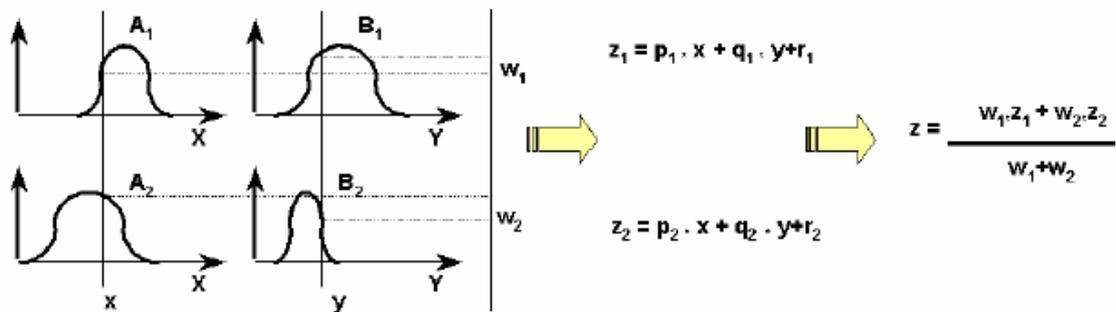


Figura 14. Sistema de inferencia difusa tipo Takagi Sugeno Kang

Los controladores difusos TSK usualmente necesitan un menor número de reglas porque su salida ya es una función lineal de las entradas en vez de un conjunto difuso constante. La base de reglas difusas usualmente se construye manualmente o mediante técnicas de adaptación automática. Agregación es la combinación de las relaciones difusas que representan las reglas difusas en una sola relación difusa. Normalmente se utilizan operadores de norma T para representar los operadores AND e *implicación*, y co-normas T para los operadores OR y *agregación*. En el próximo capítulo se incluye un análisis matemático mas profundo de estas normas y co-normas T.

4. SISTEMAS NEURO DIFUSOS

4.1. GENERALIDADES

Los sistemas neuro difusos son una combinación entre redes neuronales artificiales y controladores difusos, que permite minimizar muchos de los problemas en las aplicaciones de control inteligente. El uso de técnicas híbridas resalta las ventajas y soslaya las desventajas de las técnicas individuales.

En general, estos sistemas derivan su topología de las redes neuronales artificiales (RNA) preservando los nodos que desarrollan una función simple (sigmoide, logística o lineal), pero estas intercalan capas de nodos con funciones internas más complejas (funciones de pertenencia, implicación y concreción en subconjuntos difusos).

4.1.1. Estructura general de un sistema neuro difuso

En líneas generales podemos distinguir tres módulos componentes de un sistema neuro difuso.

- **Módulo agente:** puede ser un sistema de inferencia difusa o un sistema experto (controlador computacional) con lógica difusa,

- **Módulo diligente:** contiene una red neuronal, que continuamente está observando las entradas y salidas del agente y en determinados casos aprende las situaciones nuevas.
- **Módulo traductor:** tiene como tarea extraer de la arquitectura resultante de la red neuronal la información aprendida y expresarla en reglas utilizando predicados vagos para ser incorporada a un controlador.

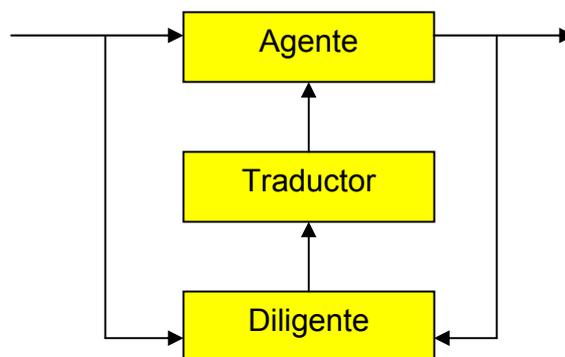


Figura 15. Estructura general de un sistema neuro difuso

4.1.2. Ventajas y desventajas de los sistemas neuro difusos

El uso de técnicas híbridas permite que sobresalgan las ventajas de cada una de las técnicas individuales y a la vez se minimice los efectos desventajosos de cada una. Entre las ventajas y desventajas del uso de los sistemas neuro difusos con respecto a un Sistema Experto (SE) para la implantación de controles difusos pueden destacarse:

- **Capacidad de aprendizaje:** Esta no permite pérdidas en el significado de las reglas originales o no se presentan distorsiones severas en la implantación del sistema después del entrenamiento.
- **Disponibilidad de algoritmos:** La disponibilidad de los algoritmos de entrenamiento supervisado, probados en redes neuronales, pueden usarse con muy pocas modificaciones en los sistemas neuro difusos.
- **Tiempo de inferencia:** Se obtiene menor tiempo frente a la programación implementada en un Sistema Experto (SE), además se procesa en forma paralela, para evitar las arduas búsquedas en árboles de decisión
- **Facilidad de programación:** La facilidad de programación de un sistema neuro difuso con respecto a la programación de un Sistema Experto.

La principal desventaja es su capacidad limitada de explicación cuando se compara con la que tienen los programas de SE.

4.1.3. Importancia de los sistemas neuro difusos en el campo del control

La creciente demanda en tecnología de nuestra sociedad requiere nuevos enfoques para los problemas actuales de control. Debido a la complejidad de algunos procesos se tiene una incipiente automatización y una gran carga de mando manual.

Aun cuando las tareas de control que se asignen a un controlador humano se

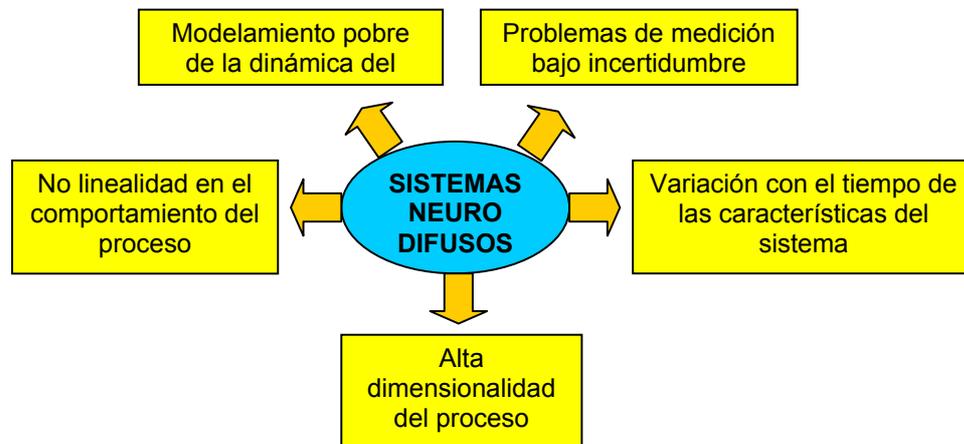


Figura 16. Importancia de los sistemas neuro difusos en el campo del control realicen de manera adecuada, su desempeño exitoso depende del grado de especialización que alcanzan los operadores, su estado de ánimo y de los distintos operarios. Por tanto se hace necesario proveer controles automáticos adecuados. Por esta razón se requiere un esquema de control que posea características inteligentes, con el cual sea posible controlar estos procesos en un grado mucho mayor que el actual y con las mismas o mejores prestaciones con las que lo hace el operador humano.

4.1.4. Pasos de desarrollo de sistemas neuro difusos

- **Obtención de los datos de entrenamiento:** Seleccionar el conjunto de datos (pares entrada-salida) que representan el comportamiento deseado del sistema. Para una combinación de variables de entrada, cada conjunto de datos da un valor de salida de ejemplo.

- **Creación del sistema de lógica difusa:** Comenzar con un sistema difuso inicial, bien sea definido por el usuario o por un algoritmo que esboce un diseño de sistema difuso a partir de los datos de entrenamiento. Es importante incluir en este diseño inicial cualquier pieza de información que se tenga de antemano, puesto que el algoritmo podría no extraerla de los datos de entrenamiento si estos son de mala calidad.
- **Definición del aprendizaje neuro difuso:** La adaptación de los sistemas neuro difusos puede ser localizada y no necesariamente generalizada como en las redes neuronales corrientes. Esto implica tener que escoger los componentes del sistema difuso que pueden ser modificados por el algoritmo de aprendizaje. Con esto se pueden ajustar las funciones de membresía o la base de reglas, si no se está seguro de su desempeño.
- **Selección del método de aprendizaje:** Definir la manera como se han de modificar las funciones de membresía y/o reglas del sistema en cada paso del aprendizaje. Este proceso se puede optimizar definiendo parámetros como la tasa de aprendizaje para funciones de membresía y la tasa de aprendizaje para reglas.
- **Fase de entrenamiento:** Aplicar el algoritmo de aprendizaje al sistema para modificar las funciones de membresía y/o reglas hasta que el error sea mínimo dentro de un criterio previamente establecido.

- **Optimización y verificación:** Realizar algunas ligeras modificaciones por medio de analizadores lógicos, herramientas de desarrollo de software y hardware, y métodos numéricos para optimizar el sistema, verificar la operación correcta del sistema. En general, el código generado para un sistema neuro difuso es mucho eficiente que la solución por medio de redes neuronales.

4.2. APRENDIZAJE EN SISTEMAS NEURO DIFUSOS

Un problema en el proceso de aprendizaje de los sistemas neuro difusos, es que las funciones de activación de las neuronas no son derivables porque utilizan operaciones matemáticas como son mínimo y máximo. Para resolver estos problemas algunas herramientas de desarrollo Neuro difuso utilizan métodos de inferencia extendidos para la lógica difusa. La metodología más común es utilizar memorias *difusas* asociativas FAMs (*Fuzzy Associative Memory*). En una implementación simple una FAM es una regla de lógica difusa con un peso asociado. Matemáticamente se puede asignar FAMs a neuronas en una red neuronal. Esto permite la utilización de un algoritmo de *backpropagation* modificado en los sistemas neuro difusos.

4.2.1. Generación neuronal de reglas de control difuso

Considérese un bloque de reglas difusas de la forma

$$\mathfrak{R}_i : \text{Si } x \text{ es } A_i, \text{ entonces } y \text{ es } B_i \quad (6)$$

Donde A_i y B_i son conjuntos difusos de las particiones de entrada y salida. Cada regla puede ser interpretada como un patrón de entrenamiento para una red neuronal multicapa, en el que la parte del antecedente de la regla representa la entrada, y el consecuente representa la salida deseada. Si se tiene un número n de reglas, el conjunto de patrones de entrenamiento puede ser representado como

$$\{(A_1, B_1), \dots, (A_n, B_n)\} \quad (7)$$

De igual manera para un sistema MISO de dos entradas, las reglas difusas tendrán la forma

$$\mathfrak{R}_i : \text{Si } x \text{ es } A_i \text{ y } y \text{ es } B_i, \text{ entonces } z \text{ es } C_i \quad (8)$$

Y el conjunto total de reglas del sistema (base de reglas) será la concatenación de términos de la forma

$$\{(A_i, B_i), C_i\}, \quad 1 \leq i \leq n \quad (9)$$

Las redes neuronales con aprendizaje no supervisado permiten establecer clases o categorías de datos (*clusterización*), por lo cual un sistema neuro difuso puede aprender por si mismo las reglas de control que lo rigen. Para esto es necesario disponer de un gran número de datos, a partir de los cuales la red pueda establecer las relaciones y categorías, que en este caso son las reglas de control.

Cuando la red aprende todas las parejas, esto es, ha establecido a qué clase (regla) pertenece cada una, almacena los centroides (representantes de cada clase) en sus pesos.

