

UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE INGENIERÍA

*“Control Neurodifuso de Humedad
para un Fitotrón”*

TESIS

Que como parte de los requisitos para obtener el
grado de Maestro en Ciencias

PRESENTA:

Ing. Carlos Alfredo Razo Montes

ASESOR:

Dr. Luis Morales Velázquez

CO-ASESOR:

Dr. René de Jesús Romero Troncoso

Santiago de Querétaro, Septiembre de 2011



Universidad Autónoma de Querétaro

Facultad de Ingeniería

Maestría en Instrumentación y
Control Automático

**CONTROL NEURODIFUSO DE HUMEDAD PARA UN FITOTRÓN
TESIS**

Que como parte de los requisitos para obtener el grado de
Maestro en Ciencias en Instrumentación y Control Automático

Presenta:

Ing. Carlos Alfredo Razo Montes

Dirigido por:

Asesor: Dr. Luis Morales Velázquez

Co-asesor: Dr. René de Jesús Romero Troncoso

SINODALES

Dr. Luis Morales Velázquez

Presidente

Firma

Dr. René de Jesús Romero Troncoso

Secretario

Firma

Dr. Roque Alfredo Osornio Ríos

Vocal

Firma

Dr. Irineo Torres Pacheco

Suplente

Firma

Dr. Ramón Guevara González

Suplente

Firma

Dr. Gilberto Herrera Ruiz

Director de la Facultad

Dr. Luis Gerardo Hernández Sandoval

Director de Investigación y Posgrado

Centro Universitario
Querétaro, Qro.
Septiembre de 2011
México

Resumen

Desde que se intentó dar respuesta a cuestiones sobre el origen y la evolución de las especies en la tierra con experimentos confiables, las condiciones adversas de la naturaleza fueron el principal reto. Hoy día, un espacio donde se tenga la capacidad de manipular las condiciones ambientales, ejerciendo un control de éstas, ha sido la necesidad primordial de los diversos campos de estudio de la biotecnología. En los últimos años se han desarrollado dispositivos que son utilizados en laboratorios y centros de investigación con el fin de simular distintas posibilidades climáticas; sin embargo, el costo de éstos es demasiado elevado además que los gastos de instalación, capacitación y mantenimiento es exagerado por parte de los proveedores, generando una dependencia tecnológica en cuanto al software de control y en muchos casos el sistema de control se limita a un simple controlador ON/OFF, incumpliendo las demandas de precisión, estabilidad y tiempo de respuesta que el sector necesita. Con base en lo anterior, en este trabajo se propone un sistema de control en tiempo real basado en tecnología FPGA para el control de humedad en un fitotrón mediante un controlador neurodifuso. El sistema fue descrito en VHDL y fue implementado en un sistema embebido diseñado por el grupo HSPdigital basado en un dispositivo de alto desempeño FPGA, el cual implica flexibilidad en el sistema de control, sin restricciones de compatibilidad o dependencia de alguna tecnología en particular y que permite la integración de distintos módulos de control para dar una solución tipo SoC al control de variables ambientales; haciendo, por lo tanto, un sistema de control más eficiente, completo y genérico. Además, el controlador desarrollado está basado en técnicas de control sumamente sofisticadas como lo son la lógica difusa y las redes neuronales artificiales el cual ofrece resultados eficientes y precisos. Los resultados de las pruebas demuestran un mejor desempeño de las técnicas de inteligencia artificial en comparativa con los métodos clásicos en el control de humedad del fitotrón.

(**Palabras clave:** Control de humedad, Controlador neurodifuso, Sistema embebido, VHDL, FPGA)

Abstract

Since attempts to answer questions about the origin and evolution of species on earth with reliable experiments, the adverse conditions of nature were the main challenge. Nowadays, a place with the capacity of manipulate the environmental conditions, exercising control over these, has been the primary need of the many fields of study in biotechnology. In recent years, have been developed devices that are used in laboratories and research centers in order to simulate different weather possibilities, but their cost is too high in addition to the cost of installation, training and maintenance is exaggerated by providers, creating a technological dependence on the software control and in many cases the control system is limited to a simple ON / OFF control, in violation of the demands for precision, stability and response time that the sector needs. Based on the above, this paper proposes a control system in real time based on FPGA technology to control moisture in a phytotron through a neurofuzzy controller. The system was described in VHDL and was implemented in an embedded system FPGA-based high-performance device, which involves flexibility in the control system, unrestricted compatibility or reliance in any particular technology and allows the integration of different control modules giving a SoC solution to control environmental variables, therefore, making an efficient control system, full and generic. In addition, the developed controller is based on highly sophisticated control techniques such as fuzzy logic and artificial neural networks, which offers efficient and accurate results. The results of the tests show a better performance of artificial intelligence techniques in comparison with conventional methods to control moisture in a phytotron.

(Keywords: Humidity control, Neurofuzzy-controller, Embedded system, VHDL, FPGA)

Dedicatorias

*Este trabajo está dedicado a todas aquellas personas que
contribuyeron para que este proyecto sea una realidad.
Principalmente, con todo mi amor y cariño,
a la familia **Razo Alba**.*

Agradecimientos

En primera, doy gracias a **Dios** por haberme permitido llegar, una vez más, a esta etapa en la historia de mi vida profesional; y a mi esposa **Synthia Susana** e hijos **Héctor Jesrael** y la pequeña **Adaline Susana**, por su eterna conflagración por hacerme una mejor persona. Son ellos quienes me han demostrado un amor con tal devoción que en ocasiones no parece humano. Absolutamente, ¡gracias por todo!..

Sabiendo de ante mano que jamás existirá una forma de agradecer en esta vida su trabajo y esfuerzo para que mis anhelos y sacrificios no sean en vano, deseo expresarles que mis ideales y logros son también suyos y constituye el legado más grande que pudiera recibir. Gracias por dejarme experimentar el tipo de amor por el que la gente puede morir. Una vez más, más que nunca: *¡Los amo tanto!*..

A a mis padres, **Arturo** y **Graciela**, pues nadie ha recibido más afecto ni un apoyo tan incondicional como el que ellos me han dado; y de igual forma a mis hermanos, **Daniel Arturo** y **Ana María**, por todo aquello que nos ha tocado vivir en los buenos y malos momentos. . .

Gracias también a todos mis **amigos y compañeros de la división de investigación y posgrado**, que si por las prisas no pongo sus nombres, ellos sabrán a quiénes me refiero, sobre todo aquellos con los que inicié a lado suyo esta gran aventura.

Por su fe, su tolerancia y su inestimable ayuda, a **mis profesores** de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro, ¡gracias a todos ellos!.

A mi asesor, el **Dr. Luis Morales Velázquez**, que sin sus críticas y sugerencias este trabajo aún seguiría inconcluso. Reservo un reconocimiento especial a mi co-asesor, el **Dr. René de J. Romero Troncoso** quien aportó su tiempo en forma generosa, a pesar de sus muchos compromisos, para guiarme en el desarrollo de este proyecto; del mismo modo al **Dr. Roque A. Osornio Ríos**. . . las palabras no son suficientes para expresar la gratitud que siento por dejarme aprender de ellos lo que en las aulas no se enseña, gracias por tan buenos consejos pero sobre todo gracias por la experiencia compartida y las aventuras vividas con los amigos de HSPdigital.

A mis sinodales, el **Dr. Irineo Torres Pacheco** y el **Dr. Ramón Guevara González**, por el apoyo proporcionado para llevar a cabo este proyecto.

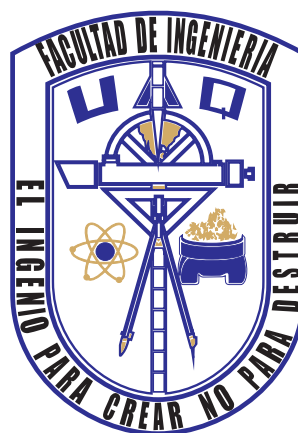
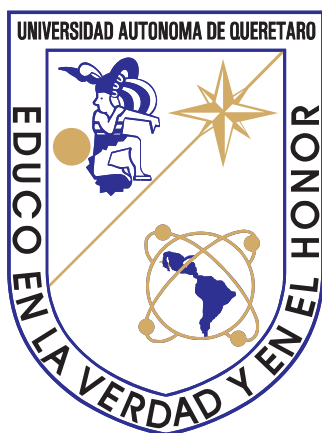
A todos ellos, por lo que fue y por lo que aún falta. . .

Gracias!!!

Carlos Alfredo Razo Montes

Agradecimientos Institucionales

A la **Universidad Autónoma de Querétaro** por permitirme el realizar mis estudios de maestría en la **Facultad de Ingeniería, campus San Juan del Río**. Le agradezco el haberme proporcionado el equipo y las instalaciones necesarias para llevar a buen termino mis estudios, y por hacerme sentir parte de esta gran institución como lo es la Universidad de Autónoma de Querétaro.



Al grupo de investigación **HSPdigital**, por haberme permitido crecer profesionalmente en mis estudios de licenciatura y maestría; y en mayor medida, por haber conocido directamente a las personas con las que siempre colaboré, que más que compañeros, puedo llamarlos amigos.



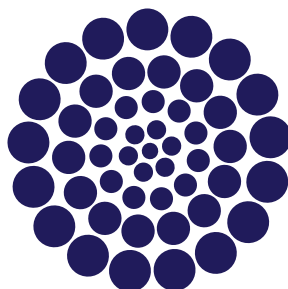
Al Consejo Nacional de Ciencia y Tecnología (CONACYT), por la beca otorgada durante el período de **julio 01, 2009** a **junio 30, 2011** con número de becario: **230464**.

Título del trabajo: Control Neurodifuso de Humedad en un Fitotrón

Autor: Ing. Carlos Alfredo Razo Montes

Asesor: Dr. Luis Morales Velázquez

Co-asesor: Dr. René de J. Romero Troncoso



CONACYT

Consejo Nacional de Ciencia y Tecnología

Índice general

Resumen	I
Abstract	II
Dedicatorias	III
Agradecimientos	IV
Agradecimientos Institucionales	V
Índice de figuras	XI
Índice de tablas	XIII
I. Introducción	1
1.1. Antecedentes	1
1.2. Objetivos	3
1.2.1. Objetivo general	3
1.2.2. Objetivos particulares	3
1.3. Justificación	4
1.4. Descripción general del proyecto	7
II. Revisión de literatura	8
2.1. Estado del arte	8
2.2. El fitotrón	10

2.2.1.	Descripción	11
2.2.2.	Aplicaciones	12
2.3.	Cómputo reconfigurable	13
2.4.	Lenguajes descriptivos de hardware	14
2.5.	Control inteligente	16
2.5.1.	Preámbulo	17
2.5.2.	Técnicas	18
2.6.	Redes neuronales artificiales	18
2.6.1.	La neurona artificial: inspiración biológica	19
2.6.2.	El perceptrón simple	20
2.6.3.	Arquitectura de red	26
2.6.4.	Características principales	30
2.6.5.	Algoritmos de aprendizaje	30
2.6.6.	Control neuronal	37
2.7.	Lógica difusa	38
2.7.1.	Comienzos	38
2.7.2.	Conjuntos difusos	39
2.7.3.	Estructura de un controlador difuso	43
III	Metodología	51
3.1.	Dinámica del sistema	54
3.1.1.	Arquitectura de la red neuronal para la identificación del sistema	55
3.2.	Esquemas de control	56
3.2.1.	Control clásico	56
3.2.2.	Control inteligente	59
3.3.	Puesta del experimento	62
3.3.1.	Instrumentación del fitotrón	65

IV.Resultados	66
4.1. Identificación de la dinámica sistema	66
4.2. Validación de los controladores	69
4.2.1. Controlador PID	69
4.2.2. Controlador difuso	70
4.3. Validación del sistema de control de humedad en el fitotrón	71
V. Conclusiones	72
Bibliografía	74
A. Códigos Fuentes	79
B. Productos Obtenidos	80

Índice de figuras

1.1. Cámara de ambiente controlado (Fuente: http://www.labotec.co.za) . . .	5
1.2. Diagrama general del proyecto	7
2.1. Diagrama de una neurona (Copyright ©2001 The McGraw-Hill Companies. All rights reserved.)	20
2.2. Arquitectura del perceptrón simple	21
2.3. Funciones de activación	23
2.4. Semiespacios que define el perceptrón	25
2.5. Funciones lógicas separables linealmente	25
2.6. Funciones lógicas no separables linealmente	26
2.7. Arquitectura de múltiples capas.	27
2.8. Red neuronal de propagación hacia adelante (<i>feedforward</i>)	28
2.9. Red neuronal de propagación hacia atrás (<i>feedback</i>)	29
2.10. Tipos de conexiones de las redes neuronales	29
2.11. Arquitectura de una red neuronal de tres capas	32
2.12. Definición de la variable lingüística <i>Temperatura</i>	40
2.13. Función de pertenencia de un conjunto nítido y difuso.	41
2.14. Funciones de pertenencia más comunes.	43
2.15. Arquitectura del controlador difuso	44
2.16. Funciones de pertenencia triangular	47
2.17. Conjuntos difusos implicados	49

3.1.	Diagrama general a bloques	53
3.2.	Sistema de control en lazo cerrado	53
3.3.	Respuesta del sistema	54
3.4.	Arquitectura de la red neuronal para la identificación del sistema	56
3.5.	Modelo hardware del sistema de control clásico	57
3.6.	Arquitectura hardware del controlador PID	59
3.7.	Modelo hardware del sistema de control inteligente	59
3.8.	Funciones de pertenencia del controlador difuso	60
3.9.	Arquitectura hardware del controlador difuso: fusificación	61
3.10.	Arquitectura hardware del controlador difuso: defusificador, base de reglas y máquina de inferencias	61
3.11.	Fitotrón sobre el cual se realizaron las pruebas funcionales	63
3.12.	Elementos del sistema de control de humedad	64
3.13.	Instrumentación del fitotrón	65
4.1.	Resultado de la identificación con la red neuronal	67
4.2.	Resultado de la identificación con la herramienta <code>ident</code> de Matlab®	67
4.3.	Validación del sistema de control clásico	69
4.4.	Validación del sistema de control difuso	70
4.5.	Validación del sistema de control de humedad del fitotrón	71

Índice de tablas

2.1. Ventajas y desventajas de las plataformas tecnológicas para el procesamiento de señales	15
2.2. Funciones de activación comúnmente usadas	22
2.3. Características de una red neuronal	30
2.4. Descripción de las variables lingüísticas	45
2.5. Valores de las variables lingüísticas	45
2.6. Base de reglas	46
3.1. Tabla de equivalencias de controladores	58
4.1. Valores de los pesos sinápticos obtenidos durante el entrenamiento de la red neuronal	68

Capítulo I

Introducción

1.1. Antecedentes

Es un hecho que la vicisitud climática en la que se ha visto inmersa la agricultura a nivel mundial, durante las últimas décadas, ha cambiado drásticamente sus prácticas y métodos. Esta perturbación de la naturaleza ha despertado la preocupación de muchas naciones, las cuales tienen entre sus prioridades más urgentes garantizar la sustentabilidad alimenticia de su país.

De este modo, la tendencia en la producción de cultivos revela cada vez más el uso inminente de tecnologías capaces de aumentar la productividad para cubrir las demandas que la sociedad impone.

La tecnología aplicada en este rubro, que mejores resultados ha ofrecido, es sin duda, el uso de controladores para el control de variables ambientales. Con éstos, se pueden manipular las condiciones ambientales y se fortalece la competencia y el estándar de calidad en los productos.

A la fecha, la tecnología de control abarca un amplio rango que va desde el control clásico, basado en modelos matemáticos, hasta el control inteligente, que se basa en la capacidad de simular las cualidades cognoscitivas del comportamiento humano como el aprendizaje, planificación y adaptación al entorno, entre otras.

Desde hace varios años, en la Universidad Autónoma de Querétaro existen diversos trabajos realizados acerca del desarrollo de controladores, tales como controladores convencionales de tres tiempos y sus derivados (P «Proporcional», I «Integral», PI «Proporcional–Integral», PD «Proporcional–Derivativo», PID «Proporcional–Integral–Derivativo»), controladores difusos y el empleo de tecnología FPGA (del inglés *Field Programmable Gate Array*, Arreglo de Compuertas Programables en Campo) utilizando lenguaje descriptivo de hardware.

Ejemplo de esto, se puede ver en el desarrollo basado en software de un sistema de control PID de temperatura aplicado en una máquina de inyección de plástico (Enríquez, 2006); en el desarrollo de un control difuso implementado en MatlabTM que posteriormente se utiliza en el paquete de LabVIEWTM para su aplicación en el sistema de riego en invernaderos y optimizar el empleo del agua (Rojas, 2003); en el desarrollo de un sistema de control *fuzzy*–PID que se encarga de la inyección de fertilizantes en un sistema de riego, el cual es implementado en un CPLD (del acrónimo en inglés *Complex Programmable Logic Device*, Dispositivo Lógico Programable en Campo) (Gómez, 2005); o bien, en el diseño en hardware de un sistema de control climático inteligente para invernaderos, donde se hace uso de las técnicas de control difuso, el cual es implementado en un FPGA (Castañeda *et al.*, 2006).

Respecto al uso de la tecnología FPGA, en los sistemas de control, es necesario resaltar trabajos donde se han integrado diversos módulos de control dentro de un sólo chip, como es el caso del desarrollo de un sistema de detección de ruptura de herramientas en los sistemas de manufactura por control numérico computarizado (CNC), donde se utilizan técnicas de procesamiento digital de señales que dan una solución del tipo SoC (del inglés *System on a Chip*, Sistema en un sólo Encapsulado) para el monitoreo en línea de la máquina (Romero, 2004);

la implementación de un sistema autocontenido, modular y flexible en un sólo circuito integrado, que hace uso del procesamiento de señales en hardware y las más recientes interfaces de comunicación, para el control de posición y generación de perfiles de un intercambiador automático de herramientas aplicación en máquinas CNC, con desempeño similar al de las tarjetas de control de posición comerciales, pero de muy bajo costo (Morales, 2007); y el diseño de un sistema de control para servomotores en CNC de alta velocidad, donde se implementaron algoritmos de control tradicionales aplicables en maquinados de alta velocidad de un alto grado de complejidad, que cumplió con los requerimientos de muestreo que exigen tales controladores (Osornio *et al.*, 2008).

1.2. Objetivos

1.2.1. Objetivo general

Generar un controlador de humedad aplicado a un fitotrón basado en las metodologías de control neuronal y difuso para la optimización de dicho parámetro en comparativa con los métodos clásicos.

1.2.2. Objetivos particulares

- i*) Desarrollar la instrumentación necesaria para el control de variables continuas de un fitotrón.
- ii*) Determinar el número de sensores, y su ubicación dentro de la cámara de crecimiento, para un óptimo monitoreo de la humedad de manera uniforme.
- iii*) Aplicar las leyes de control clásico de tres tiempos en el control de la humedad del fitotrón para determinar la dinámica de la cámara.

- iv*) Desarrollar un controlador *neurodifuso* para el control de variables continuas.
- v*) Integrar los módulos de las técnicas de control neuronal y difuso en tecnología FPGA para realizar pruebas funcionales en el control de humedad del fitotrón.
- vi*) Hacer una comparativa entre los resultados obtenidos mediante las técnicas de control clásico y los arrojados por el controlador *neurodifuso* desarrollado.

1.3. Justificación

Desde que se intentó dar respuesta a cuestiones sobre el origen y evolución de las especies en la tierra con experimentos confiables, en trabajos que se remontan a la época de Darwin¹, las condiciones adversas de la naturaleza fueron el principal reto.

Hoy día, un espacio donde se tenga la capacidad de manipular las condiciones ambientales, ejerciendo un control de éstas, ha sido la necesidad primordial de los diversos campos de estudio de la biotecnología.

Es fundamental, en el estudio detallado que intenta obtener modelos de crecimiento o la patología de los seres vivos, disponer del ambiente apropiado que se requiere para caracterizar el desarrollo de las especies, ya que éstas dependen en su totalidad de la biosfera donde se hallan inmersas.

Actualmente, el avance tecnológico ha permitido construir y acondicionar de manera artificial lugares para el cultivo de plantas o cría de algunos animales

¹Charles Robert Darwin (1809–1882), naturalista inglés.

fuera de su ambiente y clima habitual, como es el caso de los invernaderos o los estanques artificiales donde es posible manipular variables como la temperatura, la humedad o iluminación, entre otras. Y pese a que en ellos se obtienen mejores resultados que en la forma tradicional, la imprecisión debida a sus grandes dimensiones hace imposible realizar en ellos un estudio más sofisticado.

En los últimos años se han desarrollado dispositivos que son utilizados en laboratorios y centros de investigación con el fin de simular distintas posibilidades climáticas y poder evaluar el comportamiento del objeto a ensayar en dichas condiciones ambientales. Lo que se ha denominado cámara de ambiente controlado², como la ilustrada en la figura 1.1, permite controlar parámetros como temperatura, humedad e intensidad luminosa, pudiéndose también manejar otras variables como presión, nivel de CO₂, etc., según las especificaciones del ensayo de una manera mucho más precisa.



Figura 1.1: Cámara de ambiente controlado (Fuente: <http://www.labotec.co.za>)

²También conocidas como fitotrones (*phytotrons*).

Sin embargo, estas últimas presentan una serie de inconvenientes como los que se listan a continuación:

- ✘ Su costo es muy elevado.
- ✘ Los gastos de instalación, capacitación y mantenimiento son exagerados por parte de los proveedores.
- ✘ Se genera una dependencia tecnológica con el proveedor en cuanto al software de control.
- ✘ En muchos casos, el sistema de control se limita a un simple controlador ON/OFF.
- ✘ No se cumple aún con las demandas de precisión, estabilidad o tiempo de respuesta que el sector necesita.

La Universidad Autónoma de Querétaro, caracterizada por ser una institución comprometida con el desarrollo tecnológico del país, ha puesto gran interés en el tema y actualmente cuenta con el prototipo de un fitotrón que se espera cumpla con las exigencias que el área de la biotecnología demanda.

Con este proyecto, se desarrolló tecnología en el área de controladores implicando flexibilidad en los sistemas de control, sin restricciones de compatibilidad con otros sistemas y sin depender de una tecnología en particular haciendo, por lo tanto, un sistema de control más eficiente, completo y genérico.

Además, el controlador desarrollado está basado en técnicas de control sumamente sofisticadas, como lo son la lógica difusa (o *fuzzy logic*) y las redes neuronales artificiales (o *artificial neural networks*), que integran lo que se conoce hoy día como inteligencia artificial, la cual ofrece resultados eficientes y precisos.

Y con esto, sumado al afán de haber usado tecnología de punta como lo son los FPGAs y realizando todo el sistema de control a nivel hardware, se desarrolla un controlador, aplicado en el control de la humedad en un fitotrón, que cumple con las exigencias mencionadas.

Por otro lado, el trabajo plantea una comparación entre los métodos de control clásico *versus* los métodos de control moderno, y de este modo se ostenta, de manera generar, cuál de éstos desempeña un mejor cometido al ser aplicados en el control de humedad de un fitotrón.

1.4. Descripción general del proyecto

En términos generales, la figura 1.2 ilustra la descripción general del proyecto donde se puede observar un sistema de control en lazo cerrado. Es decir, puesta una referencia, ésta se compara con la medición dentro del fitotrón y la diferencia entre ambas (error) entra a un regulador, el cual determina la acción de control para alcanzar dicha referencia. En el propósito de este trabajo, el regulador es en primera instancia un controlador PID para luego ser sustituido por un controlador difuso y así obtener una comparativa entre ambas técnicas.

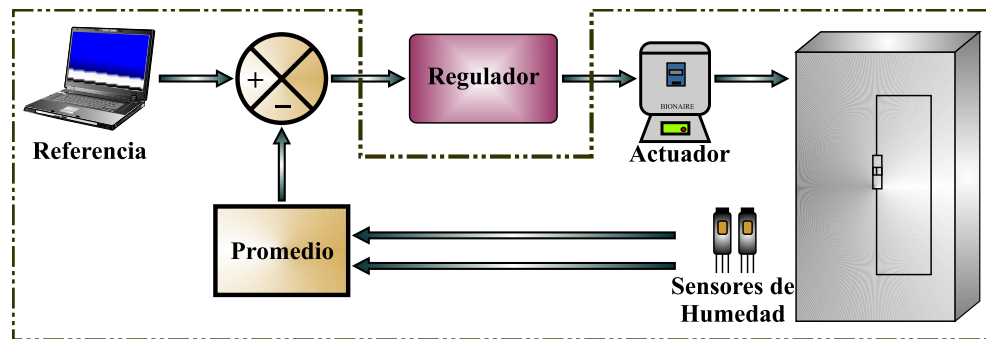


Figura 1.2: Diagrama general del proyecto

Capítulo II

Revisión de literatura

2.1. Estado del arte

Los sistemas neuro–difusos son el resultado de la simbiosis de dos tecnologías: las redes neuronales artificiales y los sistemas difusos, las cuales compiten en el campo del diseño de sistemas inteligentes con las técnicas de la inteligencia artificial y la teoría de control convencional. Estos sistemas tratan de explorar la complementariedad existente entre ambas tecnologías, aprovechando los méritos, y eludiendo los inconvenientes de cada una de ellas por separado (Navas, 2000).

En un contexto general, la integración de los sistemas difusos y las redes neuronales sigue tres líneas diferentes de trabajo que da lugar a tres tipos de sistemas (Lin y Lee, 1996), (Nie y Linkens, 1995):

- ◇ **Sistemas neuro difusos:** Usan las redes neuronales como herramienta para el diseño de sistemas difusos.
- ◇ **Redes neuronales difusas:** Traducen a términos de los sistemas difusos las redes neuronales tradicionales.
- ◇ **Sistemas híbridos *neurodifusos*:** Incorporan ambas tecnologías de forma independiente y complementaria en un nuevo sistema híbrido.

Respecto al campo de los sistemas *neurodifusos*, se han extendido múltiples enfoques que van desde la interpretación de un sistema difuso como un sistema neuronal al caso contrario, es decir, presentan una red neuronal como un sistema difuso. Incluso se han hecho estudios sobre la equivalencia entre ambas técnicas, llegando a la conclusión de que se puede aproximar el comportamiento de una red neuronal mediante un sistema difuso, y viceversa (Buckley *et al.* 1992).

Una de las principales ventajas de interpretar un sistema lógico difuso como una red neuronal es la posibilidad de utilizar los algoritmos de aprendizaje que se han venido desarrollando para estas últimas. Wang y Mendel (1992) proponen una interpretación de un sistema lógico difuso como una red neuronal de tipo radial.

Por su parte, Jang (1992) estudia la equivalencia funcional entre un controlador difuso y una red neuronal adaptiva denominada ANFIS (*Adaptive–Network–based Fuzzy Interference System*), en ésta la utilización de propagación dinámica hacia atrás del error proporciona al sistema la posibilidad de adaptar sus parámetros para minimizar el error de control.

Las posibilidades de ANFIS para el modelo de sistemas de no–lineales e incluso series caóticas se exploran en trabajos posteriores, donde incluso se formaliza la utilización de ANFIS como un sistema neuro–difuso para identificación y control (Jang y Sun, 1995).

Por otro lado, se han presentado diversos trabajos de versiones difusas de sistemas neuronales. Pal y Mitra (1992) proponen una versión difusa del algoritmo de propagación hacia atrás del error (*backpropagation*), y se proponen aplicaciones en el área de clasificación.

De manera general, la integración de ambas técnicas en los sistemas *neurodifusos* se ha venido utilizando en diferentes problemas como el tratamiento de imágenes por satélite, la predicción de la duración de las hospitalizaciones, la interpretación de señales de radar, el reconocimiento de caracteres manuscritos, el reconocimiento de voz, y la posibilidad que incorpora aprendizaje por refuerzo aplicable a problemas de control, entre muchas otras aplicaciones.

En conclusión, hay un gran interés en el área de los sistemas *neurodifusos*. Este interés nace de la facilidad para interpretar sistemas difusos como redes neuronales, y viceversa.

2.2. El fitotrón

El término se refiere, en botánica, a la herramienta de investigación usada para un completo estudio de las plantas; el cual contiene un gran número de ambientes controlados de forma individual que proporcionan los medios necesarios para estudiar el efecto de cada factor ambiental, tal como la temperatura o la luz, en un amplia gama de niveles simultáneamente (Parker, 2002).

Un fitotrón es, por tanto, la forma más compleja de instalación disponible con “*ambiente controlado*”, que se distingue de una simple instalación de cámaras para el cultivo de plantas por la condición de que es operado de tal manera que permite que un gran rango de factores ambientales puedan ser estudiados simultáneamente (Downs, 1980).

Es decir, son cuartos y gabinetes de ambiente controlado, iluminados natural y artificialmente que se usan, en conjunto, con incubadoras, cámaras de germinación de semillas, cuartos para el estudio de la duración del fotoperiodo e

invernaderos refrigerados mecánicamente para proporcionar una amplia variedad de condiciones ambientales. Los fitotrones hacen uso continuo y eficiente del espacio de ambiente controlado al no tener periodos de inactividad debido al mal funcionamiento, que periódicamente presentan las cámaras de crecimiento de plantas.

2.2.1. Descripción

Sólo a la instalación que es capaz de crear un amplia gama de climas se le debe llamar fitotrón. Sin embargo, es imposible tener un fitotrón en el que todos los factores ambientales están representados en sus gamas completas y con graduaciones adecuada (Rook, 1968). Al respecto, Lang (1962) expone que en la planificación de dichas instalaciones, debe quedar claro en qué rangos, y por cuánto, deben ser variados estos factores, debiendo permanecer constantes. Estos rangos dependerán, a su vez, de la investigación prevista, el control de la ingeniería, entre otros.