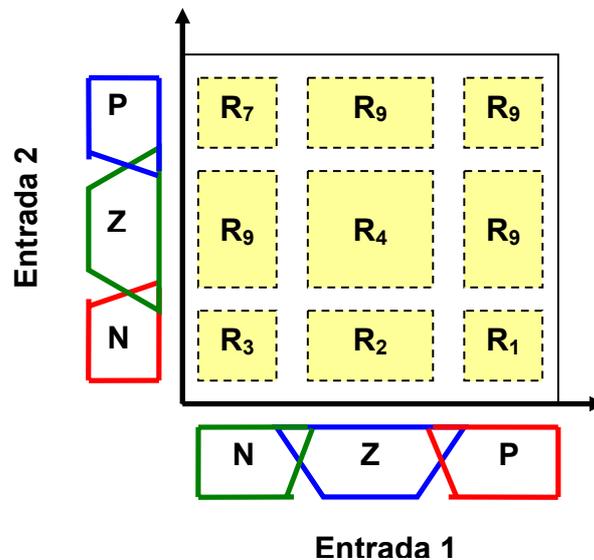


Figura 17. Representación de reglas para un espacio de dos entradas

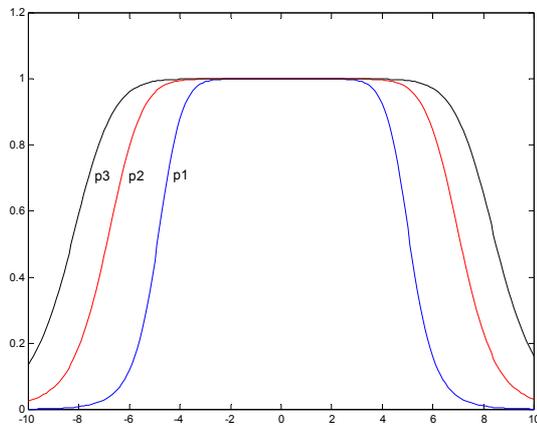
4.2.2. Ajuste neuronal de funciones de membresía

Para ajustar las funciones de pertenencia de un sistema difuso se podrían caracterizar las mismas mediante parámetros y utilizar un algoritmo de aprendizaje supervisado, que permita obtener un modelo de la función de pertenencia con una forma previamente establecida. Por ejemplo, una función de membresía del tipo campana generalizada (también se conoce como trapecio suave) se puede expresar en función de tres parámetros p , q y r de la forma:

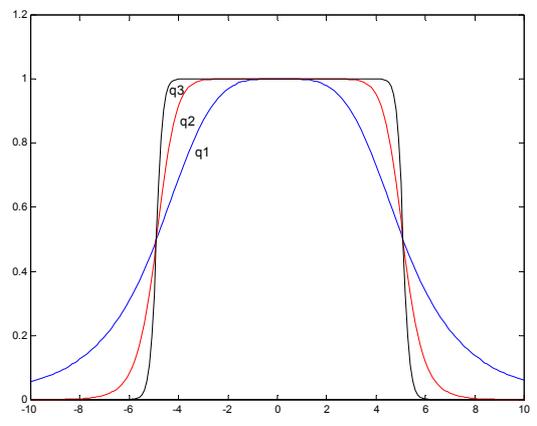
$$f(x, p, q, r) = \frac{1}{1 + \left| \frac{x-r}{p} \right|^{2q}} \quad (10)$$

En la figura 18 podemos ver como varían las funciones de membresía cuando se altera uno cualquiera de los parámetros.

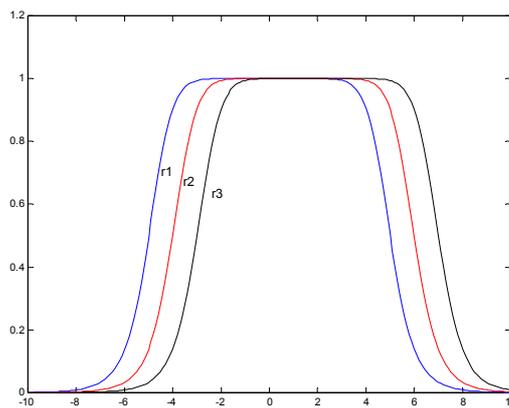
Para este proceso de ajuste también se puede utilizar aprendizaje no supervisado, para que el propio sistema establezca la forma de dicha función. Para esto se utiliza una red con tantas neuronas de entrada como entradas tenga el sistema, y tantas neuronas de salida como conjuntos difusos a los que pueden pertenecer las entradas, la cual se entrena presentando diferentes valores de entrada que activen las salidas correspondientes a los conjuntos a los que pertenece cada valor (1 si pertenece al conjunto y 0 si no es así). En esencia, esta red genera valores binarios (equivalentes a conjuntos *crisp*), pero, como posee capas y pesos ocultos, la red también ha establecido unas formas concretas para las funciones de pertenencia, de tal forma que ante un valor de entrada distinto genera como salida un conjunto de valores entre 0 y 1 (equivalentes a conjuntos difusos) que representan el grado de pertenencia a cada conjunto.



a) $p_3 > p_2 > p_1$



b) $q_3 > q_2 > q_1$



c) $r_3 > r_2 > r_1$

Figura 18. Efecto de la variación de los parámetros de una función de membresía tipo campana a) Variación del parámetro p ; b) Variación del parámetro q ; c) Variación del parámetro r

4.2.3. Ajuste de reglas del proceso de inferencia

Como las proposiciones difusas son interpretadas como relaciones difusas la pregunta clave es cómo interpretar la operación SI-ENTONCES. En cálculo proposicional clásico, Si p ENTONCES q se escribe como $p \rightarrow q$ donde la tabla de verdad de esta función lógica es la que se muestra en la tercera columna de la tabla, donde p y q son variables proposicionales cuyos valores son verdaderos (V) o falsos (F).

p	q	$p \rightarrow q$	$\sim p$	$\sim p \vee q$	$p \wedge q$	$(p \wedge q) \vee \sim p$
V	V	V	F	V	V	V
V	F	F	F	F	F	F
F	V	V	V	V	F	V
F	F	V	V	V	F	V

Tabla 7. Tabla de verdad para la función lógica implicación y sus equivalentes

Aquí podemos comprobar que las funciones lógicas

$$\sim p \vee q \tag{11}$$

y

$$(p \wedge q) \vee \sim p \tag{12}$$

son equivalentes en el sentido de que comparten la misma tabla de verdad.

Las reglas difusas SI-ENTONCES pueden ser vistas como reglas en lógica bivaluada (dos valores de verdad: verdadero o falso) reemplazando p y q con proposiciones difusas, podemos interpretar las reglas difusas SI-ENTONCES reemplazando los operadores \sim , \vee y \wedge con complemento difuso, unión difusa e intersección difusa respectivamente. Como hay una gran variedad de complementos difusos, uniones difusas e intersecciones difusas en la literatura se proponen un número distinto de reglas difusas SI-ENTONCES. Comúnmente se refieren como normas y conormas triangulares (normas-t y conormas-t). Algunas de las normas y conormas mas usadas son las que se encuentran en la tabla 8.

Normas	Mínimo	Producto algebraico	Producto limitado	Producto drástico
<i>Notación</i>	$T_{\min}(a,b)$	$T_{alg}(a,b)$	$T_{lim}(a,b)$	$T_{dras}(a,b)$
<i>Descripción</i>	$\min(a,b)$	ab	$0 \vee (a+b-1)$	$\begin{cases} a, & \text{si } b = 1 \\ b, & \text{si } a = 1 \\ 0, & \text{si } a, b < 1 \end{cases}$

Conormas	Máximo	Suma algebraica	Suma limitada	Suma drástica
<i>Notación</i>	$S_{\max}(a,b)$	$S_{alg}(a,b)$	$S_{lim}(a,b)$	$S_{dras}(a,b)$
<i>Descripción</i>	$\max(a,b)$	$a+b-ab$	$1 \wedge (a+b)$	$\begin{cases} a, & \text{si } b = 0 \\ b, & \text{si } a = 0 \\ 0, & \text{si } a, b > 1 \end{cases}$

Tabla 8. Definición de algunas normas y conormas triangulares

4.3. COMPARACIÓN DE CARACTERÍSTICAS ENTRE LAS REDES NEURONALES Y LA LÓGICA DIFUSA

A continuación se incluye un cuadro comparativo en líneas generales entre las redes neuronales y los sistemas de lógica difusa.

	Redes Neuronales	Sistemas de Lógica Difusa
Capacidad de procesamiento	Procesamiento completamente en paralelo	Limitado por la capacidad del procesador (convencional o difuso)
Naturaleza del procesamiento	Asociación de pares entrada-salida	Énfasis en el proceso de inferencia lógica
Capacidad de interpolación	Muy buena gracias a su naturaleza	Muy buena gracias a las categorías lingüísticas
Adquisición del conocimiento	Basado en el proceso de aprendizaje (conocimiento <i>a posteriori</i>)	Base de reglas previas del experto (conocimiento <i>a priori</i>)
Naturaleza de las operaciones	Productos vectoriales y funciones de transferencia derivables	Operadores de lógica difusa y funciones no derivables
Tolerancia a fallas	Tolerancia estructural (redes redundantes)	Tolerancia intrínseca (etiquetas lingüísticas)

Artículo I. **Tabla 9. Comparación entre Redes Neuronales y Sistemas de Lógica Difusa**

4.4. MODELOS COMBINADOS DE REDES NEURONALES Y SISTEMAS DIFUSOS

El estudio y aplicaciones de las diferentes combinaciones de redes neuronales y sistemas difusos se ha extendido mucho en los últimos años, y por lo mismo sería imposible e inadecuado pretender abarcar en este documento la totalidad del mismo.

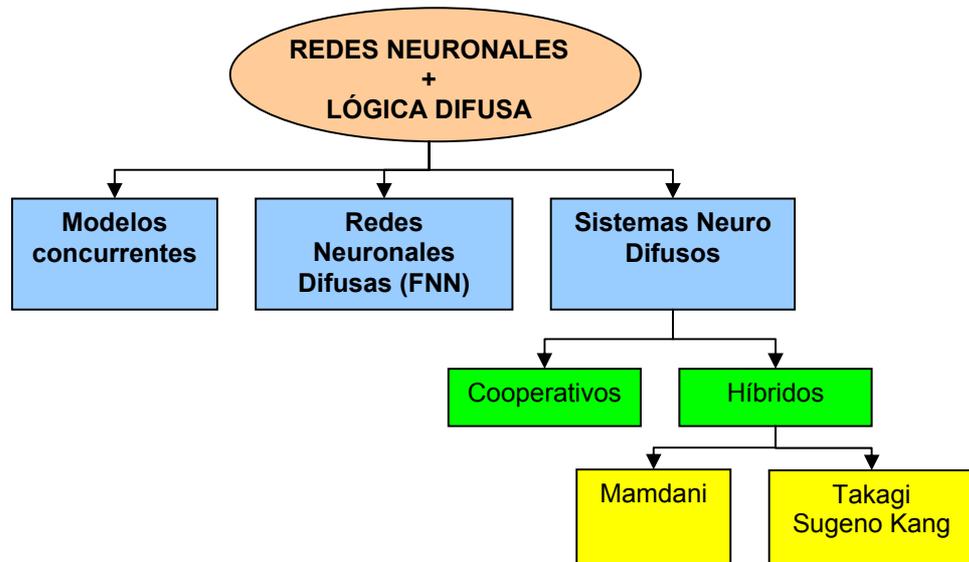


Figura 19. Combinaciones de modelos de redes neuronales y sistemas difusos

Para mayor claridad y por su aplicabilidad, nuestro interés se ha de centrar en los sistemas neuro difusos aplicables al control de procesos. En general, este extenso abanico de combinaciones se puede clasificar de la siguiente manera¹⁶:

- Modelos neuro difusos concurrentes
- Redes neuronales difusas (FNN, *Fuzzy Neural Networks*)
- Sistemas neuro difusos (NFS, *Neuro Fuzzy Systems*)

4.4.1. Modelos neuro difusos concurrentes

Estos modelos no son realmente modelos neuro difusos en el sentido de que, aunque existan redes neuronales y sistemas difusos enfocados hacia una misma labor, no existe apenas interacción entre ambos, esto es, ningún sistema es usado para determinar los parámetros del otro. Usualmente, la red neuronal solamente realiza un procesamiento previo o posterior de los datos que ingresan al sistema de control difuso.

4.4.2. Redes neuronales difusas

Esta combinación es por su naturaleza una red neuronal artificial, sólo que en ella se utilizan algunos métodos de la lógica difusa (por ejemplo, operadores difusos)

¹⁶ NAUCK, Detlef. *Beyond NeuroFuzzy: Perspectives and Directions (1995)*. University of Braunschweig. Disponible en la Web:

con el fin de mejorar su capacidad de aprendizaje o su desempeño. Estas variaciones pueden ser el uso de reglas difusas para cambiar la tasa de aprendizaje (velocidad con la que se modifican los pesos durante el proceso de aprendizaje), o trabajar con entradas de naturaleza difusa.

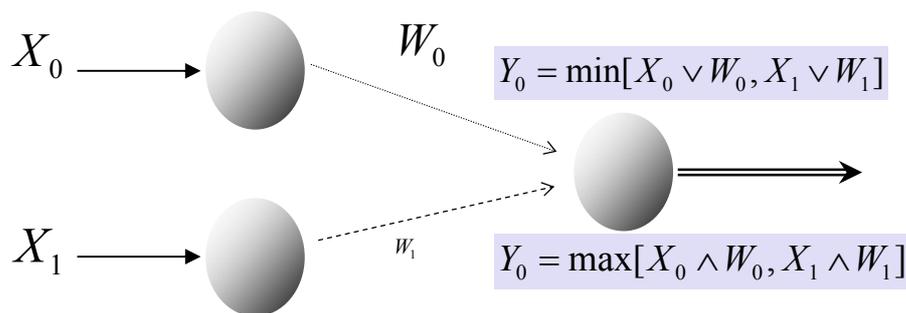


Figura 20. Esquema de una red neuronal difusa

Una clasificación de las redes neuronales difusas (FNN, *Fuzzy Neural Networks*) puede hacerse como sigue:

- Redes con señales de entrada reales y pesos difusos (FNN1)
- Redes con señales difusas y pesos reales (FNN2)
- Redes con señales y pesos difusos (FNN3)

Estrictamente hablando, las redes neuronales difusas no son sistemas neuro difusos, así que tampoco serán de nuestro interés en este documento.