Dado que las temperaturas superiores a 50 °C son letales para las plantas y su crecimiento está limitado en temperaturas por debajo del punto congelación del agua, los fitotrones operan a temperaturas entre 0 y 50 °C y, en general, por razones prácticas de 10 a 40 °C.

Las plantas responden sólo a la luz en el espectro comprendido entre frecuencias cercanas al ultravioleta y cercanas al infrarrojo, ya que el material vegetal es dañado por la luz ultravioleta intensa, mientras que las frecuencias cercanas a los rayos infrarrojos no proporcionan suficiente energía para las reacciones de fotosíntesis y, por lo tanto, son biológicamente inactivas (Went, 1962). La luz solar natural o alguna forma de iluminación artificial pueden ser utilizadas como fuente de luz; pero, por lo general, se utilizan los dos tipos.

El control de humedad en los fitotrones

A pesar de que un control de humedad completo no está disponible en todos los fitotrones, un control de humedad preciso sería muy ventajoso en un gran número de problemas botánicos, ya que el porcentaje de humedad relativa repercute de manera directa en la función fisiológica más importante de las plantas: la transpiración (Millán *et al.*, 2010). Por ejemplo, en el estudio de las condiciones que favorecen el ataque del *Dothistroma pini* en pinos, determinando la resistencia a la sequía de los diferentes tipos de plantas, o en la investigación del efecto de la humedad en el desarrollo del xilema en los árboles. Sinnott (1960) toma nota de varias referencias en las plantas herbáceas donde la humedad relativa influyó en el desarrollo del xilema.

2.2.2. Aplicaciones

Los objetivos principales de cualquier equipo de ambiente controlado, en esencia, son:

- a) Proporcionar un ambiente reproducible en cualquier instante de tiempo para comprobar el efecto de algún tratamiento, operación, o tal vez un agente patógeno, que permita que el experimento se repita en sucesión rápida bajo el mismo entorno una y otra vez, independientemente de la temporada del año o la hora del día.
- b) O bien, para crear un amplia gama de climas que permitan la integración, o la diferenciación de distintos factores climáticos que afectan el crecimiento de las plantas y su desarrollo.

Un fitotrón, por tanto, es una herramienta de investigación en la que las plantas pueden ser cultivadas en ambientes controlados y reproducibles. En términos generales, las aplicaciones que se le han dado pueden resumirse como sigue:

- ◇ Principalmente, para el estudio de cómo un ambiente controlado modifica o cambia el desarrollo y crecimiento de la planta.
- ◇ Para complementar de manera más precisa la experimentación en invernaderos o campos de cultivo.
- ◇ En estudios sobre el cultivo e introducción de nuevas variedades y especies de plantas.

Los fitotrones, en consecuencia, desempeñan un papel importante en muchas fases de la investigación ecológica ya que permiten un estudio detallado de los sistemas bioquímicos y fisiológicos afectados por las condiciones ambientales.

2.3. Cómputo reconfigurable

Los dispositivos de cómputo reconfigurable han surgido como resultado de la innovación del procesamiento digital de señales, el cual exige no sólo la búsqueda de algoritmos cada vez más eficientes sino también el desarrollo de herramientas a fin de proveer soluciones que cumpla con los requerimientos exigidos por el mercado.

Con el objeto de optimizar aún más los recursos de los bloques digitales disponibles en los dispositivos programables, se buscó una alternativa que permitiera el uso de mayor cantidad de componentes y una eficiencia superior, el

resultado fue una nueva familia de arreglos lógicos conocidos por el acrónimo de FPGA (*Field Programmable Gate Array*, arreglo de compuertas programables en campo) los cuales se basan en una cantidad muy grande celdas lógicas, muy elementales, y una interconectividad mucho mayor que sus contrapartes (Romero, 2008).

La tabla 2.1 muestra una comparativa entre las diferentes plataformas tecnológicas para el procesamiento digital de señales, con sus principales ventajas y desventajas (Contreras, 2008).

En muchos de los casos, las aplicaciones que se resuelven con técnicas de cómputo reconfigurable reemplazan el equivalente en decenas o centenas de microprocesadores y DSPs (*Digital Signal Processors*) actuales. Este alto desempeño se alcanza construyendo operadores computacionales personalizados, rutas de datos, y líneas de producción apropiadas a la tarea a resolver.

Con este alcance, las características de una aplicación en particular, tales como paralelismo, localidad, y resolución de los datos se pueden explotar enormemente. Por ello, los dispositivos de cómputo reconfigurable proporcionan los beneficios de rendimiento computacional de los ASICs (*Application Specific Integrated Circuits*), pero conservando la flexibilidad y rápida reconfigurabilidad de un microprocesador de propósito general.

2.4. Lenguajes descriptivos de hardware

Existen diversos lenguajes descriptivos que han sido desarrollados en los últimos años, pero indudablemente el lenguaje que más difusión ha tenido y que se utiliza mayormente es el VHDL (Romero, 2008).

Tabla 2.1: Ventajas y desventajas de las plataformas tecnológicas para el procesamiento de señales

Plataforma	Ventajas	Desventajas
Software	<ul style="list-style-type: none"> · Diseño en alto nivel · Interfaces estándares · Usa computadoras de propósito general 	<ul style="list-style-type: none"> · Velocidad limitada · Cómputo especulativo
DSP	<ul style="list-style-type: none"> · Circuito probado funcionalmente · Plataforma de diseño de alto nivel · Puente con C, C++ y/o Matlab · Fácil conexión con memoria · Memoria interna · Puertos de comunicación · Costo moderado · Buena velocidad 	<ul style="list-style-type: none"> · Requiere circuito de soporte periférico · Obsolescencia general · Arquitectura predefinida · No permite la optimización de ciertos algoritmos
ASIC	<ul style="list-style-type: none"> · SoC (<i>System On a Chip</i>) · Control de las variables de diseño · Optimización tecnológica · Paralelismo · Integración de módulos diérsos: RAM, ROM, etc · Procesamiento en línea 	<ul style="list-style-type: none"> · Tiempos largos de desarrollo · Costo justificable en producción masiva · Dificultad de diseño · No son reprogramables
FPGA	<ul style="list-style-type: none"> · Posibilidad SoC · Diseño del sistema por el usuario final · Libertad de arquitectura · Diseño de mediano nivel · Portabilidad · Reconfigurabilidad · Procesamiento en línea 	<ul style="list-style-type: none"> · Desperdicio inherente de recursos · Complejidad de desarrollo, crea dependencia a <i>IP-cores</i> · Desarrolla dependencia tecnológica con los fabricantes

El lenguaje descriptivo VHDL es el acrónimo que representa la combinación de VHSIC y HDL, donde VHSIC es el acrónimo de *Very High Speed Integrated Circuit* y HDL es, a su vez, el acrónimo de *Hardware Description Language* (Pardo y Boluda, 2003).

Las características que han hecho del VHDL en lenguaje universal de descripción de circuitos son:

- ◆ El ser un lenguaje estándar definido como tal por el IEEE (*Institute of Electrical and Electronics Engineers*) (ANSI/IEEE 1076-1993).
- ◆ Que los proveedores del paquete tienen que seguir el estándar, haciendo que los diseños sean portátiles a cualquier plataforma.

2.5. Control inteligente

Las primeras aplicaciones del control con retroalimentación datan del periodo 300 a. C. en Grecia, con el desarrollo de los mecanismos reguladores con flotador: el reloj de agua de Ktesibios usaba un regulador con flotador; y una lámpara de aceite inventada por Philon, aproximadamente en el año 250 a. C., usaba un regulador de flotador para mantener un nivel constante de aceite (Mayr, 1970).

De ahí, el control automático ha desempeñado un papel vital en el avance de la ingeniería y la ciencia, a tal grado que se ha convertido en una parte importante de los procesos modernos industriales y de fabricación. Esto debido a que el avance en la teoría y la práctica en el control automático proporciona los medios

para conseguir un comportamiento óptimo en los sistemas dinámicos, mejora la productividad, simplifica el trabajo de muchas operaciones manuales y rutinarias, así como de otras actividades (Ogata, 2003).

En nuestros tiempos, el incremento de las demandas tecnológicas ha generado sistemas muy complejos que requieren controladores altamente sofisticados para asegurar alto desempeño dentro de condiciones adversas. Éstas, y otras condiciones de control, no se pueden cumplir con controladores convencionales.

La adquisición del conocimiento adecuado del sistema, en ocasiones, es problemática o impráctica debido a la complejidad del sistema y al hecho de que la estructura y los parámetros, en muchos sistemas, cambian de manera significativa e impredecible con el tiempo. Es, bajo estas condiciones, donde Antsaklis y Passino (1993) afirman que se utilizan las técnicas de control inteligente.

2.5.1. Preámbulo

El control clásico consta esencialmente del tratamiento de información en un nivel numérico y actuación sobre el mundo, no apareciendo en él actividades 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 término de control inteligente se utiliza para hacer referencia a un control en el que las actividades cognoscitivas tienen especial importancia al realizar actividades de procesamiento de información con un cierto grado de abstracción (Galán, 1992).

2.5.2. Técnicas

De las técnicas de la inteligencia artificial usadas en control inteligente, según Galán *et al.* (2000), destacan:

- i.* Redes Neuronales.
- ii.* Lógica Difusa.
- iii.* Sistemas Expertos
- iv.* Algoritmos Genéticos.

2.6. Redes neuronales artificiales

Haykin (1998) define a una red neuronal como una estructura distribuida, de procesamiento paralelo, formada de neuronas artificiales, interconectadas por un gran número de conexiones, las cuales son usados para almacenar conocimiento que está disponible para ser usado.

Asimismo, explica que éstas se asemejan al cerebro humano en dos aspectos:

- 1) El conocimiento es adquirido por la red a través del proceso de aprendizaje.
- 2) Las fuerzas de interconexión neuronal, conocidas como pesos sinápticos, son usadas para almacenar el conocimiento.

Las redes neuronales artificiales constituyen una de las áreas de la inteligencia artificial que ha despertado mayor interés en los últimos años. Esto, debido a que son, potencialmente, capaces de resolver problemas cuya solución por otros métodos convencionales resulta extremadamente difícil.

La cualidad más sobresaliente de las redes neuronales es que éstas son capaces de “aprender”. En lugar de programar a una red se le presenta una serie de ejemplos, a partir de los cuales ésta aprende las relaciones principales que están implícitas en la base de datos de entrenamiento. Puesto que estas relaciones pueden ser fuertemente no-lineales, las redes pueden constituir entonces una herramienta muy general y potente para modelar procesos industriales de cualquier grado de complejidad (Aguado, 2000).

2.6.1. La neurona artificial: inspiración biológica

Las neuronas reciben señales (*inputs*) a través de las dendritas, que son las terminaciones neuronales que actúan como receptores de las señales emitidas por otras neuronas, via conexiones sinápticas que pueden ser excitantes o inhibitoras. Una neurona contiene un potencial interno continuo llamado potencial de membrana; cuando éste excede un cierto valor umbral, la neurona puede transmitir todo su potencial por medio del axón. La figura 2.1 ilustra una neurona humana.

Se estima que el cerebro humano contiene más de cien mil millones (10^{11}) de neuronas y que hay más de 1000 sinápsis a la entrada y a la salida de cada neurona. Los modelos neuronales biológicos son mucho más complejos que los modelos computacionales actuales.

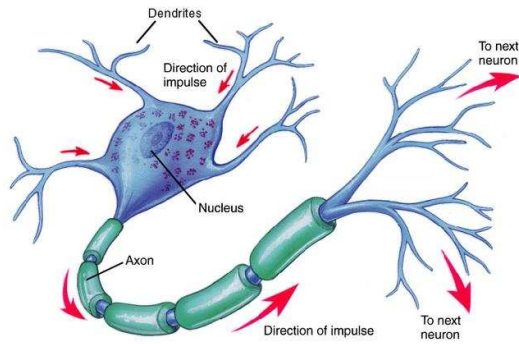


Figura 2.1: Diagrama de una neurona (Copyright ©2001 The McGraw-Hill Companies. All rights reserved.)

2.6.2. El perceptrón simple

Una neurona artificial es una herramienta diseñada para emular la forma en que el cerebro humano funciona. El modelo de neurona más conocido es de McCulloch-Pitts (Nanayakkara *et al.*, 2010). El perceptrón simple es el caso más sencillo de redes neuronales artificiales que presenta una sola neurona de cómputo. Fue desarrollado por Frank Rosenblatt en 1958 basándose en la regla de aprendizaje de Hebb y de los modelos de neuronas biológicas de McCulloch y Pitts.

En la figura 2.2 se presenta la arquitectura del perceptrón simple constituido por neuronas de entrada (cuadros) que simplemente almacenan entradas a la red, una neurona de procesamiento (círculo) y los canales de flujo de información o interconexiones entre las neuronas (flechas). Las variables $x_1^p, x_2^p, \dots, x_n^p$ pueden ser considerada como las entradas correspondientes en las dendritas a través de las conexiones sinápticas $w_1^p, w_2^p, \dots, w_n^p$, mientras que la variable y^p podría considerarse la señal transmitida mediante el axón (McCulloch y Pitts, 1943). Además, en la figura 2.2 se observa que existen dos etapas de procesamiento para cada neurona: la excitación y la activación.

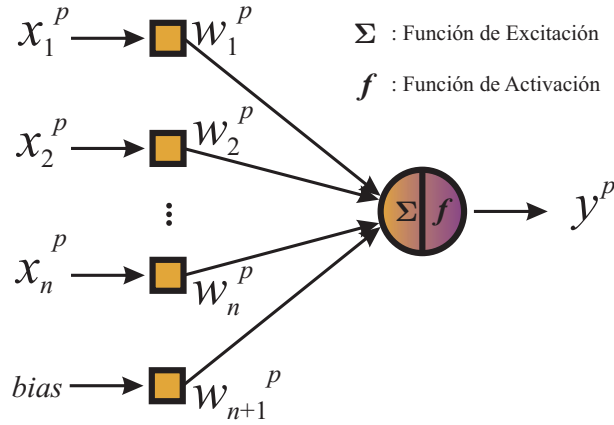


Figura 2.2: Arquitectura del perceptrón simple

Función de excitación

La función de excitación, Σ , presentada a la figura 2.2 en cada presentación p , de manera generalizada para una red con j neuronas de procesamiento, es la suma de cada variable de entrada x_i^p multiplicada por su peso de interconexión $w_{i,j}^p$ con la neurona j , como se muestra en la ecuación (2.1). Si el peso es positivo, la conexión se denomina excitatoria; si es negativo, se denomina inhibitoria.

$$s_j^p = \sum_{i=1}^n w_{i,j}^p \cdot x_i^p + w_{n+1,j}^p \quad (2.1)$$

En la ecuación (2.1), w_{n+1}^p corresponde al peso de la salida x_{n+1}^p definida como *bias*, la cual siempre tomará el valor de 1 con el fin de que la neurona de procesamiento pueda modelar salidas distintas de 0 cuando todas las entradas sean 0.

Función de activación

La función de activación, \mathbf{f} , tiene como objetivo limitar el rango de valores de la respuesta de la neurona. Generalmente los rangos de valores se limitan a $[0, 1]$ o $[-1, 1]$; sin embargo, otros rangos son posibles de acuerdo a la aplicación o problema a resolver (Martín del Brío, 2002).

Existen diversas funciones de activación y la decisión entre una u otra dependerá nuevamente de la aplicación o problema a resolver. Las funciones de activación comúnmente utilizadas, y con las cuales se han obtenido resultados satisfactorios en diversas aplicaciones, son las que se enlistan a continuación, y en la figura 2.3 se muestran de manera gráfica.

Tabla 2.2: Funciones de activación comúnmente usadas

Función	Ecuación
Limitador fuerte (<i>hardlim</i>)	$h_j^p = \begin{cases} 1 & \text{si } s_j^p \geq 0 \\ 0 & \text{si } s_j^p < 0 \end{cases}$
Limitador fuerte simétrico (<i>hardlims</i>)	$h_j^p = \begin{cases} -1 & \text{si } s_j^p < 0 \\ 0 & \text{si } s_j^p = 0 \\ 1 & \text{si } s_j^p > 0 \end{cases}$
Lineal positiva (<i>poslin</i>)	$h_j^p = \begin{cases} 1 & \text{si } s_j^p \geq 0 \\ s_j^p & \text{si } s_j^p < 0 \end{cases}$
Lineal (<i>purelin</i>)	$h_j^p = s_j^p$
Lineal saturado (<i>satlin</i>)	$h_j^p = \begin{cases} 0 & \text{si } s_j^p < 0 \\ s_j^p & \text{si } 0 \leq s_j^p \leq 1 \\ 1 & \text{si } s_j^p > 1 \end{cases}$
Lineal saturado simétrico (<i>satlins</i>)	$h_j^p = \begin{cases} -1 & \text{si } s_j^p < -1 \\ s_j^p & \text{si } -1 \leq s_j^p \leq 1 \\ 1 & \text{si } s_j^p > 1 \end{cases}$
Sigmoidal logarítmica (<i>logsig</i>)	$h_j^p = \frac{1}{1+e^{-as_j^p}} \quad s \in \mathfrak{R}, \quad a > 0$
Tangente sigmoidal hiperbólica (<i>tansig</i>)	$h_j^p = \frac{e^{as_j^p} - e^{-as_j^p}}{e^{as_j^p} + e^{-as_j^p}} \quad s \in \mathfrak{R}, \quad a > 0$

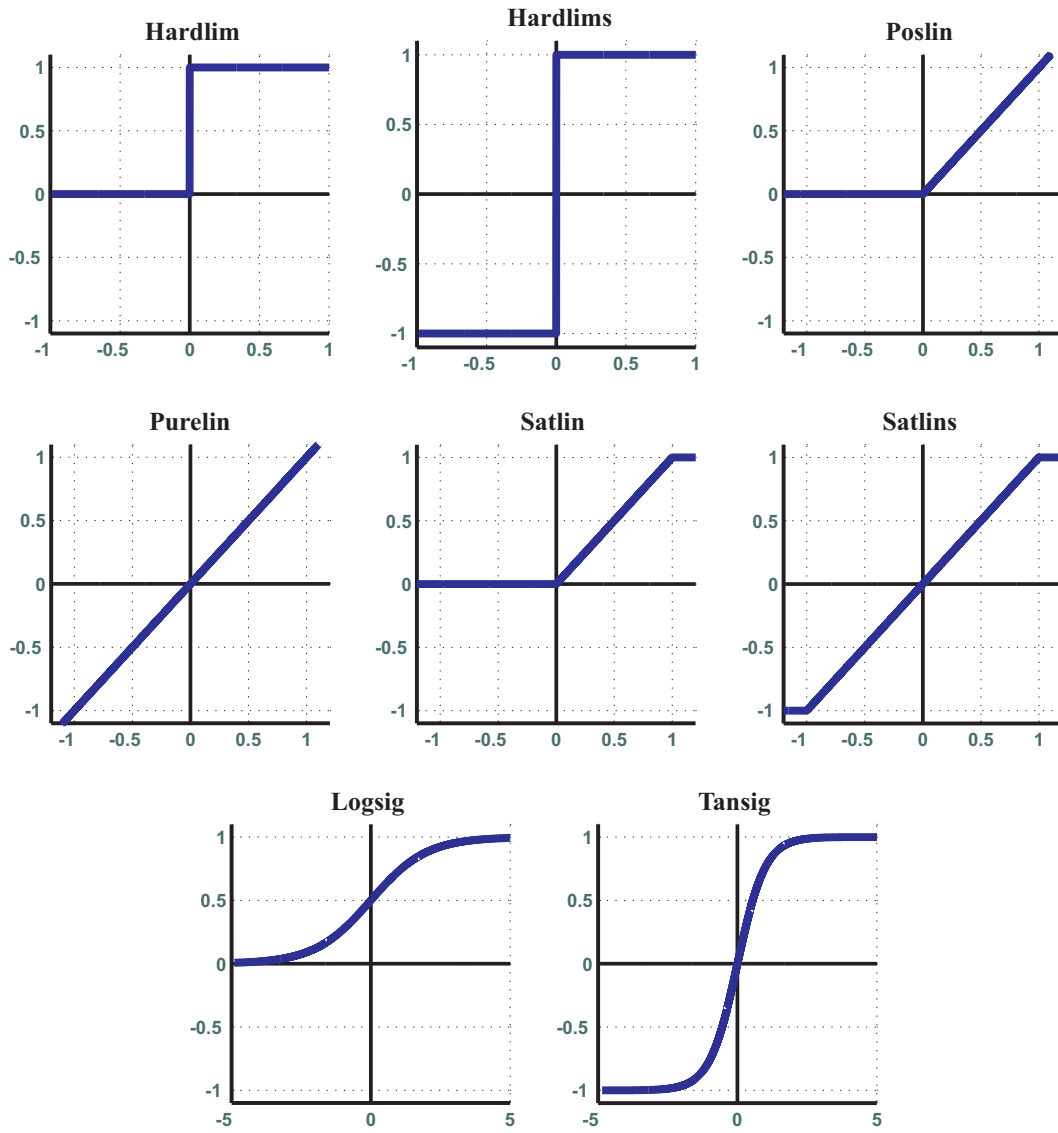


Figura 2.3: Funciones de activación

Alcances del perceptrón simple

El perceptrón simple es capaz de aprender una función f de \mathbb{R}^n en $\{0,1\}$, para funciones lógicas, que aplica un patrón de entrada $\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^T \in \mathbb{R}^n$ en la salida deseada $y \in \{0,1\}$; es decir, $f(\mathbf{x}) = y$. La información disponible para esta función está dada por p pares de patrones de entrenamiento, comúnmente llamados presentaciones, como se da en la ecuación (2.2).

$$\{\mathbf{x}^1, y^1\}, \{\mathbf{x}^2, y^2\}, \dots, \{\mathbf{x}^p, y^p\} \quad (2.2)$$

Donde, en la ecuación (2.2), $\mathbf{x} \in \mathbb{R}^n$ y $f(\mathbf{x}^i) = y^i \in \{0,1\}$, para $i = 1, 2, \dots, p$.

Dicha función f realiza una partición en el espacio \mathbb{R}^n de patrones de entrada. Por una parte estarían los patrones con salida $y = 1$ y, por otra parte, los patrones con salida $y = 0$. Por lo tanto, se dice que la función f realiza una clasificación de patrones de entrada, siempre y cuando estos patrones sean linealmente separables.

Conjuntos linealmente separables

Se dice que dos conjuntos de puntos A y B son linealmente separables en un espacio n -dimensional si existen $n + 1$ números reales w_1, w_2, \dots, w_n , y θ , de manera que cada punto $(x_1, x_2, \dots, x_n) \in A$ satisface $\sum_{i=1}^n w_i x_i \geq \theta$ y cada punto $(x_1, x_2, \dots, x_n) \in B$ satisface $\sum_{i=1}^n w_i x_i < \theta$.

La figura 2.4 ilustran los conjuntos de puntos A y B linealmente separables en el espacio \mathbb{R}^2 para $n = 2$.

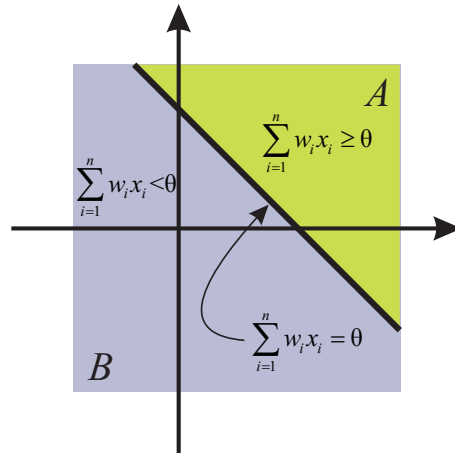


Figura 2.4: Semiespacios que define el perceptrón

Ejemplo de funciones f linealmente separables son, entre otras, la función lógica AND cuyo valor de salida es igual a 1 (punto rojo) cuando ambas entradas x_1 y x_2 son igual a 1, de lo contrario, el valor de la salida es igual a 0 (puntos azules), como se muestra en la figura 2.5 (a); y la función lógica OR cuyo valor de salida es 1, como se observa en la figura 2.5 (b), cuando al menos una de las entradas x_1 y x_2 es 1.

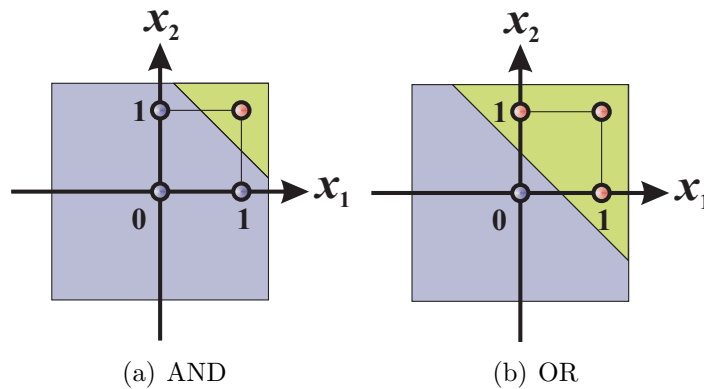


Figura 2.5: Funciones lógicas separables linealmente

A su vez, las funciones lógicas XOR y XNOR no pueden ser aprendidas por un único perceptrón, pues se requieren al menos dos líneas, como se muestra en la figura 2.6, para separar las clases $\{0, 1\}$. Debe utilizarse al menos una capa adicional de perceptrones para permitir su aprendizaje.

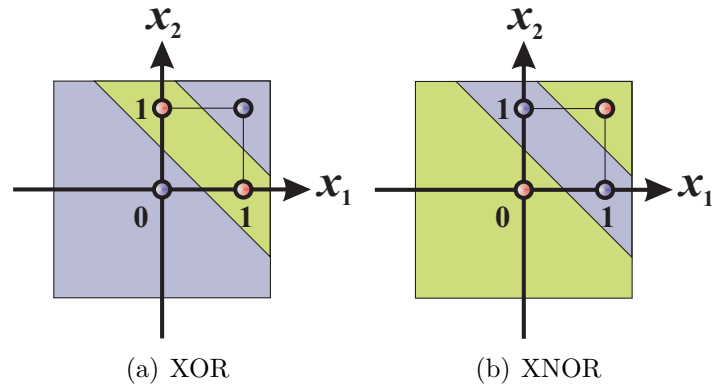


Figura 2.6: Funciones lógicas no separables linealmente

2.6.3. Arquitectura de red

El comportamiento de la red está determinado por su topología, los pesos sinápticos de las conexiones y la función característica de activación de las neuronas. En términos muy generales, los siguientes parámetros son importantes para definir la arquitectura de una red neural:

- ◆ Número de capas.
- ◆ Número de neuronas en cada capa.
- ◆ Tipo de interconexión entre las neuronas.

La red neuronal está constituida esencialmente por la interconexión de dos, o más neuronas. La definición de arquitectura es un punto importante en el diseño de una red neuronal, por que ella restringe un tipo de problema que puede ser tratado.

La primera clasificación la da el número de capas, dividiéndose en:

- I) **Redes de una capa.**
- II) **Redes multicapas** (*layered networks*), las cuales se dividen en:
 - a. Capa de entrada.
 - b. Capa(s) intermedia(s) u oculta(s).
 - c. Capa de salida.

En la figura 2.7 se muestra la construcción de una red neuronal de múltiples capas.

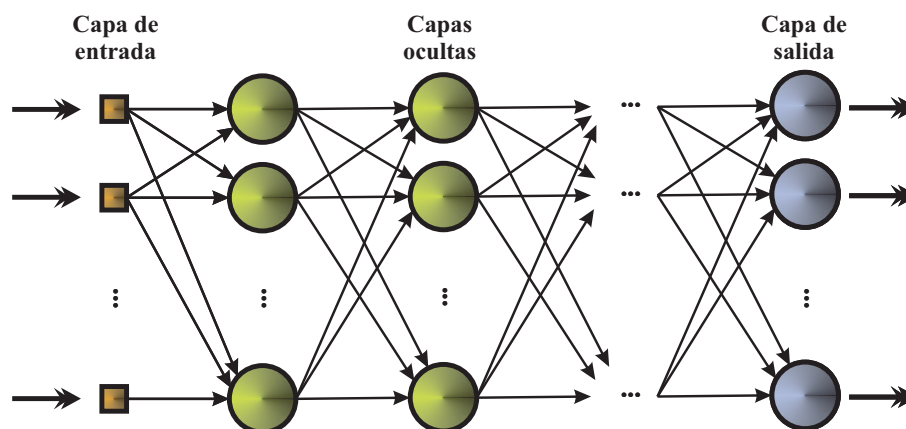


Figura 2.7: Arquitectura de múltiples capas.

La segunda clasificación la da el flujo de las señales, dividiéndose principalmente en dos tipos:

- 1) **Redes unidireccionales o de propagación hacia adelante** (*feedforward*): En la arquitectura de estas redes multicapa ninguna salida de las neuronas de procesamiento es entrada a las demás neuronas de la misma capa o de capas previas, como se observa en la figura 2.8. La información circula en un único sentido, desde las neuronas de entrada hasta las neuronas de salida de la red, denominada conexión hacia adelante, figura 2.10 (a).
- 2) **Redes de propagación hacia atrás** (*feedback*): En la arquitectura de esta red neuronal las salidas de las neuronas de procesamiento pueden servir de entradas a otras neuronas, como se aprecia en la figura 2.9, ya sea del mismo nivel (conexiones laterales, figura 2.10 (b)) o de niveles previos (conexiones hacia atrás, figura 2.10 (c)). Las redes neuronales que presentan lazos cerrados se denominan sistemas recurrentes. En las redes compuestas por una única capa de neuronas, se establecen conexiones laterales y, en ocasiones, conexiones autorrecurrentes, figura 2.10 (d).