4.4.3. Sistemas neuro difusos

La combinación de redes neuronales y sistemas difusos en el que se aprovecha la capacidad de aprendizaje de la red para modificar los parámetros del sistema

difuso es lo que realmente constituye un sistema difuso. Y aunque se podría simplificar esto diciendo que es una red neuronal que modifica los parámetros de un sistema difuso, existen sistemas neuro difusos en los que ambos componentes no se pueden diferenciar entre sí con claridad. Con base en esta claridad para distinguir o no la red neuronal del sistema difuso, una vez mas podemos subdividir los sistemas neuro difusos en cooperativos e híbridos.

4.4.3.1. Sistemas neuro difusos cooperativos

En estos sistemas la red neuronal se utiliza para determinar los parámetros de un sistema difuso. Después de la fase de entrenamiento, el sistema difuso trabaja sin la red neuronal. La literatura científica¹⁷ ha identificado 4 clases de sistemas neuro difusos cooperativos, según los parámetros que modifican y según los el tipo de aprendizaje se realice *on-line* u *off-line*. La figura ilustra estas 4 clases.

a. Sistemas neuro difusos que modifican las funciones de membresía

(off-line): La red neuronal tiene la libertad para modificar los parámetros de las funciones de membresía de los sistemas difusos, pero no puede alterar de ninguna forma las reglas de inferencia de los mismos. El ajuste solo se realiza antes de la operación final (off-line) con datos de entrenamiento.

¹⁷ NAUCK, Detlef y KRUSE, Rudolf. *Choosing Appropriate NeuroFuzzy Models (1994)*. University of Braunschweig. Disponible en la Web:

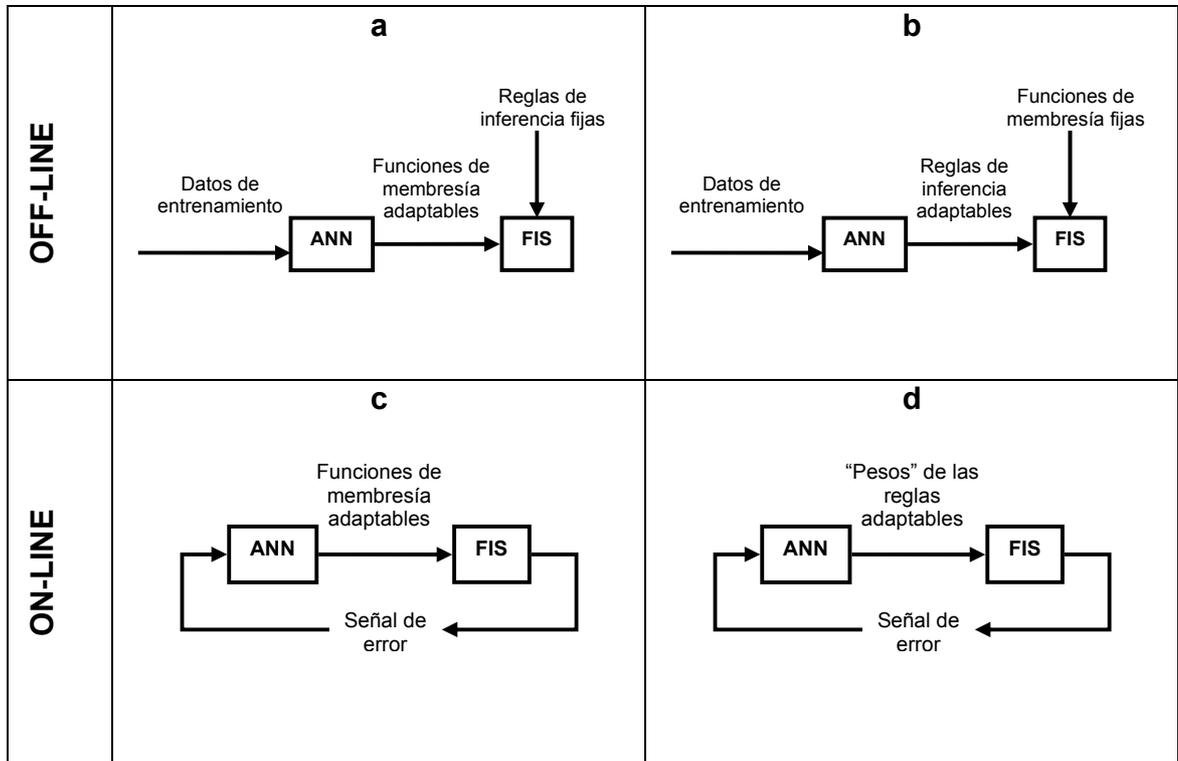


Figura 21. Sistemas neuro difusos cooperativos

- b. Sistemas neuro difusos que modifican las reglas de inferencia (off-line):** La red neuronal realiza un ajuste off-line de las reglas de inferencia del sistema difuso pero no puede modificar las funciones de pertenencia a cada uno de los conjuntos difusos.
- c. Sistemas neuro difusos que modifican las funciones de membresía (on-line):** El ajuste de las funciones de membresía se realiza en la etapa de operación del sistema difuso. Estos sistemas requieren de la evaluación de algún tipo de señal de error para el proceso de aprendizaje en la red neuronal.

d. Sistemas neuro difusos que modifican los “pesos” de las reglas: Este tipo de sistemas realiza el ajuste del sistema difuso mediante “pesos” que equivalen a la importancia de una regla. Por tanto, este no es un sistema difuso en propiedad. Cabe anotarse que este sistema puede obtener el mejor desempeño para tareas de control pero a costa de problemas en la interpretabilidad de las reglas difusas.

4.4.3.2. Sistemas neuro difusos híbridos

Como su nombre lo indica, estos sistemas se caracterizan básicamente por tener una estructura híbrida entre las redes neuronales y los sistemas difusos. De esta manera pueden ser interpretados como una red neuronal especial con parámetros difusos, o como un sistema difuso implementado en forma paralela y distribuida. Una forma común de representarlos es como una red neuronal de múltiples capas. De acuerdo con lo que vimos en el capítulo sobre Lógica Difusa, existen dos esquemas generales de sistemas neuro difusos híbridos, cuyas características varían para dar origen a los sistemas neuro difusos que están en desarrollo.

- **Sistema neuro difuso híbrido tipo Mamdani**

Este sistema se puede representar como una red neuronal de 5 capas, las cuales son:

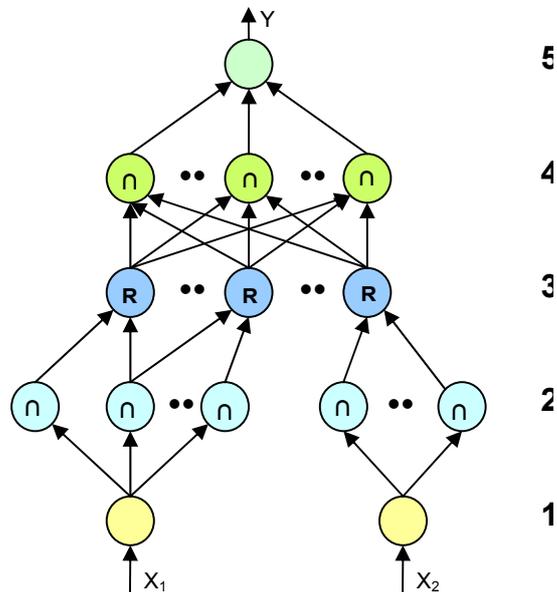


Figura 22. Arquitectura de un sistema neuro difuso tipo Mamdani

- **Capa1 (Entradas):** no realizan ningún tipo de procesamiento sobre los valores de entrada de las variables, sino que simplemente se encargan de distribuir estos valores a la capa siguiente.
- **Capa 2 (Fuzzificación):** cada nodo corresponde a una etiqueta lingüística (frío, caliente, etc) para cada una de las variables de entrada. En otras palabras, la salida representa el grado de pertenencia $\mu_i(x)$ de un valor en la entrada x a un conjunto difuso i , típicamente en el intervalo $[0,1]$.
Matemáticamente

$$O_i^1 = \mu_{A_i}(x), \quad i = 1, 2, \dots \quad (13)$$

Cada nodo puede ser un nodo sencillo que representa una función de membresía simple, o compuesto de nodos multi-capas que calculen una función de membresía compleja.

- **Capa 3 (Antecedentes de reglas):** los nodos de esta capa evalúan los antecedentes de las reglas, es decir los valores que indican el cumplimiento de cada una de las reglas. Normalmente, la salida de estos nodos es la evaluación de una norma T. Por ejemplo, si la regla 1 evalúa las proposiciones difusas simples

$$(x \text{ es } A) \text{ y } (y \text{ es } B) \quad (14)$$

Entonces la fuerza de esa regla (salida de ese nodo) se evalúa como

$$O_i^2 = t[\mu_A(x), \mu_B(y)] \quad (15)$$

- **Capa 4 (Consecuentes):** en esta capa se combina los antecedentes de las reglas y determina el grado al que pertenecen a la salida. El numero de nodos de esta capa es igual al numero de reglas
- **Capa 5 (Combinación/defuzzificacion):** combina los consecuentes de las reglas utilizando operadores de co-norma T y calcula la salida final del sistema difuso después de la defuzzificacion.

- **Sistema neuro difuso híbrido tipo Takagi Sugeno Kang**

En este modelo, las capas 1,2 y 3 tienen las mismas funciones que en los sistemas neuro difusos tipo Mamdani.

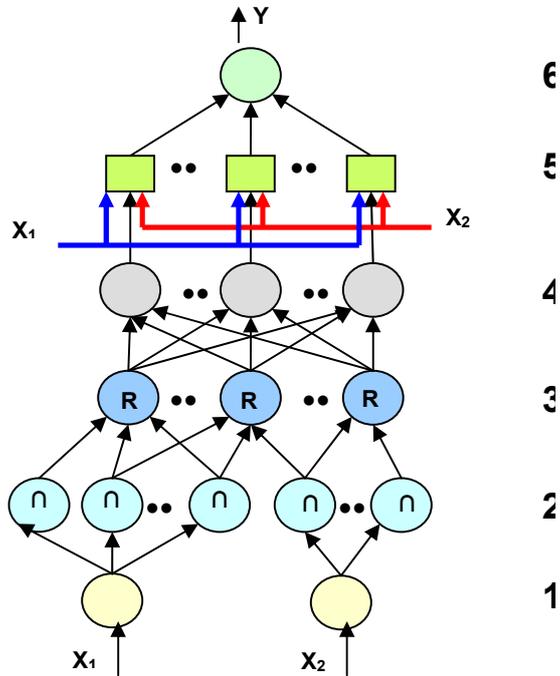


Figura 23. Arquitectura de un sistema neuro difuso tipo Takagi Sugeno Kang

El funcionamiento de las otras capas es el siguiente:

- **Capa 4 (Normalización de las fuerzas de las reglas):** las fuerzas de las reglas individuales se dividen entre la suma de las fuerzas de todas las reglas. Las salidas de los nodos de esta capa vienen dadas por

$$O_i^3 = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2, \dots \quad (16)$$

- **Capa 5 (Consecuentes de reglas):** cada nodo en esta capa tiene una función de la forma

$$O_i^4 = \bar{w}_i f_i = \bar{w}_i (p_i x_1 + q_i x_2 + r_i) \quad (17)$$

Donde w_i es la salida de la capa 4, y p_i , q_i y r_i es el conjunto de parámetros de la transformación lineal entre entradas y salidas.

- **Capa de inferencia de reglas:** calcula la salida total como la suma de las señales de entrada (multiplicadas por sus pesos)

$$O_i^5 = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (18)$$

4.5. PROTOTIPOS DE SISTEMAS NEURO DIFUSOS HÍBRIDOS

La siguiente exposición tendrá como objetivo presentar algunos de los datos mas relevantes acerca de los sistemas neuro difusos híbridos que se han desarrollado hasta la fecha y sus principales aplicaciones, aunque se intentará enfatizar en aquellos sistemas aplicables al control automático.

4.5.1. FALCON

FALCON es el acrónimo para *Fuzzy Adaptive Learning Control Network* (Red Difusa Adaptativa de Control de Aprendizaje), un modelo que fue propuesto por Lin y Lee¹⁸. Su estructura equivale a un FIS de tipo Mamdani, por consiguiente, contiene las 5 capas de este modelo (ver figura). La única variante con el modelo

¹⁸ LIN, C T y LEE, C S G. *Neural Network based Fuzzy Logic Control and Decision System*. IEEE Transactions on Comput. (40(12): pp. 1320-1336, 1991.

original es que posee 2 nodos por cada variable de salida en la capa de salida: uno para los datos de entrenamiento (y_m) y otro para los datos reales (y'_m).

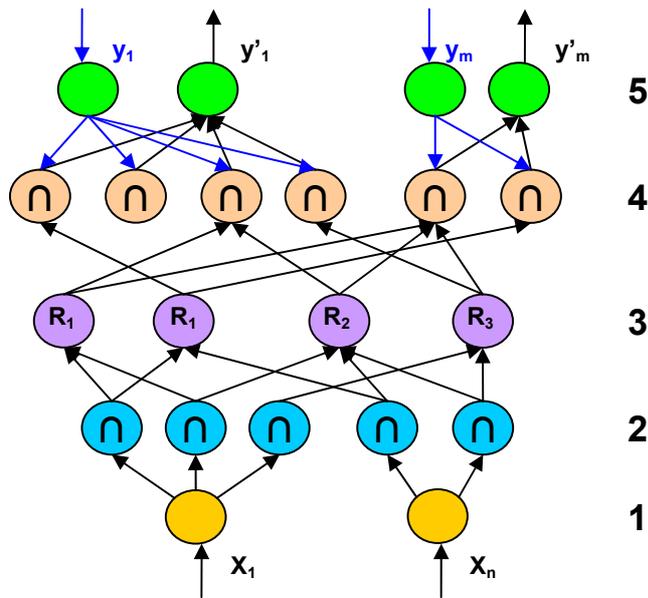


Figura 24. Arquitectura de un FALCON

El modelo FALCON usa dos tipos de algoritmos de aprendizaje: aprendizaje no supervisado para localizar las funciones de membresía o base de reglas iniciales, y aprendizaje supervisado (gradiente descendiente) para el ajuste óptimo de los parámetros de las funciones de membresía.

El algoritmo *backpropagation* (BP) está basado en la propagación hacia atrás del error E_k para cada neurona como

$$E_k = \frac{1}{2}(y_k - o_k)^2 \quad (19)$$

Donde y_k es la salida deseada para esa neurona y o_k es el valor actual de esa salida. A partir de esta medida de error, los parámetros del sistema neuro difuso son modificados de forma general como

$$p(t+1) = p(t) - \eta \frac{dE_k}{dp} \quad (20)$$

Donde $\eta > 0$ es la velocidad de aprendizaje del sistema y t representa el número de ajustes que se hacen al parámetro en cuestión.

FALCON puede presentar problemas al trabajar con datos insuficientes o ruidosos. En estos casos es útil utilizar funciones de membresía gaussianas de la forma

$$f(x, c, \sigma) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma} \right)^2} \quad (21)$$

Cuando el número de datos de entrenamiento no es suficiente para que el algoritmo de aprendizaje no supervisado defina una cantidad suficiente de funciones de membresía para todo el universo de la variable de entrada, se presentan indefiniciones que equivalen a errores infinitos en la salida. También el ruido en las entradas puede alterar sustancialmente el desempeño de este sistema, ocasionando errores grandes o incluso indefiniciones. En un sistema de control deben evitarse estas indefiniciones en la salida, y generalmente es preferible obtener valores con errores pequeños.

4.5.2. ANFIS

Este es uno de los sistemas neuro difusos mas antiguos y esta entre los mas usados. ANFIS es el acrónimo para *Adaptive Neuro Fuzzy Inference System* (Sistema de Inferencia Adaptativo Neuro Difuso), un modelo desarrollado originalmente por Roger Jang¹⁹. Su arquitectura implementa las 6 capas de un FIS tipo Takagi Sugeno Kang y se puede ilustrar en la figura

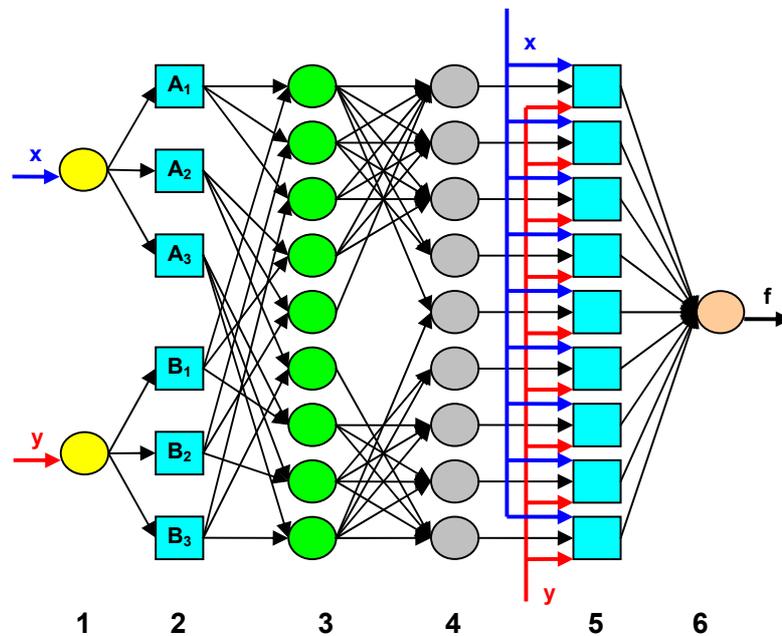


Figura 25. Arquitectura de un ANFIS

Existen dos conjuntos de parámetros: S1, que representa las funciones de membresía, y S2, que representan los coeficientes de las funciones de inferencia lineal.

¹⁹ JANG, Roger. *Neuro-Fuzzy Modeling: Architectures, Analyses and Applications*, PhD Thesis. University of California, Berkeley, July 1992.

$$S1 = \{\{a_{11}, b_{11}, c_{11}\}, \{a_{12}, b_{12}, c_{12}\}, \dots, \{a_{1p}, b_{1p}, c_{1p}\}, \dots, \{a_{np}, b_{np}, c_{np}\}\} \quad (22)$$

$$S2 = \{\{c_{10}, c_{11}, \dots, c_{1n}\}, \dots, \{c_{p^0}, c_{p^1}, \dots, c_{p^n}\}\} \quad (23)$$

El procedimiento de aprendizaje de un ANFIS consta de dos partes. En la primera parte se propagan hacia delante los patrones de entrada, se mantiene constante S1 y se estima S2 usando un algoritmo de mínimos cuadrados (aprendizaje off-line). En la segunda parte, los patrones son propagados de nuevo pero esta vez hacia atrás y se utiliza un algoritmo de gradiente descendiente (tipo *back-propagation*) para modificar S1, manteniendo constante S2.

4.5.3. GARIC

El modelo GARIC fue desarrollado por Bherenji y Khedkar²⁰, y su nombre es el acrónimo para *Generalized Approximated Reasoning based Intelligent Control* (Control Inteligente basado en Razonamiento Aproximado Generalizado). Este modelo no implementa ni un FIS de tipo Mamdani ni un FIS de tipo Takagi Sugeno.

A diferencia de los otros modelos, GARIC consta de dos módulos llamados Red de Selección de Acción (ASN, *Action Selection Network*) y Red de Evaluación de Acción (AEN, *Action Evaluation Network*). La AEN es una red que se encarga de

²⁰ BHERENJI, H R y KHEDKAR, P. *Learning and Tuning Fuzzy Logic Controllers through Reinforcements*. IEEE Transactions on Neural Networks, Vol (3), pp. 724-740, 1992.

evaluar críticamente las acciones de la ASN. La estructura de una ASN consta de 5 capas como se muestra en la figura

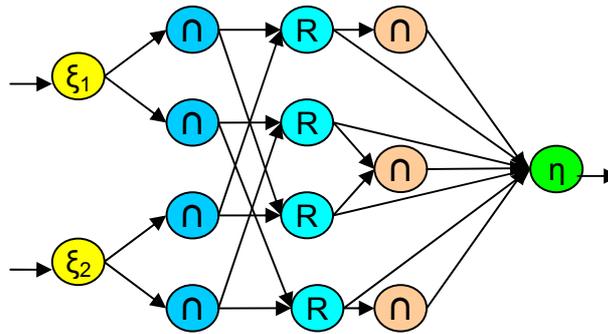


Figura 26. Arquitectura de una ASN

Aunque a primera vista se asimila al FIS de tipo Mamdani, es importante observar dos diferencias importantes con este modelo. En primer lugar, se puede observar que la última capa recibe las salidas tanto de la capa de consecuentes de reglas, como de la capa de antecedentes de reglas, haciendo que la salida sea función de los pesos w_i de las reglas del sistema. Además de esto, GARIC hace uso de un método de media de máximos locales para calcular las salidas de la capa de consecuentes de reglas, por lo que es necesario que estas salidas sean defuzzificadas antes de que sean acumuladas en el valor final de salida del controlador.

Aunque su desempeño es bueno, GARIC no es fácilmente interpretable como los modelos tradicionales Mamdani y Takagi Sugeno. Además, tampoco es fácil realizar la inicialización de la base de reglas a partir del conocimiento previo,

aunque una versión mejorada del modelo original puede manejar este usando un algoritmo refinado de aprendizaje por refuerzo.

4.5.4. NEFCON

NEFCON es el acrónimo para *Neuro Fuzzy Controller* (Controlador Neuro Difuso), un modelo que fue desarrollado inicialmente por Detlef Nauck y Rudolf Kruse²¹. Aunque implementa un FIS tipo Mamdani, su estructura esta simplificada a solo tres capas, las cuales realizan todas las tareas necesarias para el FIS Mamdani:

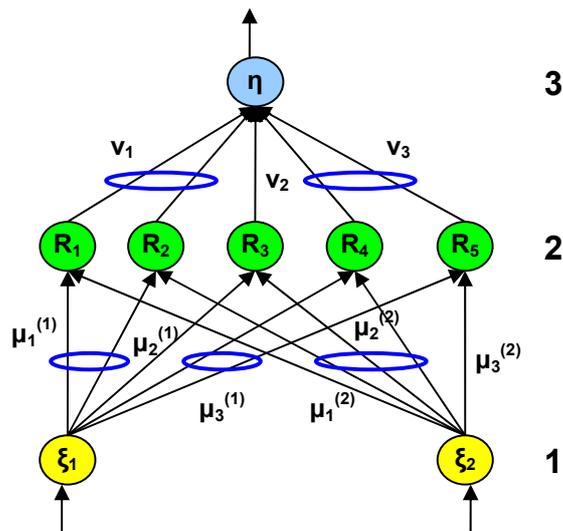


Figura 27. Arquitectura de un NEFCON

- **Capa de entrada:** se encarga de recibir los valores de entrada, actuar como interfaz de fuzzificación y distribuir estas entradas a la capa de reglas.

²¹ NAUCK, Detlef y KRUSE, Rudolf. *Neuro-Fuzzy Systems for Function Approximation*, 4th International Workshop Fuzzy-Neuro Systems, 1997.

- **Capa de reglas:** su labor es la de evaluar las reglas difusas a partir de los conjuntos difusos definidos en la capa anterior.
- **Capa de salida:** esta capa se encarga de realizar la inferencia difusa y defuzzificar la salida.

Por ejemplo, para el sistema de la figura podemos expresar el nodo R_1 con la relación

$$R_1: \text{SI } (\xi_1 \text{ es } A_1^{(1)}) \text{ Y } (\xi_2 \text{ es } A_1^{(2)}) \text{ ENTONCES } (\eta \text{ es } B_1) \quad (24)$$

Las conexiones en el NEFCON corresponden a pesos de naturaleza difusa en lugar de pesos reales. Para las conexiones entre la primera y la segunda capa (capa de reglas) es muy común definir **pesos compartidos** para diferentes conexiones. En la figura estos se ilustran con elipses alrededor de dichas conexiones y se caracterizan porque, durante el aprendizaje, el algoritmo realiza las mismas modificaciones a estos pesos compartidos para asegurar la integridad de la base de reglas.