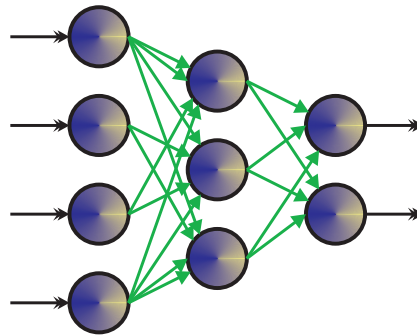


Figura 2.8: Red neuronal de propagación hacia adelante (*feedforward*)

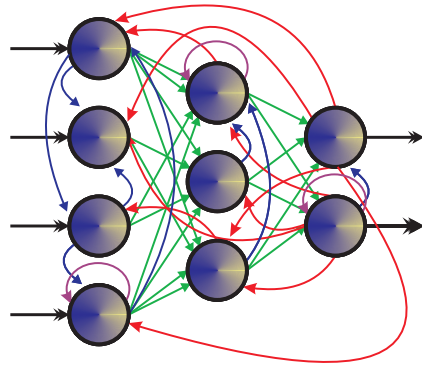


Figura 2.9: Red neuronal de propagación hacia atrás (*feedback*)

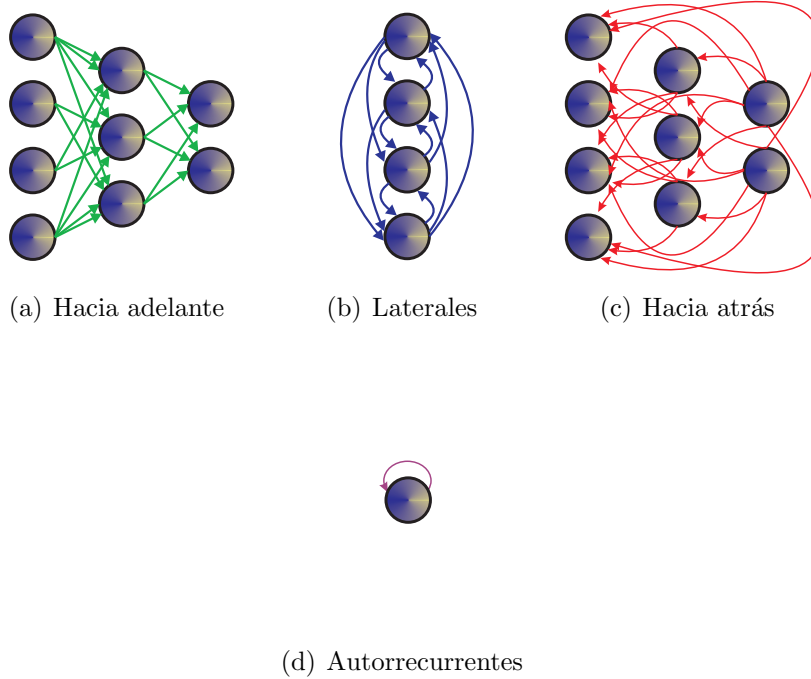


Figura 2.10: Tipos de conexiones de las redes neuronales

2.6.4. Características principales

En el ámbito de las redes neuronales artificiales, las características que destacan son las que se exponen en la tabla 2.3.

Tabla 2.3: Características de una red neuronal

Característica	Descripción
◇ Aprendizaje adaptivo:	Las redes neuronales artificiales aprenden a realizar tareas a partir de un conjunto de datos dados en el proceso de aprendizaje.
◇ Auto-organización:	Pueden crear su propia representación u organización de la información recibida.
◇ Operación en tiempo real:	Las operaciones realizadas pueden ser llevadas a cabo por computadoras en paralelo, o dispositivos de hardware especiales que aprovechan esta capacidad.
◇ Tolerancia a fallos parciales:	La destrucción parcial de una red daña parcialmente el funcionamiento de la misma, pero no la destruye completamente. Esto es debido a la redundancia de la información contenida.

2.6.5. Algoritmos de aprendizaje

El aprendizaje se refiere al proceso por el cual los parámetros (pesos sinápticos) de una red neuronal artificial se adaptan (aprenden) por estimulación del medio ambiente, mediante un conjunto bien definido de reglas para actualizar los pesos sinápticos entre dos neuronas, denominado entrenamiento. Existen muchos algoritmos de aprendizaje, cada uno sirve para determinar redes neuronales.

Entre los principales algoritmos de aprendizaje se tienen:

- i)* **Corrección de Error.** Es un algoritmo muy conocido, basado en la regla Delta que busca minimizar la función de error usando un gradiente descendente. Este es el principio usado en el algoritmo *Back-Propagation*, utilizado para el entrenamiento de redes de múltiples capas como el *Multilayer-Perceptron* (Freeman y Skapura, 1991).
- ii)* **Hebbian.** Son dos neuronas que están simultáneamente activas a conexiones entre ellas, la cual debe ser fortalecida o de lo contrario podría debilitarse (Hebb, 1949). Este algoritmo es utilizado en el Modelo de Hopfield (Hopfield, 1982).
- iii)* **Competitivo.** Dos neuronas de una capa compiten entre sí por el privilegio de permanecer activas, tal que una neurona con mayor actividad será la única que participará en el proceso de aprendizaje. Es usado en mapas de Kohonen (Kohonen, 1988) y en redes ART (Carpenter *et al.*, 1992).
- iv)* **Boltzmann.** Es una regla de aprendizaje estocástico obtenido a partir de principios de la teoría de la información y la termodinámica. El objetivo del aprendizaje de Boltzmann es ajustar los pesos de conexión de tal forma que el estado de las unidades visibles satisfaga una distribución de probabilidades deseada en particular (Acidey *et al.*, 1985).

Mientras que el paradigma de aprendizaje hace referencia al modelo del medio ambiente en el cual la red neuronal artificial opera, y se clasifican en:

- a)* **Supervisado:** se utiliza un agente externo que indica a la red la respuesta deseada para el patrón de entrada.

b) **No supervisado (auto-organización)**: No existe un agente externo indicando la respuesta deseada para los patrones de entrada. Este tipo de aprendizaje es utilizado en los modelos de Mapas de Kohonen (Kohonen, 1988), redes ART1, ART2 (Carpenter *et al.*, 1992).

▷ **Por refuerzos**: Es una variante de aprendizaje supervisado a la cual se informa a la red solamente una critica de corrección de salida de red y no la respuesta correcta en sí.

Algoritmo de retro-propagación del error

Dada una red neuronal de multicapa, con una sola capa oculta de neuronas, donde el número de neuronas en la capa de entrada está dado por nE , el número de neuronas en la capa oculta es dado por nO y nS define las neuronas que conforman la capa de salida, tal como se puede ver en la figura 2.11.

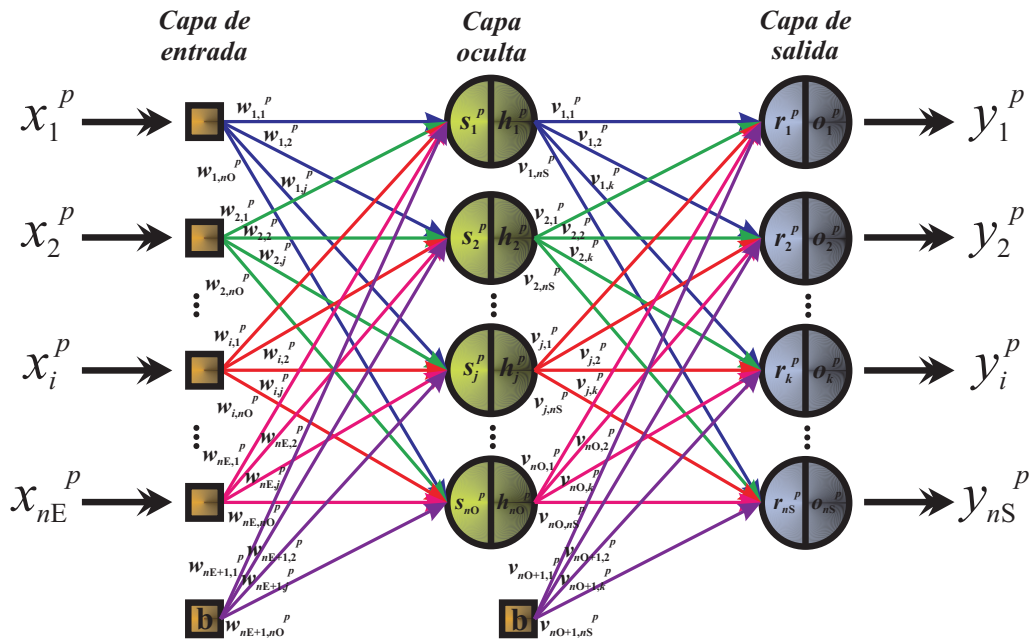


Figura 2.11: Arquitectura de una red neuronal de tres capas

La función de excitación s_j^p para la muestra p y la j -ésima neurona de la capa oculta está dada por la ecuación (2.3), donde el peso sináptica $w_{i,j}$ corresponde a la interconexión de la i -ésima neurona de la capa de entrada con la j -ésima neurona de la capa oculta. De este modo, la función de activación h_j^p de la j -ésima neurona de la capa oculta y la muestra p , por lo tanto, es una función de s_j^p tal como lo expresa la ecuación (2.4).

$$s_j^p = \sum_{i=1}^{nS} x_i^p w_{i,j} + w_{nS+1,j}^p = f(w_{i,j}^p) \quad (2.3)$$

$$h_j^p = f(s_j^p) \quad (2.4)$$

A su vez, la función de excitación r_k^p para la muestra p y la k -ésima neurona de la capa de salida está dada por la ecuación (2.5), donde el peso sináptica $v_{j,k}$ corresponde a la interconexión de la j -ésima neurona de la capa oculta con la k -ésima neurona de la capa de salida. Así, la función de activación o_k^p de la k -ésima neurona de la capa de salida y la muestra p es función de r_k^p tal como se ve en la ecuación (2.6).

$$r_k^p = \sum_{j=1}^{nO} h_j^p v_{j,k} + v_{nO+1,k}^p = f(v_{j,k}^p) \quad (2.5)$$

$$o_k^p = f(r_k^p) \quad (2.6)$$

El error, por definición, es la diferencia entre el valor deseado y el valor que se está leyendo. Así pues, la ecuación 2.7 expresa la función del error de la k -ésima neurona de la capa de salida.

$$e_k^p = y_k^p - o_k^p \quad (2.7)$$

Criterio de minimización del error

El criterio a minimizar en la muestra p se define por la ecuación (2.8), o también en la ecuación (2.9) (Aguado, 2000).

$$E_p = \frac{1}{2} \sum_{k=1}^{nS} (e_k^p)^2 = \frac{1}{2} \sum_{k=1}^{nS} (y_k^p - o_k^p)^2 \quad (2.8)$$

$$E_p = \frac{1}{2} \sum_{k=1}^{nS} (y_k^p - f(r_k^p))^2 \quad (2.9)$$

El proceso de aprendizaje de la red neuronal consiste, entonces, en presentar secuencialmente las entradas x_i^p con $p = 1, 2, \dots, nP$ e $i = 1, 2, \dots, nE$, calcular las salidas de la red o_k^p con $k = 1, 2, \dots, nO$ y $p = 1, 2, \dots, nP$, los errores e_k^p y el criterio E_p y aplicar algún procedimiento de minimización de la función E_p con respecto a los coeficientes de pesos $w_{i,j}$ y $v_{j,k}$ de manera que estos se vayan aproximando paulatinamente a los valores que garantizan un error mínimo entre las salidas de la red o_k y los datos de salida y_k .

El criterio de minimización de la función E_p con respecto a los coeficientes de pesos $w_{i,j}$ y $v_{j,k}$ que será empleado es el método conocido como “descenso más rápido” (*steepest descent*) que consiste en moverse siempre en la dirección del negativo del gradiente de la función E_p con respecto a los coeficientes de pesos $w_{i,j}$ y $v_{j,k}$.

El gradiente de la función E_p es un vector unidimensional cuyas componentes son las derivadas parciales $\frac{\partial E_p}{\partial v_{j,k}^p}$ y $\frac{\partial E_p}{\partial w_{i,j}^p}$, como lo muestra la ecuación (2.10).

$$\nabla E_p = \begin{bmatrix} \frac{\partial E_p}{\partial v_{j,k}^p} \\ \frac{\partial E_p}{\partial w_{i,j}^p} \end{bmatrix} \quad (2.10)$$

De este modo, el resultado de la derivada parcial $\frac{\partial E_p}{\partial v_{j,k}^p}$ que conforma el gradiente de la función E_p , está dada en la ecuación (2.11)¹.

$$\frac{\partial E_p}{\partial v_{j,k}^p} = - \sum_{k=1}^{nS} (y_k^p - o_k^p) o_k^p (1 - o_k^p) h_j^p \quad (2.11)$$

Si se define $\delta_k^p = \sum_{k=1}^{nS} (y_k^p - o_k^p) o_k^p (1 - o_k^p)$, la ecuación (2.11) puede ser expresada de la forma:

$$\frac{\partial E_p}{\partial v_{j,k}^p} = -\delta_k^p h_j^p \quad (2.12)$$

¹Este resultado se calcula suponiendo una función de activación *sigmoide*, para las neuronas de la capa de entrada y de salida.

Por otro lado, el resultado de la derivada parcial $\frac{\partial E_p}{\partial w_{i,j}^p}$ es presentado por la ecuación (2.13).

$$\frac{\partial E_p}{\partial w_{i,j}^p} = -\delta_k^p \sum_{j=1}^{nO} v_{j,k}^p h_j^p (1 - h_j^p) x_i^p \quad (2.13)$$

Definiendo, ahora, $\Delta_j^p = \delta_k^p \sum_{j=1}^{nO} v_{j,k}^p h_j^p (1 - h_j^p)$, la ecuación (2.13) puede ser expresada de la forma:

$$\frac{\partial E_p}{\partial w_{i,j}^p} = -\Delta_j^p x_i^p \quad (2.14)$$

Teniendo en cuenta las ecuaciones (2.12) y (2.14), el resultado del gradiente de la función E_p definido en la ecuación (2.10) está dado por la ecuación (2.15).

$$\nabla E_p = \begin{bmatrix} -\delta_k^p h_j^p \\ -\Delta_j^p x_i^p \end{bmatrix} \quad (2.15)$$

El algoritmo de retro-propagación aplicado al entrenamiento de las neuronas de una red de 3 capas consiste en moverse en la dirección del negativo del gradiente, dando un paso de actualización de los valores de los coeficientes $v_{j,k}^p$ y $w_{i,j}^p$. Es decir:

$$\begin{bmatrix} v_{j,k}^p \\ w_{i,j}^p \end{bmatrix} = \begin{bmatrix} v_{j,k}^{p-1} \\ w_{i,j}^{p-1} \end{bmatrix} - \eta \nabla E_p \quad (2.16)$$

$$\begin{bmatrix} v_{j,k}^p \\ w_{i,j}^p \end{bmatrix} = \begin{bmatrix} v_{j,k}^{p-1} \\ w_{i,j}^{p-1} \end{bmatrix} - \eta \begin{bmatrix} -\delta_k^p h_j^p \\ -\Delta_j^p x_i^p \end{bmatrix}$$

Donde de la ecuación (2.16) son evidentes las relaciones presentadas en las ecuaciones (2.17) y (2.18) y a η se le denomina constante de aprendizaje.

$$v_{j,k}^p = v_{j,k}^{p-1} + \eta \delta_k^p h_j^p \quad (2.17)$$

$$w_{i,j}^p = w_{i,j}^{p-1} + \eta \Delta_j^p x_i^p \quad (2.18)$$

2.6.6. Control neuronal

Las redes neuronales, en control, se utilizan por su capacidad de aprender el comportamiento no lineal de las variables de un proceso. Esta capacidad se puede utilizar para el diseño de sistemas que funcionen como simulador, identificador de procesos o como sistema que optimiza el funcionamiento de un controlador (Omidvar y Elliot, 1997).