Las conexiones entre la segunda y tercera capa (capa de salida) son calculadas con operaciones difusas como el máximo y el mínimo, de manera que el conjunto difuso de salida tiene la forma

$$v(y) = \max_{R_j} \left\{ \min \left\{ \mu_{j_1}^1(x_1), \dots, \mu_{j_n}^n(x_n), v_j(y) \right\} \right\} \quad (25)$$

El proceso de aprendizaje del NEFCON puede ser dividido en dos fases: la primera fase esta diseñada para aprender una base de reglas inicial si no se dispone de conocimiento previo, o para completar una base de reglas definida

manualmente. La segunda fase consiste en optimizar esta base de reglas intercambiando o modificando los conjuntos difusos de las reglas. Ambas fases utilizan un **error difuso**, que describe la calidad del sistema actual, para aprender u optimizar la base de reglas. Este error difuso considera también el signo de un valor óptimo de salida η_{opt} que debe ser conocido previamente. Para un conjunto de valores de x se puede expresar como

$$E(x_1, \dots, x_n) = \text{sgn}(\eta_{opt})e(x_1, \dots, x_n) \quad (26)$$

Existe una tercera etapa en la que se modifican los conjuntos difusos de acuerdo a un algoritmo de *back-propagation* y el error difuso.

Aunque NEFCON fue diseñado para trabajar como un controlador inteligente, los algoritmos fueron modificados posteriormente para realizar funciones de aproximación (NEFPROX) o de clasificación de patrones (NEFCLASS).

4.5.5. SONFIN

SONFIN es el acrónimo para *Self Constructing Neural Fuzzy Inference Network* (Red de Inferencia Neuro Difusa Auto Constructiva). Este modelo desarrollado por Lin y Juang²², implementa un sistema modificado de inferencia difusa tipo Takagi-Sugeno-Kang como se presenta en la figura.

²² JUANG, Chia Feng y LIN, Chin Teng. *An Online Self Constructing Neural Fuzzy Inference Network and its Applications*, IEEE Transactions on Fuzzy Systems, Vol 6, No.1, pp.12-32, 1998.

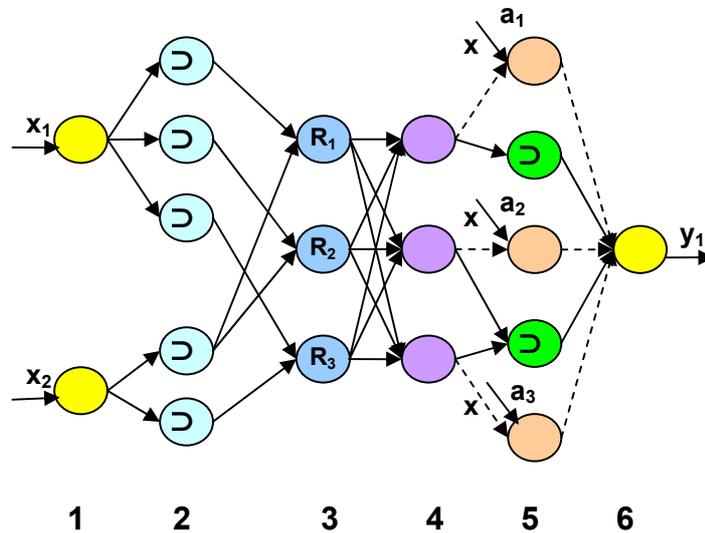


Figura 28. Arquitectura de un SONFIN

En el proceso de identificación de la estructura de la capa de antecedentes de reglas, se divide el espacio de entradas de manera flexible de acuerdo con un algoritmo de categorización. De la misma manera, a cada regla se le asigna solo un valor *singleton*. Después de esto se agregan algunos términos que representen las variables de entradas a las neuronas en la parte de consecuentes de reglas, a medida que transcurre el aprendizaje. Los parámetros de esta capa son ajustados mediante algoritmos de mínimos cuadrados medios o de mínimos cuadrados recursivos.

Para aumentar la capacidad de representación del conocimiento del SONFIN se puede incorporar transformaciones lineales para cada variable de entrada, hasta que sean necesarias solo unas pocas reglas o se alcance una mayor precisión.

Las transformaciones lineales apropiadas también son aprendidas dinámicamente en la fase de identificación de parámetros del SONFIN.

Un SONFIN puede ser usado para operación normal en cualquier instante durante el proceso de aprendizaje sin entrenamiento repetido en el patrón cuando se requiera una operación *on-line*. SONFIN es quizás el modelo neuro difuso computacionalmente mas costoso. Además, el modelo es adaptable a las especificaciones del usuario sobre precisión requerida.

4.5.6. FUN

FUN es el acrónimo para *Fuzzy Net* (Red Difusa) un modelo neuro difuso desarrollado por Sulzberger *et al*²³, y cuya estructura se ilustra en la figura.

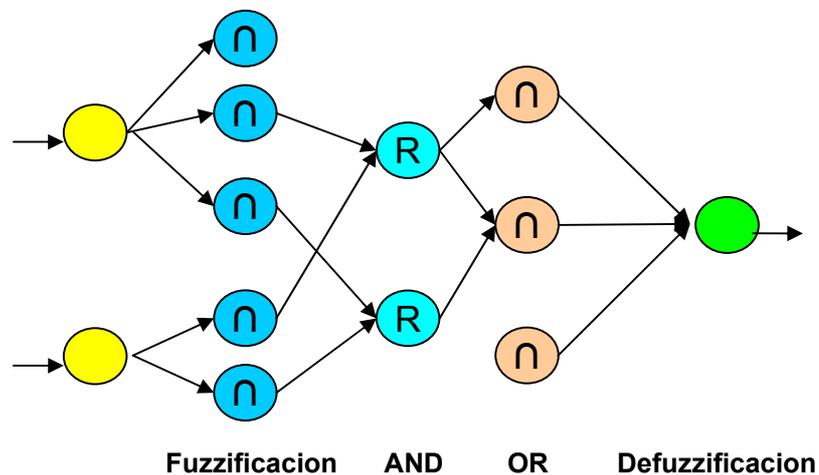


Figura 29. Arquitectura de un FUN

²³ Sulzberger SM *et al*. *FUN: Optimization of Fuzzy Rule Based Systems Using Neural Networks*, In Proceedings of IEEE Conference on Neural Networks, San Francisco, pp 312-316, March 1993.

En FUN se han definido unas neuronas especiales que pueden evaluar funciones lógicas para garantizar una traducción inequívoca de las reglas difusas y las funciones de membresía a la red neuronal. Las variables de entradas son almacenadas en las neuronas de entrada. Las neuronas en la primera capa oculta contienen las funciones de membresía y realizan la fuzzificación de los valores de entrada. En la segunda capa oculta se calculan las conjunciones (normas T). Las funciones de membresía de las variables de salida se almacenan en la tercera capa de salida. Su función de activación es una OR (conormas T) difusa finalmente las neuronas de salida contienen las variables de salida y tienen una función de activación de tipo defuzzificación.

La red es inicializada con un número fijo de reglas difusas y las funciones de membresía correspondientes, y después de eso usa un procedimiento estocástico que cambia aleatoriamente los parámetros de las funciones de membresía y las conexiones dentro de la estructura de red. El proceso de aprendizaje es conducido por una función de costo, que se evalúa después de la modificación aleatoria. Si la modificación produce un desempeño mejorado se mantiene, de lo contrario se deshace. FUN puede ser entrenado con las estrategias estándar de entrenamiento de redes neuronales, tales como aprendizaje supervisado o con refuerzo.

4.5.7. EfuNN

EfuNN es un acrónimo para *Evolving Fuzzy Neural Network* (Red Neuro Difusa

Evolutiva). Este modelo, que fuera desarrollado por Kasabov y Qun²⁴, implementa un sistema de inferencia de tipo Mamdani con una estructura dinámica basada en la regla “El ganador toma todo” (aprendizaje competitivo) para la activación de los nodos de la capa de reglas.

En EfuNN todos los nodos son creados durante el proceso de aprendizaje. Los nodos que representan las funciones de membresía (segunda capa) pueden ser modificados durante el aprendizaje. En esta capa pueden evolucionar nuevas neuronas si, para un determinado vector de entrada, el grado de pertenencia a los conjuntos difusos existentes es menor que un cierto umbral. Cada nodo de la capa de reglas (tercera capa) es definido por dos vectores de pesos de conexión que son ajustados a través de una técnica híbrida de aprendizaje.

En la figura 30 podemos ver un esquema general de este modelo donde se aprecia la variabilidad de la cantidad de nodos por capa.

Una versión modificada de este modelo es el dmEfuNN (*Dynamic Evolving Fuzzy Neural Network*) que se basa en la selección dinámica de diferentes nodos de reglas (no solo los nodos de las reglas “ganadoras”) para cada nuevo vector de entradas, cuyos valores de activación se usan para calcular los parámetros dinámicos de la función de salida. Otra diferencia es que mientras EfuNN implementa sistemas de inferencia de tipo Mamdani, dmEfuNN estima la regla Takagi-Sugeno-Kang basado en un algoritmo de mínimos cuadrados.

²⁴ KASABOV, N y QUN, Song. *Dynamic Evolving Fuzzy Neural Networks with 'm-out-of-n' Activation Nodes for On-line Adaptive Systems*, Technical Report TR99/04, Department of information science, University of Otago, 1999.

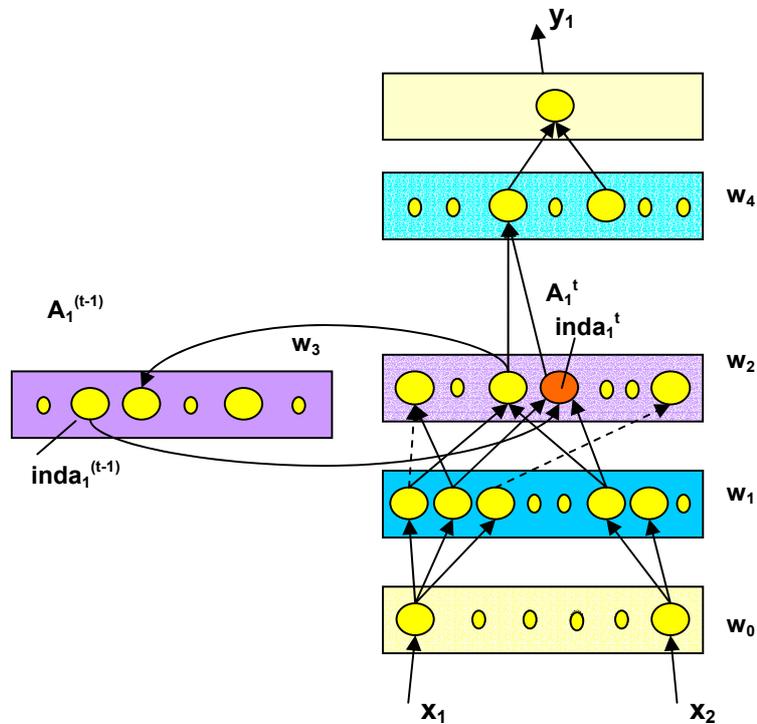


Figura 30. Arquitectura de un EfuNN

4.5.8. Diseño evolutivo de sistemas neuro difusos

Desarrollado por Ajito Abraham y Baikunth Nath²⁵. Este modelo se puede considerar como una estructura evolutiva que puede asumir diferentes modelos de inferencia (Mamdani, Takagi-Sugeno-Kang), diferentes parámetros de aprendizaje, diferentes arquitecturas o diferentes parámetros de las funciones de pertenencia.

Para lograr la capacidad de evolución este modelo se apoya en algoritmos

²⁵ ABRAHAM, Ajith y NATH, Baikunth. *Evolutionary Design of Neuro-Fuzzy Systems – A Generic Framework*, In Proceedings of the 4th Japan – Australia joint Workshop on Intelligent and Evolutionary Systems, Japan, November 2000.

genéticos. En este modelo solamente se definen las capas y no el número de neuronas, tal y como se muestra en la figura 31.

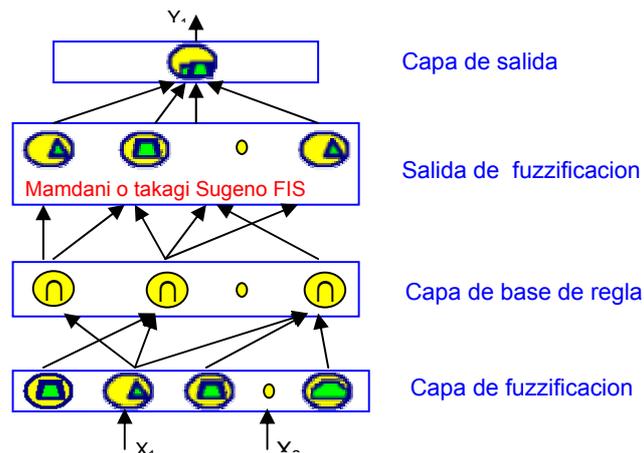


Figura 31. Arquitectura de un diseño evolutivo de sistemas neuro-difusos

El proceso de la búsqueda evolutiva se encarga de decidir el tipo y cantidad óptima de nodos y conexiones entre capas. La capa de *fuzzificación* y la capa de antecedentes de reglas funcionan como en los otros modelos neuro difusos. La parte del consecuente de la regla se determina de acuerdo al sistema de inferencia dependiendo al tipo de problema, el cual a su vez es adaptado de acuerdo con el mecanismo de búsqueda evolutivo. Los operadores de *defuzzificación* y agregación también serán adaptados de acuerdo al FIS escogido por el algoritmo evolutivo.