2.7. Lógica difusa

La lógica difusa² (o *fuzzy logic*) es considerada una generalización de la teoría general de conjuntos que permite que elementos de un universo tengan grados intermedios de pertenencia a un conjunto por medio de una función característica. Con esta idea se modifica el concepto bivalencial (0 y 1) de la lógica booleana, la cual pasa a ser un caso particular de los conjuntos difusos (Betza-beth, 2007). Es decir, es el área de la investigación especializada en la ciencia de la computación que permite matices de gris que no requieren que todo sea un extremo blanco o negro (Stair y Reynolds, 2000).

2.7.1. Comienzos

En 1965, L. A. Zadeh introduce la teoría de conjuntos difusos como un mecanismo para representar la ambigüedad e imprecisión de los conceptos empleados en el lenguaje natural (Zadeh, 1965). Estos conjuntos difusos fueron definidos como una extensión de los conjuntos clásicos capaz de modelar la vaguedad propia de los conceptos humanos.

A mediados de los 70's, llega la ampliación del concepto de conjuntos al de lógica, apareciendo la lógica difusa y sus aplicaciones a sistemas de control.

Hoy día, son muchas las aplicaciones, tanto industriales como domésticas, que hacen uso de este paradigma.

²No existe un consenso claro respecto a la mejor traducción del término *fuzzy*. Algunos autores optan por *difuso*, mientras que otros lo hacen por *borroso*.

2.7.2. Conjuntos difusos

La Teoría de Conjuntos Difusos se puede utilizar para representar expresiones lingüísticas de las que hace uso para describir conjuntos o algoritmos. Los conjuntos difusos son capaces de captar por sí mismos la vaguedad de las palabras o, inclusive, algunas frases. La habilidad humana de comunicarse mediante definiciones vagas o inciertas es un atributo importante de la inteligencia.

Variables lingüísticas

Una variable lingüística posee valores que bien pueden ser palabras o sentencias que se enmarcan en un lenguaje predeterminado. Cada una de estas palabras o términos se conocen como etiqueta lingüística, y se representan por medio de un conjunto difuso sobre lo que se denomina universo de discurso de la variable.

Para esclarecer este concepto, consideremos el siguiente ejemplo: la temperatura *normal* del cuerpo humano es de 36 °C. En un paciente que se reporte una lectura de su temperatura corporal de 35.9 °C, siguiendo la tendencia booleana de la lógica tradicional, no se podría afirmar que esta medición muestra una temperatura normal del cuerpo humano. Sin embargo, la temperatura puede catalogarse como “*baja*”, “*normal*”, “*alta*” o “*muy alta*” para un médico especializado considerando un cierto rango de valores, y es en base a ese rango que emite su diagnóstico.

Cada uno de esos términos es una etiqueta lingüística que puede definirse como un conjunto difuso. Los conjuntos que forman la variable se muestran en la figura 2.12.

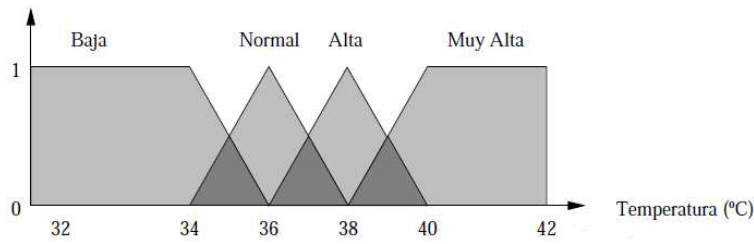


Figura 2.12: Definición de la variable lingüística *Temperatura*.

Función de pertenencia

Un conjunto nítido es un conjunto clásico que puede ser definido mediante el listado de todos sus elementos, o mediante el establecimiento de una condición que exprese los elementos que pertenecen a ese conjunto, como por ejemplo la condición dada por la ecuación (2.19).

$$A = \{x | a \leq x \leq b\} \tag{2.19}$$

Para este conjunto, se puede definir una función de pertenencia³ cero–uno, denotada por $\mu_A(x)$ de modo que este definida por la ecuación (2.20):

$$\mu_A(x) = \begin{cases} 0 & \text{si } x \in A \\ 1 & \text{si } x \notin A \end{cases} \tag{2.20}$$

De este modo, dado un conjunto nítido, y cualquier elemento se sabe si este pertenece o no al conjunto. En cambio, en un conjunto difuso, la transición entre pertenecer o no pertenecer a él, es gradual. Esta transición progresiva va a

³También llamada *función característica*

venir definida por su función de pertenencia.

Es decir, en un conjunto difuso, F , se puede definir un intervalo de discurso, U , caracterizado por una función de pertenencia que toma valores en el intervalo $[0, 1]$ y se denota por la ecuación (2.21). El valor asignado por $\mu_F(u)$ corresponde al grado en el cual el valor u tiene el atributo F ; o visto de otra manera, la función $\mu_F(u)$ nos indica cuál es el grado de pertenencia de u al atributo F .

$$\mu_F(u) : U \rightarrow [0, 1] \quad (2.21)$$

Como se observa en la figura 2.13, un conjunto difuso se puede ver como una generalización de un conjunto nítido, donde la función de pertenencia sólo toma los valores 0 ó 1.

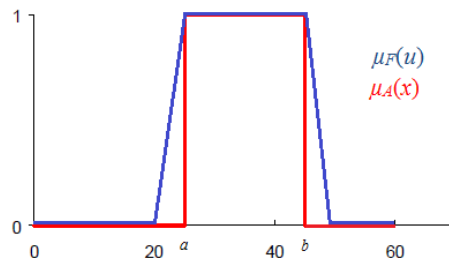


Figura 2.13: Función de pertenencia de un conjunto nítido y difuso.

El conjunto difuso F en U puede ser representado por un conjunto de pares ordenados compuestos por el elemento genérico u y su función de pertenencia $\mu_F(u)$, mediante la ecuación (2.22):

$$F = \{(u, \mu_F(u)) \mid u \in U\} \quad (2.22)$$

Para cada conjunto difuso, existe asociada una función de pertenencia para sus elementos, que indican en qué medida el elemento forma parte de ese conjunto difuso.

Existen, dependiendo de su forma, diferentes tipos de función de pertenencia, algunas de ellas se muestran en la figura 2.14, y enseguida se describe cómo está constituida cada una de ellas:

- ◆ **Triangular.** Se puede definir con respecto a tres parámetros $\{a, b, c\}$ que determinan las coordenadas en x de sus esquina, mediante la ecuación (2.23):

$$\mu_{\text{triangulo}}(x; a, b, c) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{x-a}{b-a} & \text{si } a \leq x \leq b \\ \frac{c-x}{c-d} & \text{si } b \leq x \leq c \\ 0 & \text{si } c \leq x \end{cases} \quad (2.23)$$

- ◆ **Trapezoidal.** Se puede establecer una función trapezoide en base a sus cuatro parámetros $\{a, b, c, d\}$, mediante la ecuación (2.24):

$$\mu_{\text{trapezoide}}(x; a, b, c, d) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{x-a}{b-a} & \text{si } a \leq x \leq b \\ 1 & \text{si } b \leq x \leq c \\ \frac{d-x}{d-c} & \text{si } c \leq x \leq d \\ 0 & \text{si } d \leq x \end{cases} \quad (2.24)$$

- ◆ **Gaussiana.** Una función gaussiana se caracteriza por su centro y su radio $\{c, \sigma\}$, definida por la ecuación (2.25):

$$\mu_{\text{gaussiana}}(x; c, \sigma) = e^{-\left(\frac{x-c}{\sigma}\right)^2} \quad (2.25)$$

- ◆ **Campana generalizada.** Como caso más general de función gradual, podemos definir una campana con tres parámetros $\{a, b, c\}$ que definen la amplitud de la campana $2a$, su centro c , y la pendiente lateral $-\frac{b}{2a}$, mediante la ecuación (2.26):

$$\mu_{\text{campana}}(x; a, b, c) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}} \quad (2.26)$$

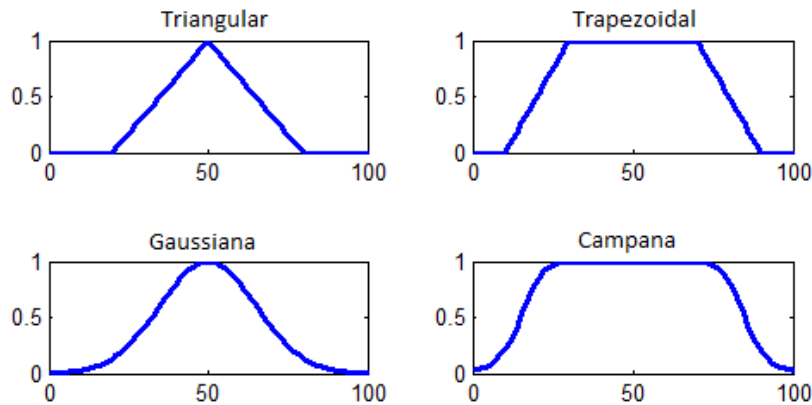


Figura 2.14: Funciones de pertenencia más comunes.

2.7.3. Estructura de un controlador difuso

El control difuso proporciona una metodología formal para la representación, manipulación e implementación de un conocimiento humano heurístico acerca de cómo controlar un sistema. En la figura 2.15 se presenta el diagrama a

bloques de un sistema de control difuso en lazo cerrado, donde las salidas de la planta están definidas como $y(t)$, las entradas como $u(t)$ y la referencia por $r(t)$. Se observan, además, cuatro componentes principales.

- i)* **Base de Reglas:** Contiene el conocimiento experto resumido en un conjunto de reglas del tipo: **if _ then**.
- ii)* **Máquina de Inferencias:** Simula el proceso de toma de decisiones del experto, tomando en cuenta la base de reglas y el conocimiento difuso del proceso.
- iii)* **Fusificador:** Es una interfaz que transforma la información de entrada al controlador en información lingüística que puede ser interpretada por la base de reglas y la máquina de inferencias.
- iv)* **Defusificador:** Esta interfaz convierte las conclusiones de la máquina de inferencias en acciones de control.

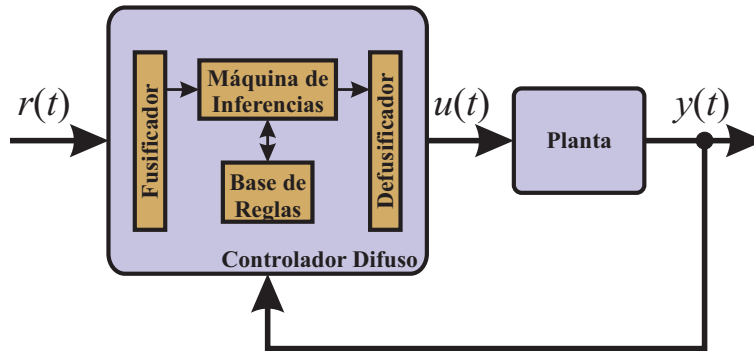


Figura 2.15: Arquitectura del controlador difuso

Básicamente, el controlador difuso se puede ver como una máquina que toma decisiones de forma artificial que opera en un sistema de lazo cerrado en tiempo real. Es decir, se toman los datos de salida de la planta $y(t)$, se comparan

con la referencia de entrada $r(t)$, y entonces despide cuál debiera ser la entrada de la planta $u(t)$ para asegurarse que el objetivo sea alcanzado.

Base de reglas

Su poniendo que un experto proporciona una descripción del mejor control de la planta en un lenguaje natural, se toma esta descripción “lingüística” y se carga al controlador. La descripción lingüística proporcionada por el experto puede descomponerse en varias partes. Estas serán “variables lingüísticas” que describen cada una de las entradas y salidas del controlador difuso que varían en el tiempo. Suponiendo un sistema cuyas variables lingüísticas se describan como en la tabla 2.4.

Tabla 2.4: Descripción de las variables lingüísticas

Variable lingüística	Descripción
“error”	$e(t)$
“cambio del error”	$\frac{de(t)}{dt}$
“acción de control”	$u(t)$

Así como $e(t)$ toma valores reales, las variables lingüísticas asumen valores lingüísticos. Se asume, entonces, que las variables “error”, “cambio del error” y “acción de control” toman cada uno de los valores lingüísticos representados en la tabla 2.5.

Tabla 2.5: Valores de las variables lingüísticas

Magnitud de la variable lingüística	Codificación
“negativo grande”	-2
“negativo pequeño”	-1
“cero”	0
“positivo pequeño”	1
“positivo grande”	2

Las variables y los valores lingüísticos proporcionan un lenguaje experto para expresar sus ideas acerca del proceso de generación de decisiones del control en el contexto del marco establecido por la selección de entradas y salidas del controlador difuso. Para llegar al fusificador se requiere un conjunto de reglas como:

- ◇ Si el **error** es *negativo grande* y el **cambio del error** es *negativo grande*, entonces la **acción de control** es *positivo grande*.

El número de reglas posibles se calcula mediante la ecuación (2.27), donde n_{\max_r} representa el número máximo de reglas, n_{vl} el número de valores lingüísticos y n_{ve} el número de variables de entrada. Para el ejemplo se han planteado 5 posibles valores lingüísticos (“*negativo grande*”, “*negativo pequeño*”, “*cero*”, “*positivo pequeño*” y “*positivo grande*”) para las 2 variables de entrada (“**error**” y “**cambio del error**”, la “**acción de control**” es una variable de salida). Por tanto, se tiene: $n_{\max_r} = 5^2 = 25$, como se muestra en la tabla 2.6.

$$n_{\max_r} = n_{vl}^{n_{ve}} \tag{2.27}$$

Tabla 2.6: Base de reglas

Acción de Control u		Cambio en el error \dot{e}				
		-2	-1	0	1	2
Error e	-2	2	2	2	1	0
	-1	2	2	1	0	-1
	0	2	1	0	-1	-2
	1	1	0	-1	-2	-2
	2	0	-1	-2	-2	-2

La tabla 2.6 representa, además, el conocimiento abstracto que el experto tiene sobre cómo controlar el sistema dado el error y su derivada como entradas. Nótese la diagonal de ceros y la simetría de la matriz que conforma la base de reglas.