La figura 32 ilustra la estructura de cálculo y la interacción de varios procedimientos de búsqueda. Para cada parámetro de aprendizaje hay una

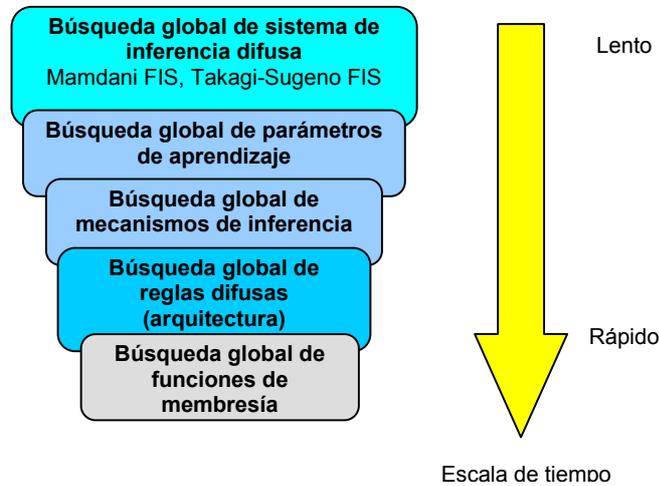


Figura 32. Estructura computacional de un diseño evolutivo de sistemas neuro-difusos

búsqueda global de mecanismo de inferencia que procede en una escala más rápida de tiempo en un ambiente decidido por los parámetros de aprendizaje, sistema de inferencia y el problema. Similarmente para cada arquitectura, la evolución de los parámetros de las funciones de membresía procede a una escala mas rápida de tiempo decidido por la arquitectura mecanismo de inferencia, regla de aprendizaje, tipo de sistema de inferencia y el problema. La jerarquía de los diferentes procesos de adaptación confía en el conocimiento previo, por ejemplo, si hay mas conocimiento previo acerca de la arquitectura que del mecanismo de inferencia entonces es mejor implementar la arquitectura en un nivel superior de la estructura de cálculo.

5. APLICACIONES ACTUALES Y PERSPECTIVAS DE LOS SISTEMAS NEURO DIFUSOS EN EL CONTROL DE PROCESOS

5.1. APLICACIONES ACTUALES DE LOS SISTEMAS NEURO DIFUSOS

5.1.1. Sistemas de control supervisorio jerárquico basados en sistemas neuro difusos

El objetivo de estos sistemas es trabajar con procesos complejos con muchas variables de entrada y de salida. Una solución actual para esto son los controladores convencionales MIMO, pero estos presentan problemas con el comportamiento extremadamente no lineal y las metas de control “conflictivas” del proceso. La opción que se presenta es un sistema supervisorio que divide la tarea global de control en niveles, donde un nivel superior que evalúa el estado del sistema a partir de una clasificación con base en las señales de sensores y luego activa los módulos de control de menor jerarquía.

La tarea del sistema neuro difuso dentro de este esquema es la evaluación del estado del proceso a partir de las señales de los sensores, labor que puede ser realizada con modelo como el NEFCLASS (variante del NEFCON para labores de clasificación). Se han desarrollado aplicaciones de este sistema supervisorio para

ambientes industriales (planta de manufactura de tubos de vidrio²⁶) y de robótica (control de robots con múltiples sensores²⁷).

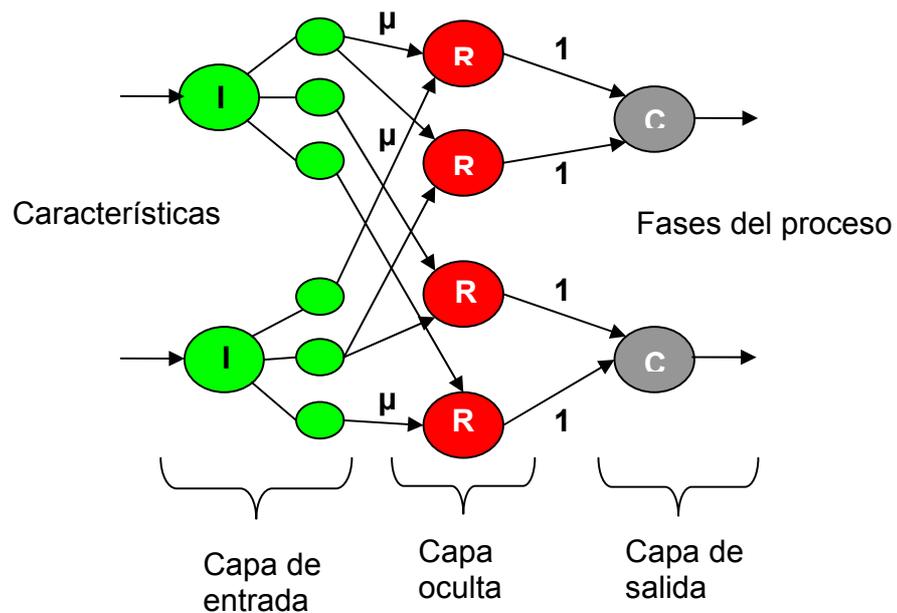


Figura 30. Sistema neuro difuso para evaluación de características

5.1.2. Control del torque en motores eléctricos

Los motores de reluctancia conmutados (SMR, Switched Reluctance Motor) son un tipo de motores que se caracterizan por la simplicidad en su circuito de manejo y una cierta tolerancia a fallas en su conversor, sin embargo, su aplicación está

²⁶ FREY, C H y KUNTZE, B. *A Neuro Fuzzy Supervisory Control System for Industrial Batch Processes*. Fraunhofer Institute for Information and Data Processing. Disponible en la Web: http://www.sfb588.uni-karlsruhe.de/publikationen/2000_17.pdf

²⁷ GIESEN, K. *Structure Variable Multi-Sensoric Supervisory Control of Humna Interactive Robots* (2004). Fraunhofer Institute for Information and Data Processing. Disponible en la Web: http://www.iitb.fraunhofer.de/servlet/is/8293/CCA2004_Taiwan.pdf

limitada por la presencia de rizado en su forma de onda de torque. El estudio de estos motores ha demostrado que este inconveniente se puede minimizar controlando adecuadamente la forma de onda de corriente. Para esto se ha propuesto utilizar un compensador neuro difuso²⁸ en serie con un controlador PI convencional, dentro de un lazo de control de velocidad. Los resultados obtenidos mostraron las potencialidades de la incorporación de esta señal de compensación, y el efecto de la variación en la forma de las funciones de membresía para la reducción de rizado y armónicos.

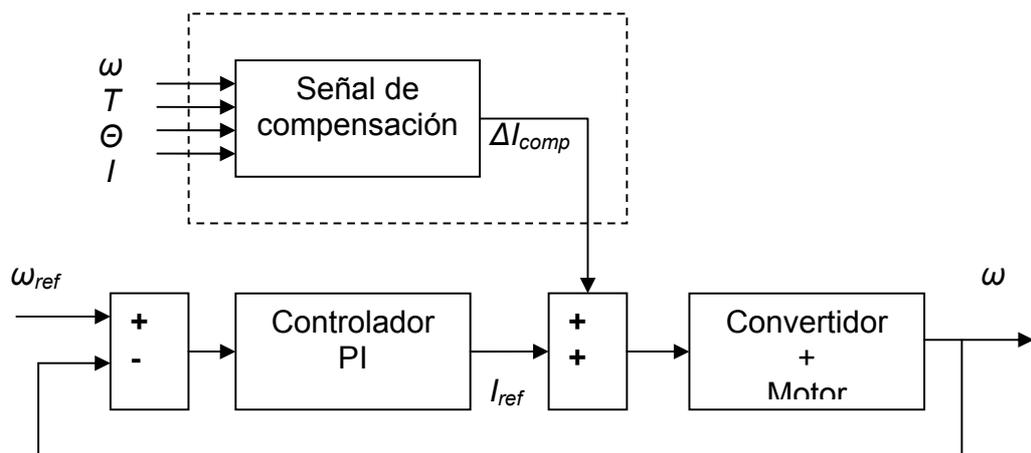


Figura 31. Esquema de compensación para el control de rizado en el torque

Otro tipo de motor con al que se han aplicado los sistemas neuro difusos son los motores de inducción controlados por PWM. El esquema desarrollado se

²⁸ HENRIQUES, Luís *et al.* *Torque Ripple Minimization in a Switched Reluctance Drive by Neuro-fuzzy Compensation.* (1999) Universidade Federal do Rio de Janeiro. Disponible en la Web: <http://arxiv.org/pdf/cs.RO/0010003>

denomina Control Neuro Difuso de Torque Directo²⁹ (DTNFC, *Direct Torque Neuro Fuzzy Control*) y utiliza un ANFIS para conseguir un control de alto desempeño de flujo y torque.

5.1.3. Sistemas inteligentes para el control de la anestesia en los quirófanos de un hospital

Se ha desarrollado un sistema inteligente para la administración de anestesia³⁰ que utiliza lecturas electro-encefalográficas, medidas de los latidos del corazón y de la presión sanguínea.



Figura 32. Control de anestesia en un quirófano

²⁹ GRABOWSKI, Pawel *et al.* A Simple Direct-Torque Neuro Fuzzy Control of PWM-Inverter-Fed Induction Motor Drive. (2000). Institute of Control and Industrial Electronics, Warsaw University of Technology. Disponible en la Web: http://powerelec.ece.utk.edu/pubs/bose_trans_ie_aug_2000.pdf

³⁰ LINKENS, D A. *Intelligent computer-based systems for anaesthesia control in hospital operating rooms.* Department of Automatic Control and Systems Engineering University of Sheffield. Disponible en la Web: <http://cyber.felk.cvut.cz/EUNITE03-BIO/pdf/Linkens.pdf>

Después de un pre-procesamiento, estas señales son llevadas a un sistema neuro difuso que a su vez clasifica la profundidad de la anestesia. Además, las señales de los latidos del corazón y la presión sanguínea son utilizadas como una segunda medida basada en un sistema de Lógica Difusa. Las dos medidas son combinadas después para dar una indicación final de la profundidad de la anestesia. Esta indicación es realimentada hacia un sistema de control que regula la infusión de la droga, mantener así el estado anestésico. La arquitectura de este sistema ha sido validada a través de la simulación extensiva y la realización de pruebas clínicas.

5.1.4. Control neuro difuso de un sistema Diesel de viento basado en biomasa

Esta aplicación consiste en un controlador neuro difuso para un motor Diesel de viento compuesto por una turbina de viento con un generador de inducción conectado a un barraje AC en paralelo con un generador Diesel que tiene un generador síncrono³¹.

³¹ JURADO, Francisco y SAENZ, José R. *Neuro-Fuzzy Control in Biomass-Based-Wind-Diesel Systems (2002)*. Disponible en la Web: <http://www.psc02.org/papers/s31p01.pdf>

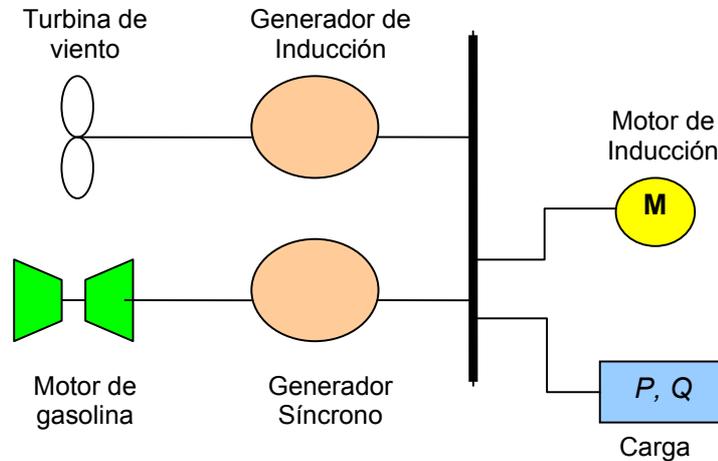


Figura 33. Sistema de potencia Diesel de viento

El controlador consta de dos partes: un controlador principal que controla la velocidad del motor, y un regulador automático de voltaje. En ambos casos, las entradas son el error en la variable de proceso y su derivada. En esta aplicación se muestra que a través de la sintonización de los controladores de Lógica Difusa, se puede alcanzar un desempeño temporal óptimo del sistema dentro de un amplio margen de condiciones de operación, en comparación con un controlador de Lógica Difusa con parámetros fijos y un controlador PID.