Fusificador

Si se utilizan funciones de pertenencia triangulares para llevar a cabo el proceso de fusificación de la entrada, tal como la que se muestra en la figura 2.16, es posible describir el conjunto difuso mediante el ancho, w , que es la diferencia entre los valores en la base de la función de pertenencia y el centro, b , que es el valor donde la pertenencia, μ , es igual a 1. De este modo, mediante la ecuación (2.28) es posible obtener el valor de pertenencia en el instante t .

$$\mu = \begin{cases} \frac{2(t-b)}{w} + 1 & \text{si } t_1 \leq t \leq b \\ \frac{2(b-t)}{w} + 1 & \text{si } b \leq t \leq t_2 \end{cases} \quad (2.28)$$

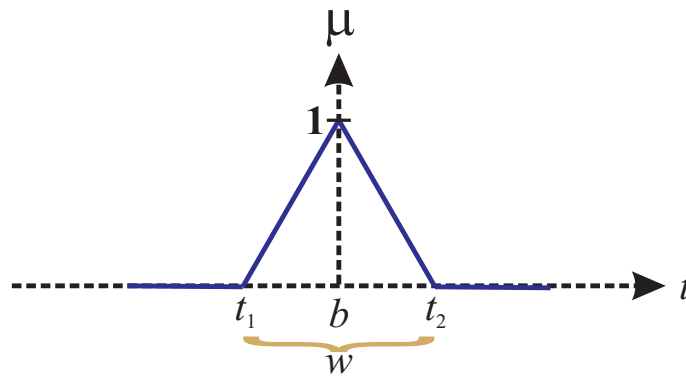


Figura 2.16: Funciones de pertenencia triangular

De este modo se obtiene el valor de pertenencia para **error**, $\mu_e(t)$, y **cambio del error**, $\mu_{\frac{de(t)}{dt}}$, en cada uno de los conjuntos difusos: *negativo grande* (-2), *negativo pequeño* (-1), *cero* (0), *positivo grande* (1) y *positivo pequeño* (2). Es decir, $\mu_e(t)$ y $\mu_{\frac{de(t)}{dt}}$ pueden ser expresadas en forma vectorial por las ecuaciones (2.29) y (2.29), respectivamente.

$$\mu_e(t) = \left[\mu_{e_{\{-2\}}}(t) \quad \mu_{e_{\{-1\}}}(t) \quad \mu_{e_{\{0\}}}(t) \quad \mu_{e_{\{1\}}}(t) \quad \mu_{e_{\{2\}}}(t) \right] \quad (2.29)$$

$$\mu_{\frac{de(t)}{dt}} = \left[\mu_{\frac{de_{\{-2\}}(t)}{dt}} \quad \mu_{\frac{de_{\{-1\}}(t)}{dt}} \quad \mu_{\frac{de_{\{0\}}(t)}{dt}} \quad \mu_{\frac{de_{\{1\}}(t)}{dt}} \quad \mu_{\frac{de_{\{2\}}(t)}{dt}} \right] \quad (2.30)$$

Máquina de inferencias

En la etapa de inferencias, el valor de la pertenencia de la **acción de control**, $\mu_u(t)$, en función de $\mu_e(t)$ y $\mu_{\frac{de(t)}{dt}}$ que debe aplicarse en cada reglas se calcula, principalmente, mediante los dos criterios que se describen a continuación:

- i*) **El mínimo:** Establece que el porcentaje de la función de pertenencia de la salida que debe aplicarse en cada regla depende del valor de pertenencia mínimo entre la pertenencia del error y la pertenencia del cambio del error, tal como se muestra en la ecuación (2.31).
- ii*) **El producto:** Establece que el porcentaje de la función de pertenencia de la salida que debe aplicarse en cada regla lo determina el producto del valor de pertenencia del error y la pertenencia del cambio del error, tal como se muestra en la ecuación (2.32).

$$\mu_{u_{\{k\}}}(t) = \min \left(\mu_{e_{\{i\}}}(t), \mu_{\frac{de_{\{j\}}}{dt}}(t) \right) \quad (2.31)$$

$$\mu_{u_{\{k\}}}(t) = \left(\mu_{e_{\{i\}}}(t) \right) \cdot \left(\mu_{\frac{de_{\{j\}}}{dt}}(t) \right) \quad (2.32)$$

En las ecuaciones (2.31) y (2.32), $i, j = 1, 2, \dots, 5$ representan el valor de cada una de las pertenencias de $\mu_{e(t)}$ y $\mu_{\frac{de(t)}{dt}}$, de tal modo que $k = 1, 2, \dots, 25$ para cada combinación i, j posible. Así se completa la matriz $\mu_u(t)$ con las conclusiones de las 25 reglas.

Defusificador

Por último, la etapa de defusificación opera en los conjuntos difusos implicados producidos por la máquina de inferencias, como los presentados en la imagen, y combina sus efectos para proporcionar la salida, u^{def} , “más certera” del controlador, la cual representa las conclusiones del controlador difuso que son representadas por los conjuntos difusos implicados.

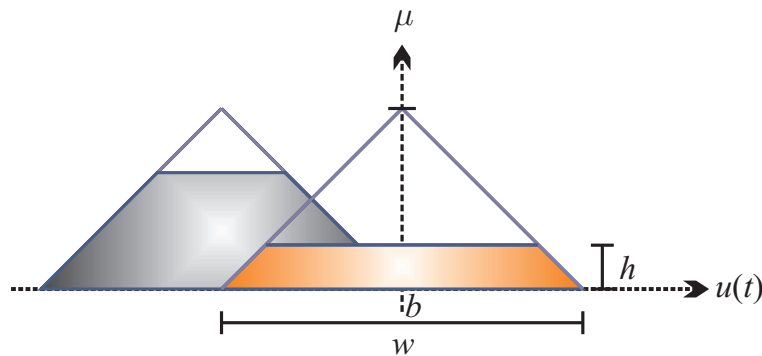


Figura 2.17: Conjuntos difusos implicados

Existen diversas aproximaciones para la defusificación, las más comunes se describen a continuación:

i) **Centro de gravedad (CDG)**: Descrito por la ecuación (2.33).

$$u^{\text{def}} = \frac{\sum_i b_i \int \mu_{\{i\}}(t)}{\sum_i \int \mu_{\{i\}}(t)} \quad (2.33)$$

Donde b_i es el centro de la función de pertenencia i , y $\int \mu_{\{i\}}(t) = w_i \left(h_i - \frac{h_i^2}{2} \right)^4$ para un ancho w_i y una altura (o inferencia) h_i de cada conjunto difuso i implicado.

ii) **Método de implicación–producto**: Descrito, también, por la ecuación (2.33) pero con $\int \mu_{\{i\}}(t) = \frac{w_i h_i}{2}$.

iii) **Promedio de centros**: Descrito por la ecuación (2.34).

$$u^{\text{def}} = \frac{\sum_i b_i \mu_{\{i\}}(t)}{\sum_i \mu_{\{i\}}(t)} \quad (2.34)$$

⁴En el caso de triángulos simétricos truncados, como los presentados en la figura 2.17

Capítulo III

Metodología

A grandes rasgos, la figura 3.1 ilustra la descripción general a bloques del proyecto en la cual se pueden observar cuatro elementos constructores principales, los cuales se describen a continuación:

- 1. Elemento sensor y actuador:** Los sensores de humedad son del tipo capacitivos de la compañía Honeywell©, modelo HIH-4000-002 que operan en un rango del 0 a 95 % de humedad relativa (o RH por las siglas en inglés de *Relative Humidity*); 2010); en tanto que los actuadores de humedad son de la marca BIONAIRE©, modelo BWM401-LA013 para el humidificador de vapor caliente y BMD100-LA013 para el deshumidificador.
- 2. Unidad DRC:** Se hizo uso de un controlador distribuido reconfigurable DRC (por las siglas de *Distributed Reconfigurable Controller*), el cual es una plataforma de hardware que se resume en la unión de una unidad de microposicionamiento MPU (acrónimo en inglés de *Micro-processing Unit*) basada en tecnología FPGA, la cual está dedicada exclusivamente a las tareas de procesamiento de datos, con un sistema para interconexión de medios SIM (siglas en inglés de, *System for Interconnection of Media*), el cual es una tarjeta que contiene todos los elementos necesarios para conectar el MPU con el mundo real como convertidores ADC (*analog-to-digital converter*) o DAC (*digital-to-analog converter*); conformando, entonces, una unidad comple-

ta de procesamiento en tiempo real con capacidad de monitoreo y control (Morales *et al.*, 2010) desarrollada por el grupo de mecatrónica de la Universidad Autónoma de Querétaro, campus San Juan del Río – Facultad de Ingeniería. En el bloque FPGA se integran los módulos del sistema de control *neurodifuso* para el procesamiento digital de las señales en hardware (o HSP por las siglas en inglés de *Hardware Signal Processing*), donde se tiene como entrada las señales de humedad adquiridas, muestreadas y convertidas por el módulo MPU y de salida una señal de control modulada por ancho de pulsos (o PWM por las siglas en inglés de *Pulse–Width Modulation*) para el control de los actuadores.

- 3. Acondicionamiento de señal:** Este bloque incluye a la etapa de amplificación de las señales analógica proveniente de los sensores de humedad para ser ajustada a los rangos de operación del DRC, y el optoacoplamiento de la señal PWM para la activación de los actuadores.
- 4. Interfaz de usuario:** La comunicación entre el usuario y la unidad DRC se realiza a través de una interfaz de comunicación basada en el protocolo estándar USB (*Universal Serial Bus*, bus universal en serie). El usuario establece el punto de referencia de humedad relativa y se obtiene la medición de ésta en el interior de la cámara, además de tener el control del inicio y fin del proceso.

La metodología para realizar el trabajo propuesto consiste en desarrollar un sistema de control donde el usuario pueda manipular el porcentaje de humedad en el interior del fitotrón. De este modo, se tiene un sistema de control en lazo cerrado como el de la figura 3.2, donde dada la referencia \mathbf{r} , se calcula el error \mathbf{e} a partir de la salida \mathbf{y} , el cual entra al controlador y éste ejerce una acción de control \mathbf{u} que intenta llevar la salida de la planta \mathbf{y} a alcanzar la referencia.

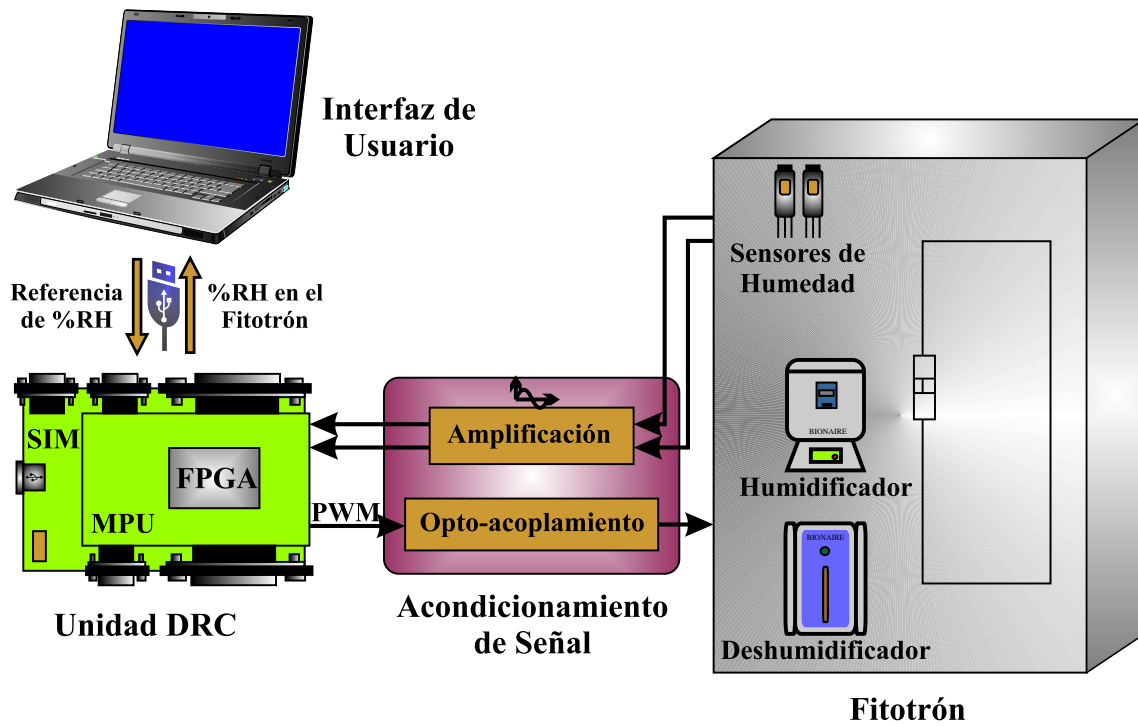


Figura 3.1: Diagrama general a bloques



Figura 3.2: Sistema de control en lazo cerrado

3.1. Dinámica del sistema

Lo primero que se hizo fue obtener la respuesta del sistema al ser estimulado. Para esto, se puso la capacidad del actuador al 100 % obteniendo el resultado presentado en la figura 3.3, donde es fácil apreciar un tiempo considerable de retardo, de aproximadamente dos minutos.

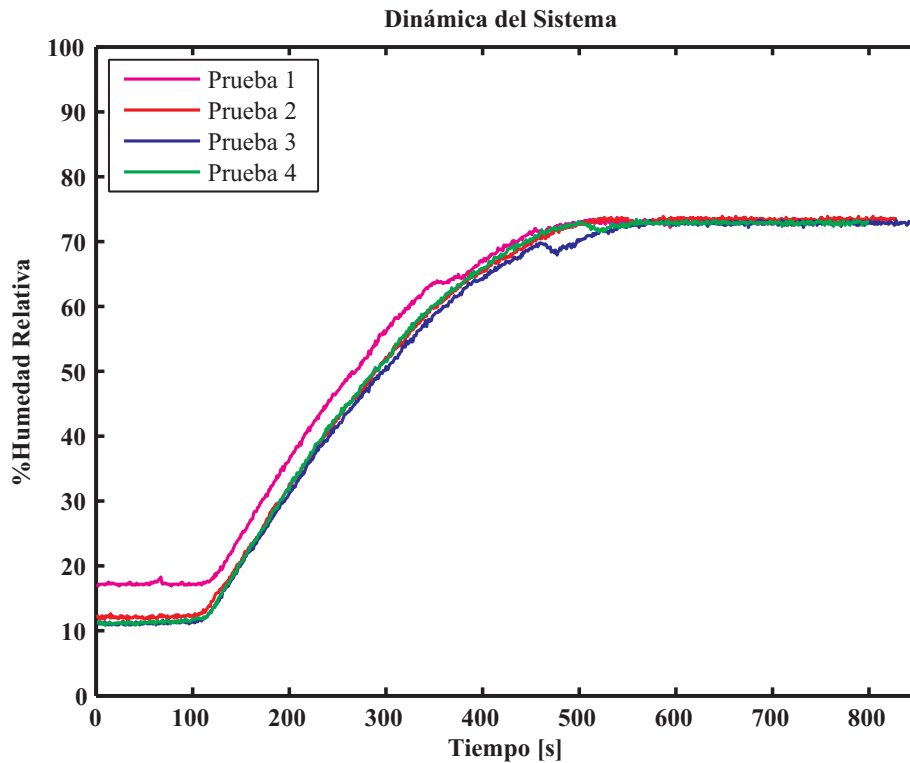


Figura 3.3: Respuesta del sistema

Obsérvese como el resultado de las 4 pruebas realizadas es similar; a partir de esto, lo siguiente es obtener un modelo matemático de la dinámica presentada por la planta a controlar.