5.1.5. Control neuro difuso de estructuras utilizando realimentación de aceleración

Este esquema de control describe una nueva aproximación para la reducción de

las vibraciones inducidas por el ambiente en edificios³². En esta aproximación, la energía de cada edificio se disipa a través de unos amortiguadores *magnetoreológicos* (MR), cuyas propiedades de amortiguamiento son modificadas continuamente a través de un controlador neuro difuso. Este esquema de control semi-activo se basa en el desarrollo de una correlación entre las aceleraciones del edificio (entrada al controlador) y el voltaje aplicado al amortiguador MR (salida del controlador), y a su vez, esta correlación forma la base para el desarrollo de una estrategia de control neuro difuso.

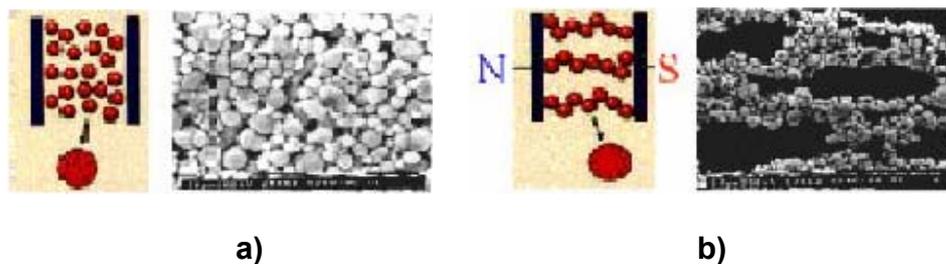


Figura 34. Fluidos magnetoreológicos. a) sin campo magnético; b) con campo magnético

Después de una simulación y comparación con otras técnicas de control, se encontró que los amortiguadores MR son mecanismos de control menos efectivos que los amortiguadores pasivos, para un modelo de edificios con un solo grado de libertad, mientras que son superiores cuando se usan con estructuras con múltiples grado de libertad para reducción de la aceleración lateral.

³² SCHURTER, Kyle y ROSCHKE, Paul. *Neuro-Fuzzy Control of Structures Using Acceleration Feedback*. (2001).Dept. of Civil Engineering, Texas A&M University. Disponible en la Web: <http://www.iop.org/EJ/article/0964-1726/10/4/322/sm1422.pdf>

5.1.6. Sistema de balanceo de columnas de fluido

Otro ejemplo de aplicación de estos sistemas es un controlador para un sistema de balanceo de columnas de fluido, que balancea una columna inestable contenida entre dos tanques, uno en cada extremo bombeando hacia atrás o hacia delante de los tanques. El problema fue formulado como un problema de múltiples entradas y una sola salida (MISO) y el control en tiempo real fue evaluado contra un controlador PID. Después de una corta simulación el algoritmo proporcionó resultados razonables comparados con el controlador PID y se sugirió mayor investigación sobre control en tiempo real (RTC).

5.2. PERSPECTIVAS FUTURAS DE LOS SISTEMAS NEURO DIFUSOS

Lo que hemos descrito hasta ahora es simplemente la “punta del *iceberg*” que representan los sistemas neuro difusos en las ciencias. Desde luego que no se pueden considerar como la solución última a los retos de la Ingeniería del Control y del Procesamiento de Datos, pero hay indicios entre los investigadores que hablan de las grandes posibilidades a futuro con estos sistemas.

Hablando del campo del control, estos sistemas deben ser refinados aún pero su capacidad de contingencia e imitación de las capacidades del operador humano, los colocan en un lugar importante dentro del espectro de técnicas de control

inteligente. Es importante señalar que un triunfo que han obtenido hasta ahora los sistemas neuro difusos es la interpretabilidad de la compleja tarea de control en términos del lenguaje natural. No obstante, el desarrollo ha mostrado que si se quieren obtener resultados mejores desde el punto de vista del control, se tiene que ir perdiendo inteligibilidad del sistema, algo que hace que estos se comporten mas como redes neuronales, es decir, como cajas negras de entradas y salidas.

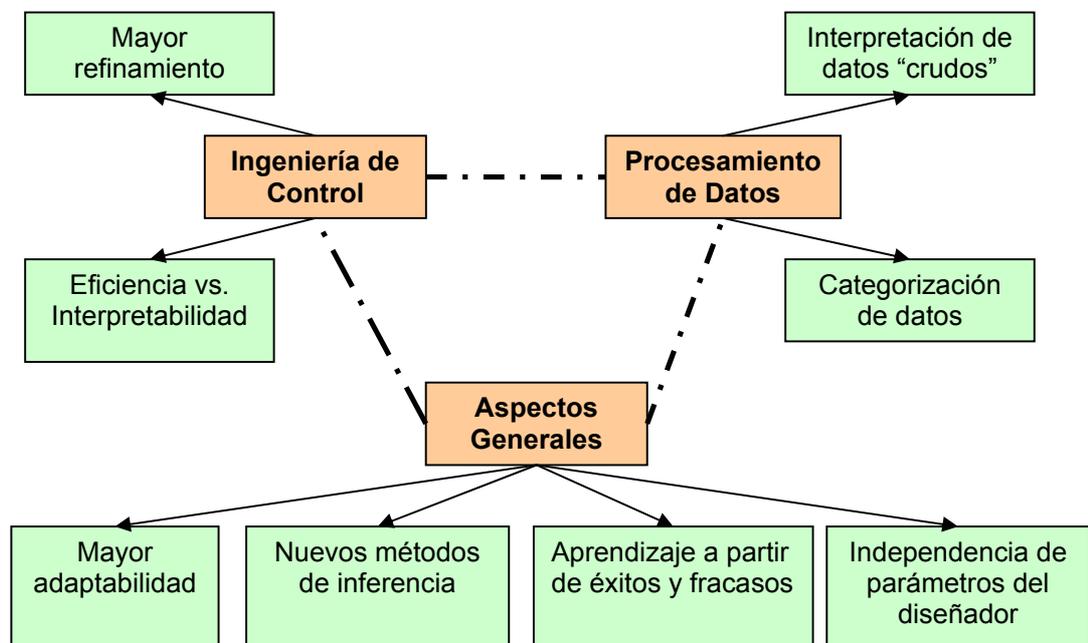


Figura 35. Perspectivas de los sistemas neuro difusos

Pero la inteligencia e inteligibilidad de los sistemas neuro difusos les ha guiado hacia labores que inicialmente no se les planteó, e incluso nosotros mismos hemos obviado en este documento: el procesamiento y análisis de datos. En un mundo donde cada vez hay mayor disponibilidad de datos "crudos", el análisis inteligente de estos para transformarlos en información útil, labor hasta ahora

exclusiva del intelecto humano, puede verse apoyada en gran medida por los sistemas neuro difusos que se aplican a categorización de datos. Basta una breve ojeada a los trabajos de investigación actuales sobre sistemas neuro difusos para constatar el creciente número de aplicaciones en el análisis e interpretación de datos.

Como la mayoría de los sistemas biológicos, que se pueden adaptar a cualquier ambiente, se necesitan sistemas neuro difusos totalmente adaptables para encarar los problemas del mundo real del futuro. La mayoría de los modelos neuro difusos existentes, se basan en los parámetros de red especificados por el usuario, pero para que el sistema sea completamente adaptable, su desempeño debe ser independiente de estos parámetros. La inteligencia de tales sistemas podría ser aumentada o mejorada mas adelante si el proceso de adaptación pudiera aprender de los éxitos y de los errores y aplicar ese conocimiento a problemas nuevos

Se debe enfocar más la investigación hacia el desarrollo de otros métodos de inferencia que sean computacionalmente económicos y eficientes. Los métodos de aprendizaje de gradiente descendiente son usados para la adaptación (aprendizaje de parámetros), en los sistemas difusos. Los algoritmos de optimización de segundo orden (gradiente conjugado, algoritmo Levenberg Marquardt, etc) han sido exitosos en aprendizaje más rápido en las redes neuronales y alcance de un desempeño mejor en general. Se podrían extender técnicas similares fácilmente a los sistemas neuro difusos.

Durante todo el tiempo estuvimos tratando de imitar el cerebro humano: su manera de razonar, de modelar la incertidumbre, el aprendizaje, etc. Es tiempo que empecemos a pensar en un sistema que trabaje mejor que los humanos ³³

³³ ABRAHAM, Ajith. *Beyond Integrated NeuroFuzzy Systems: Reviews, Prospects, Perspectives and Directions*. School of Computing and Information Technology. Disponible en la Web: <http://ajith.softcomputing.net>

CONCLUSIONES

- En el presente documento se ha podido constatar y exponer de manera didáctica como se ha recurrido una y otra vez a las técnicas inteligentes de procesamiento de la información ante las limitaciones que tienen los sistemas clásicos de control frente a las características del mundo real como son la no linealidad y la variancia en el tiempo. Este denominado Control Inteligente es un universo que aborda no solo tecnologías diferentes (Sistemas Expertos, Lógica Difusa, Redes Neuronales, Algoritmos Genéticos, entre otros) sino también estrategias diferentes al simple control realimentado.
- Para poder presentar los sistemas neuro difusos primero se realiza una descripción breve acerca de las técnicas independientes de Redes Neuronales y Lógica Difusa. Por fuerza de la extensión y la comprensibilidad de este documento no fue posible hacer un análisis mas profundo de ambos, pero en todo caso queda al lector el recurso de investigar las referencias que se han incluido para complementar estos temas. Y aunque perfectamente se pudieran señalar omisiones importantes, se considera que se ha incluido la información mas pertinente al tema central de esta monografía.
- Es importante anotar que se ha logrado clasificar los sistemas neuro difusos de una manera clara, que hace distinción entre los diferentes tipos de combinaciones entre Redes Neuronales y Lógica Difusa. En este aspecto se

debe subrayar el enfoque específico hacia los sistemas neuro difusos híbridos, ósea aquellos en los que resulta difícil distinguir la red neuronal del sistema difuso, por estar profundamente combinados. Además se expusieron ocho prototipos actuales y cinco casos de aplicación en diferentes áreas del control.

- Al realizar este trabajo se pudo comprender el lugar que ocupan estas técnicas de control que aún se rotulan como “no convencionales” en contraposición con las técnicas “clásicas” de variable compleja y análisis en el espacio de estados. Sin demeritar de ninguna manera lo que se ha alcanzado hasta el momento con estas técnicas, gracias a esta investigación se ha verificado el interés de los científicos hacia las técnicas neuro difusas. Si bien hoy en día el panorama revela un amplio dominio de las técnicas clásicas en la industria y un uso limitado de las técnicas emergentes (como complemento de las técnicas tradicionales), en el mediano y largo plazo estas llegarán para complementar capacidades extra que no se han tenido hasta el momento, e incluso reemplazarlos en algunas tareas. En este proceso, el desarrollo técnico ya ha comenzado hace tiempo pero mientras se decanta una técnica con la suficiente confiabilidad y aceptación cultural la investigación y el desarrollo desempeñan un papel importante.

RECOMENDACIONES

El presente documento no ha pretendido comprender totalmente un tema con extenso y complejo como éste, e incluso a medida que se avanza en el estudio del mismo, se pueden notar otras oportunidades de investigación y desarrollo fuera de los modestos alcances de esta monografía. Concientes de esta realidad, y preocupados por el avance científico de la Universidad Tecnológica de Bolívar los autores consideran oportuno realizar algunas recomendaciones para estudiantes y docentes.

- Desarrollar un estudio más profundo y concienzudo de los fundamentos matemáticos de las teorías sobre Redes Neuronales y Lógica Difusa, aprovechando el alto nivel de formación de la planta docente, en conjunto con estudiantes de últimos niveles interesados en el tema y con actitudes positivas hacia las matemáticas.
- Plantear una línea de investigación enfocada hacia los sistemas neuro difusos dentro del Grupo de Investigación en Automatización Industrial y Control Automático (GAICO) de la UTB.
- Utilizar los recursos informáticos que poseen los laboratorios de la UTB y la información disponible en la Web para examinar las herramientas de software disponibles³⁴ para simulación y diseño de ANFIS, NEFCON, y otros sistemas

³⁴ Se deben resaltar los toolboxes de MATLAB® y los software gratis de la Universidad de Magdeburgo (<http://fuzzy.cs.uni-magdeburg.de/>)

neuro difusos. Esto se puede orientar, por ejemplo, a la realización de una guía de laboratorios en Sistemas Neuro Difusos bajo MATLAB.

- Fomentar el interés entre los estudiantes de Ingenierías Eléctrica y Electrónica hacia los sistemas neuro difusos a través de actividades académicas (como exposiciones, trabajos breves de investigación, lectura de documentos científicos) y de divulgación (Feria de la Ciencia) a nivel universitario.
- Planear y ofrecer en el mediano plazo a los estudiantes de estas carreras un curso de Introducción al Control Inteligente dentro de las opciones de electivas complementarias.
- Crear otras fuentes de conocimiento estructurado (monografías, proyectos de investigación, trabajos de grado) acerca de las aplicaciones de los sistemas neuro difusos al procesamiento y análisis de datos. Podría investigarse sobre caracterización de estados de un sistema a partir de información de sensores, o clasificación automática de patrones en señales usando técnicas neuro difusas.
- Orientar los esfuerzos a largo plazo hacia la realización de prototipos que empleen los conceptos teóricos expuestos aquí, bien sea en alguno de los modelos aquí descritos (sistema de balanceo de columnas de fluido) o en otro desarrollado posteriormente (sistemas de péndulo invertido, regulación de temperatura en cámaras para neonatos).

BIBLIOGRAFÍA

- **Libros**

BROWN, Martin y HARRIS, Chris. *Neurofuzzy Adaptive Modelling and Control (Prentice-Hall International Series in Systems and Control Engineering)*. Prentice Hall, 1995.

FULLER, Robert. *Introduction to Neuro-Fuzzy Systems (Advances in Soft Computing)*. Physica-Verlag Heidelberg, 2000

HILERA, José Ramón y MARTÍNEZ, Víctor José. *Redes neuronales artificiales: fundamentos, modelos y aplicaciones*. México D. F : Alfaomega : Ra-ma, 2000

MARTÍN del Brío, Bonifacio y SANZ Molina, Alfredo. *Redes neuronales y sistemas difusos*. México D. F : Alfaomega, 2002

NGUYEN, Hung et al. *A First Course in Fuzzy and Neural Control*. Chapman & Hall/CRC, 2002

TSOUKALAS, Lefteri y UHRIG, Robert. *Fuzzy and Neural Approaches in Engineering (Adaptive and Learning Systems for Signal Processing, Communications and Control Series)*. Wiley-Interscience, 1997

- **Documentos científicos**

ABRAHAM, Ajith. *Beyond Integrated NeuroFuzzy Systems: Reviews, Prospects, Perspectives and Directions*. School of Computing and Information Technology.

Disponible en la Web: <http://ajith.softcomputing.net>

_____. *Neuro Fuzzy Systems: State-of-the-art Modeling Techniques*. School of Computing & Information Technology. 2001

Disponible en la Web en: <http://ajith.softcomputing.net>

BARAJAS, Leandro y BARÓN, María. *Sistema De Control Integrado Neurofuzzy*.

Trabajo de grado. Universidad Distrital Francisco José De Caldas. 1998

Disponible en la Web: <http://home.comcast.net/~lgbarajas/pdf/SCINEF-Full.pdf>

JANTZEN, Jan. *Neural and Neurofuzzy Control*. Technical University of Denmark.

2003. Disponible en la Web: <http://fuzzy.iau.dtu.dk/download/nfcon.pdf>

NAUCK, Detlef y KRUSE, Rudolf. *Choosing Appropriate NeuroFuzzy Models*.

University of Braunschweig. 1994

Disponible en la Web:

VILLASEÑOR LOZANO, Carlos. *Modelado difuso neuronal con algoritmo de aprendizaje estable*. Tesis de maestría Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional de México. 2003

Disponible en la Web:

<http://www.ctrl.cinvestav.mx/Profesores/WenYu/pdf/MaTesCA.pdf>

ANEXOS

ANEXO A.

CARACTERÍSTICAS PRINCIPALES DE LAS TÉCNICAS DE CONTROL

CARACTERÍSTICA	TÉCNICAS CLÁSICAS		TÉCNICAS INTELIGENTES		
	Análisis en frecuencia	Espacio de estados	Redes Neuronales	Sistemas de Lógica Difusa	Sistemas neuro difusos
Fundamento teórico matemático	Variable compleja	Álgebra lineal	Algoritmos no lineales de adaptación	Conjuntos difusos	Algoritmos no lineales de adaptación
Almacenamiento de la información	Polinomios en variable compleja	Matrices y derivadas	Pesos de conexión	Conocimiento experto	Enfoque variable
Modelamiento entrada salida	Función de transferencia	Matriz de transición de estados	Parejas de patrones	Reglas lingüísticas	Enfoque variable
Interpretabilidad de las leyes de control	Expresiones matemáticas	Expresiones matemáticas	No es posible	Lenguaje natural	Variable
Comportamiento de la respuesta	Determinístico	Determinístico	No determinístico	No determinístico	No determinístico
Precisión	Excelente	Excelente	Buena	Buena	Buena
Adaptabilidad	Ninguna	Ninguna	Aprendizaje y sintonización	Sintonización opcional	Aprendizaje y sintonización
Manejo de múltiples entradas y salidas	Imposible	Sencillo	Sencillo	Sencillo	Sencillo

CARACTERÍSTICAS PRINCIPALES DE LAS TÉCNICAS DE CONTROL

(Continuación)

CARACTERÍSTICA	TÉCNICAS CLÁSICAS		TÉCNICAS INTELIGENTES		
	Análisis en frecuencia	Espacio de estados	Redes Neuronales	Sistemas de Lógica Difusa	Sistemas neuro difusos
Procesos con parámetros variables	No aplica	No aplica	Buen desempeño	Buen desempeño	Buen desempeño
Efectos de ruido	Sensible	Sensible	Robusto	Robusto	Robusto
Tolerancia a fallas	Ninguna	Ninguna	Muy buena	Buena	Muy buena
Procesamiento de la información	Secuencial	Secuencial	Paralelo	Depende de hardware	Paralelo
Complejidad algorítmica	Baja	Moderada	Moderada	Alta	Alta
Requerimiento de hardware	OpAmp's, Microcontroladores, PLC's, PC's	PLC's, PC's	Procesadores neuronales	Procesadores difusos PC's	PC's
Costos de implementación	PC's	Moderados	Moderados	Altos	Altos

ANEXO B.

GLOSARIO DE TÉRMINOS

Agente: componente de un sistema neuro difuso. Puede ser un sistema de inferencia difusa o un sistema experto con lógica difusa.

Algoritmos genéticos: modelos de adaptación guiado por un esquema evolutivo similar al de los seres vivos.

ANFIS: acrónimo para *Adaptive Neuro Fuzzy Inference System* (Sistema de Inferencia Adaptativo Neuro Difuso). Prototipo neuro difuso híbrido.

Backpropagation: algoritmo de aprendizaje para redes neuronales basado en la propagación hacia atrás del error.

Base de reglas: bloque de un sistema difuso encargado de almacenar el conocimiento en forma de una base de datos y una base de reglas.

Capas: conjunto de neuronas cuyas entradas provienen de la misma etapa de procesamiento de los datos y cuyas salidas se dirigen hacia otra etapa de procesamiento común.

Control inteligente: mecanismo capaz de razonar, comprender y aprender de la tarea de control que realiza, mejorando permanentemente su desempeño.

Defuzzificador: bloque de un sistema difuso encargado de convertir el conjunto de valores de salida expresados como valores difusos en un solo valor puntual.

Diligente: componente de un sistema neuro difuso. Contiene una red neuronal, que continuamente está observando las entradas y salidas del agente y en determinados casos aprende las situaciones nuevas.

DMEFUNN: acrónimo para *Dynamic Evolving Fuzzy Neural Network* (Red Neuro Difusa Evolutiva Dinámica). Prototipo neuro difuso híbrido.

EFUNN: acrónimo para *Evolving Fuzzy Neural Network* (Red Neuro Difusa Evolutiva). Prototipo neuro difuso híbrido.

FALCON: acrónimo para *Fuzzy Adaptive Learning Control Network* (Red Difusa Adaptativa de Control de Aprendizaje). Prototipo neuro difuso híbrido.

Fuzzificador: bloque de un sistema difuso encargado de la traducción de los valores físicos de las entradas en valores lingüísticos que formen parte del universo de discurso normalizado.

FUN: acrónimo para *Fuzzy Net* (Red Difusa). Prototipo neuro difuso híbrido.

Función de transferencia: función matemática que determina el valor que toma la salida de la neurona dado un valor en su estado de activación.

Función de membresía: función matemática que evalúa el grado de pertenencia de un valor a una categoría lingüística.

GARIC: acrónimo para *Generalized Approximated Reasoning based Intelligent Control* (Control Inteligente basado en Razonamiento Aproximado Generalizado). Prototipo neuro difuso híbrido.

Lógica difusa: planteamiento que permite evaluar cuantitativamente el valor de verdad/falsedad de una afirmación.

Modelos concurrentes: combinación de redes neuronales y sistemas de control difuso utilizados en conjunto pero sin modificarse entre sí.

NEFCON: acrónimo para *Neuro Fuzzy Controller* (Controlador Neuro Difuso). Prototipo neuro difuso híbrido.

Neurona artificial: elemento de procesamiento similar a una neurona biológica capaz de recibir los estímulos presentes en sus múltiples entradas, ponderarlos de acuerdo con su importancia relativa, y producir una respuesta.

Redes de función de base radial: redes neuronales con neuronas cuya función de activación es una campana gaussiana.

Red neuronal: estructura de procesamiento de datos basada en cálculos aritméticos simples ejecutados por elementos de procesamiento llamados neuronas, y conexiones entre estas representadas matemáticamente por factores de multiplicación llamados pesos.

Redes neuronales difusas: Esta combinación es por su naturaleza una red neuronal artificial, con el fin de mejorar su capacidad de aprendizaje o su desempeño.

Regla composicional de inferencia: regla que permite inferencia de las salidas de un sistema de control difuso dados unos ciertos valores de entradas y reglas de expertos.

Sistemas expertos: sistemas computacionales capaces de emular el comportamiento de los expertos humanos.

Sistemas neuro difusos cooperativos: combinación de redes neuronales y sistemas difusos en los que se utiliza la primera para realizar adaptaciones al segundo.

Sistemas neuro difusos híbridos: combinación de redes neuronales y sistemas difusos en los que se evalúa el proceso de inferencia difusa a través de las capas de una red neuronal especial.

SONFIN: acrónimo para *Self Constructing Neural Fuzzy Inference Network* (Red de Inferencia Neuro Difusa Auto Constructiva). Prototipo neuro difuso híbrido.

Traductor: componente de un sistema neuro difuso que extrae la información aprendida desde la arquitectura resultante de la red neuronal, y la expresar en reglas difusas de control.

Umbral: valor mínimo que debe superar la suma neta de entradas ponderadas a una neurona para que se active la salida.

ANEXO C.

ÍNDICE ALFABÉTICO DE MATERIAS

A

AEN 66

Algoritmos

de entrenamiento 7

Genéticos 5

Amortiguadores

magnetoreologicos 84

ANFIS 65

Antecedente de reglas 60

Aprendizaje

supervisado 22

no supervisado 23

on-line 24

offline 24

Aprendizaje neuro difuso 44

ASN 66

B

Backpropagation 63

Base de reglas difusas 35

C

Capas 21, 27

Centroide 36

Centro de gravedad 36

Centro de suma 37

Clusterizacion 46

Conormas T 51

Consecuente de reglas 60

Control

adaptativo lineal 15

con modelo interno 12

de anestesia 81

de estructuras 83

de torque 79

Experto 7

Inteligente 3

inverso directo 10

por aprendizaje con
refuerzo 14
predictivo 13

D

Defuzzificador 36, 60
DmEFUNN 74

E

EFUNN 73
Estado de activacion 19

F

FAM 45
FALCON 62
Fase de entrenamiento 44
FUN 72
Funcion de activacion 19
Función de membresía 32
 Gaussiana 64
Funcion de transferencia 19
Fuzzificador 34, 59

G

GARIC 66

M

Modelamiento en adelanto 7
Modelo inverso de la planta 10
Metodos de centroide 36
Modelos neuro difusos
concurrentes 54

N

NEFCON 68
 NEFPROX 70
 NEFCLASS 70
Neurona artificial 16
Normalizacion de fuerzas de las
reglas 61
Normas T 51

P

Perceptrón
 monocapa 25
 multicapa 26
Pesos sinápticos 19
Programación de ganancias 7

R

Redes de funciones de base radial 28

Regla de inferencia 36

Reglas de operador 7

S

Sistemas Expertos 5

Sistema de inferencia

 Mamdani 38, 58

 Takagi Sugeno Kang 38, 60

Sistemas difusos de control 31

Sistemas neuro difusos

 cooperativos 56

 híbridos 58

SMR (Swicthed Reluctante Motor) 79

SONFIN 70

V

Variables de estado 8

Variables programadoras 8

Redes neuronales artificiales 16

Redes neuronales difusas 54