3.1.1. Arquitectura de la red neuronal para la identificación del sistema

Las redes neuronales multicapa, aunque son de naturaleza estática, logran un comportamiento dinámico retroalimentando sus entradas con muestras de sus salidas y entradas anteriores. La identificación con redes neuronales está relacionada a la unicidad de los pesos sinápticos; es decir, dos redes con diferentes parámetros pueden producir un comportamiento idéntico entrada/salida.

Una manera de representar los sistemas lineales y no-lineales con redes neuronales, para una aproximación correcta, es incrementar las entradas de las redes neuronales con señales correspondientes a entradas y salidas con diferentes estados de tiempo. Así el sistema lineal o no-lineal queda descrito por la ecuación de diferencias (3.1), para una constante de tiempo t , donde $y, x \in \mathbb{R}$ son las salidas y entradas, respectivamente, y $n \leq m$ el tamaño de los vectores de muestras, como argumento de g que las evalúa.

$$y(t+1) = g(y(t), y(t-1), \dots, y(t-n); x(t), x(t-1), \dots, x(t-m)) \quad (3.1)$$

Se propuso, entonces, una red neuronal para obtener los parámetros de la dinámica del sistema de control de humedad en el fitotrón, cuya arquitectura consta de 5 neuronas en la capa de entrada, 1 neurona correspondiente a la capa oculta y 1 en la capa de salida, tal como se ilustra en la figura 3.4.

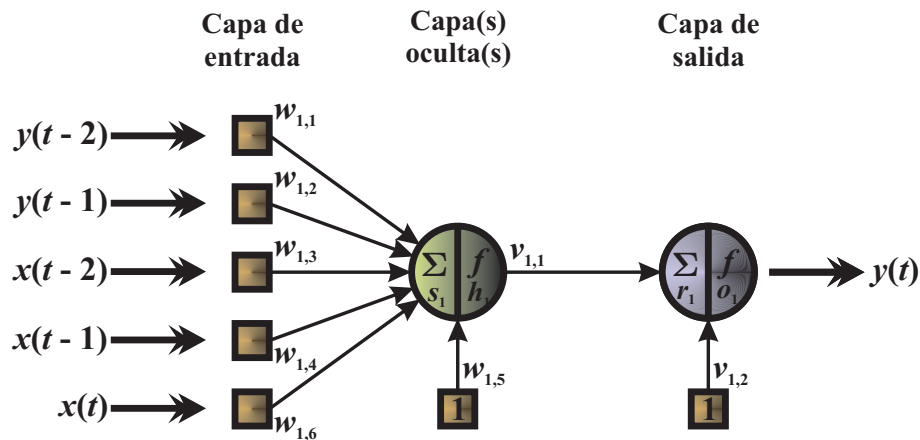


Figura 3.4: Arquitectura de la red neuronal para la identificación del sistema

3.2. Esquemas de control

3.2.1. Control clásico

En la figura 3.5 se muestra el diagrama a bloques del sistema digital del control de humedad que considera el retraso del arranque del sistema. Se observan, también, el bloque digital **Muestreo a 1s** que se encarga de obtener las muestras provenientes del convertidor ADC en la unidad SIM y entregar listas al FPGA de la unidad MPU las señales digitales provenientes de los sensores, el bloque **Controlador PID** que describe el controlador de tres tiempos, el bloque **PWM** que genera la señal de control, el bloque **Retraso** donde se lleva la cuenta del retraso del proceso y el bloque **FSM** en el cual se hace el control de las señales digitales de tal modo que el sistema pueda compensar el retraso de la dinámica de la planta. En el diseño, la identificación de la planta a controlar así como la sintonización del controlador PID se hacen de manera externa al proceso.

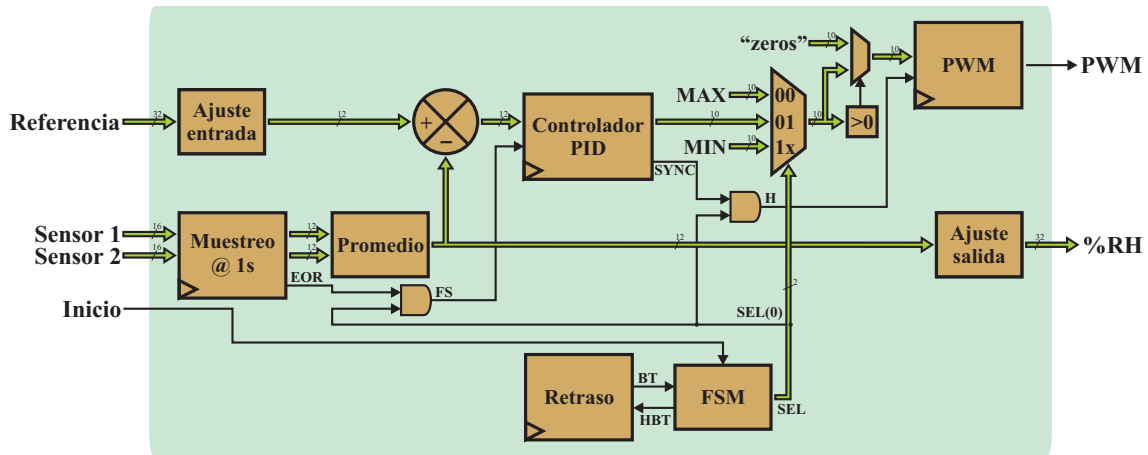


Figura 3.5: Modelo hardware del sistema de control clásico

Arquitectura del controlador PID

El diseño del controlador de tres términos se hizo de manera genérica, de manera tal que permita, además, programar las leyes de control derivadas del control PID: P (Proporcional), PI (Proporcional-Integral), PD (Proporcional-Derivativo); por lo tanto, el diseño se limitó al desarrollo de la estructura digital para implementar la ecuación (3.2), la cual representa la ecuación en tiempo discreto del controlador PID pero expresada en su ecuación en diferencias. En la tabla 3.1 se presentan las leyes de control de tres tiempos y sus derivaciones para tiempo continuo y discreto, así como los coeficientes que relacionan a las ganancias para cada ley de control con la ecuación (3.2).

$$y(k) = a_0x(k) + a_1x(k-1) + a_2x(k-2) + b_1y(k-1) + b_2y(k-2) \quad (3.2)$$

Se puede observar, también, en la ecuación (3.2) una relación directa con la ecuación de un filtro de respuesta infinita al impulso (IIR, por las siglas en inglés de *Infinite Impulse Response*). Sin embargo, los métodos de sintonización para los controladores presentados en la tabla 3.1 son los propuestos por Hernández (2005).

Tabla 3.1: Tabla de equivalencias de controladores

Controlador	F. T. Continua	F. T. Discreta	Equivalencias
P	$u(t)=K_p e(t)$	$G(z)=a_0$	$a_0=K_p$
PI	$u(t)=K_p e(t)+K_i \int_0^t e(t)dt$	$G(z)=\frac{a_0+a_1 z^{-1}}{2-2z^{-1}}$	$a_0=K_i+2K_p$ $a_1=K_i-2K_p$
PD	$u(t)=K_p e(t)+K_d \frac{de(t)}{dt}$	$G(z)=\frac{a_0+a_1 z^{-1}}{1-1z^{-1}}$	$a_0=K_p+2K_d$ $a_1=K_p-2K_d$
PID	$u(t)=K_p e(t)+K_i \int_0^t e(t)dt+K_d \frac{de(t)}{dt}$	$G(z)=\frac{a_0+a_1 z^{-1}+a_2 z^{-2}}{1-b_1 z^{-1}-b_2 z^{-2}}$	$a_0=K_p+K_i+K_d$ $a_1=-(K_p+2K_d)$ $a_2=K_d$ $b_1=1$ $b_2=0$

La figura 3.6 presenta la arquitectura hardware del controlador digital de tres tiempos descrito en la ecuación (3.2). Las operaciones del controlador digital involucran, básicamente, sumas acumuladas y multiplicaciones por lo que fue necesario inferir una unidad de multiplicación – acumulación (multiplication–accumulation, MAC), cuyo funcionamiento se resume en multiplicar dos números, sumar el resultado de la multiplicación al acumulador y almacenar el resultado de la suma en el acumulador; el limitador antes de la salida de la MAC es indispensable para evitar errores de salida y la máquina de estados (FSM, *Finite State Machine*) se encarga de controlar la MAC y el contador señala el índice del término que se está calculando. La entrada \mathbf{xk} debe desplazarse cada periodo de muestreo \mathbf{FS} al siguiente registro.

El formato de entrada \mathbf{xk} está dado por $\mathbf{ex.fx}$, el formato de la salida \mathbf{yk} por $\mathbf{ey.fy}$ y el formato de coeficientes por $\mathbf{ec.fc}$. Es importante señalar que por la forma que está descrita la unidad MAC, los cálculos y ajustes de punto decimal se realizan de manera automática.

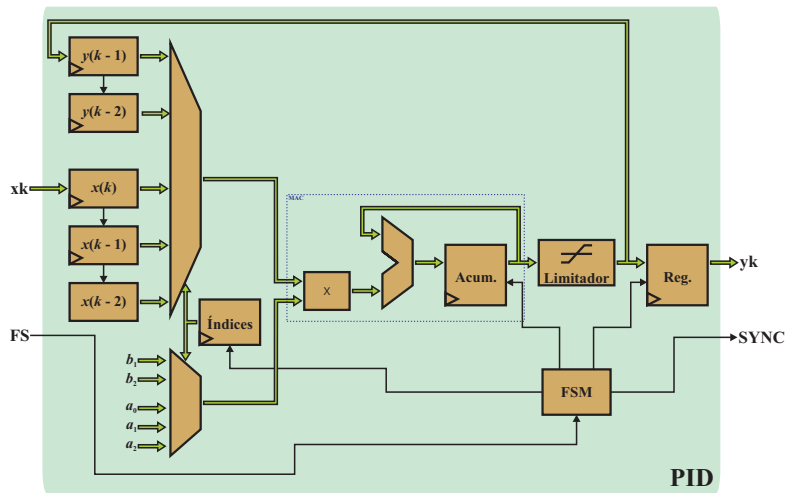


Figura 3.6: Arquitectura hardware del controlador PID

3.2.2. Control inteligente

En la figura 3.7 se muestra el diagrama a bloques del sistema digital del control inteligente de humedad. Obsérvese que a diferencia del esquema de control clásico presentado en la figura 3.5, este esquema no considera el retraso del arranque del sistema.

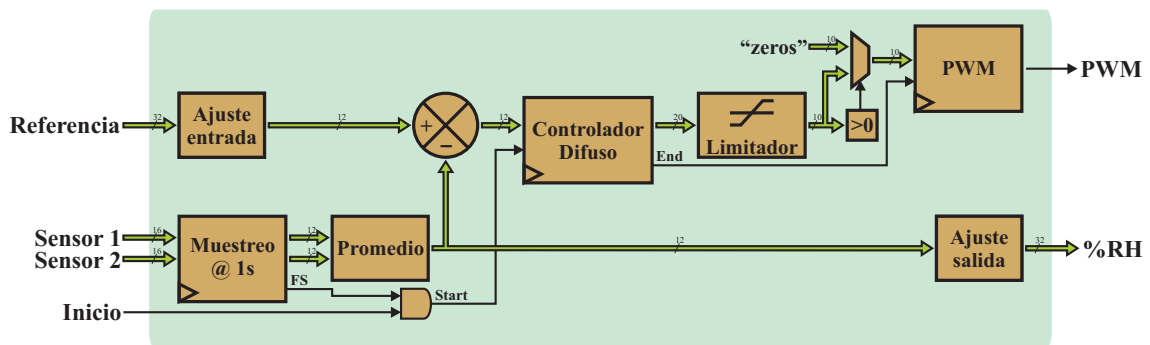


Figura 3.7: Modelo hardware del sistema de control inteligente

Arquitectura del controlador difuso

El bloque del fusificador presentado en la figura 3.9, implementado para el uso de conjuntos difusos triangulares como en la figura 3.8, se encarga de obtener la pertenencia del error y de la derivada del error y almacena estos valores mediante registros, los registros **Pe** almacenan el valor de la pertenencia del error, y el registro de salida **Pd** almacena el valor de la pertenencia para cada conjunto de la derivada del error. El proceso de fusificación se realiza mediante una evaluación directa de las ecuaciones ante el valor del error y de la derivada en los bloques aritméticos. El cálculo se realiza para cada corrida y la derivada del error es obtenida de la diferencia aritmética entre el error actual y el error anterior.

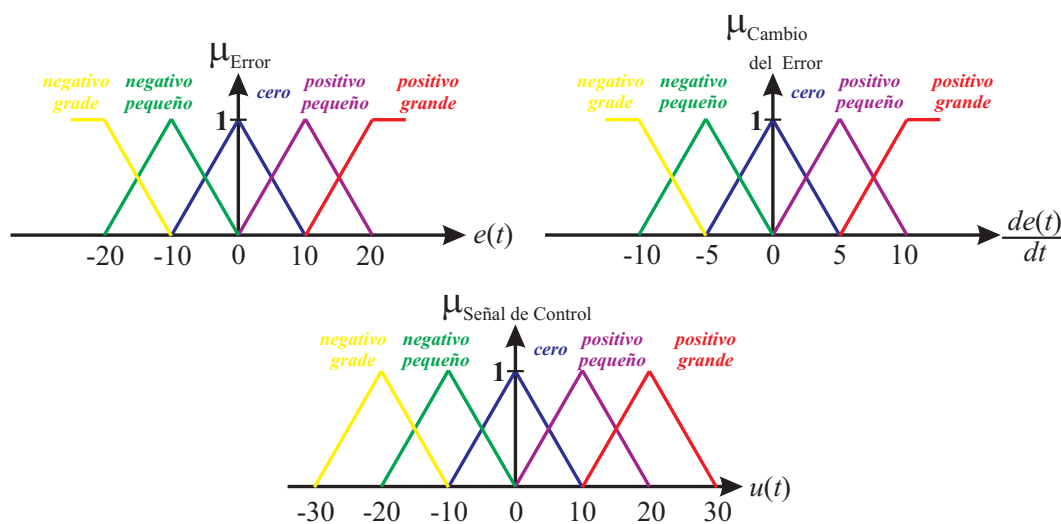


Figura 3.8: Funciones de pertenencia del controlador difuso

En síntesis, el fusificador calcula las pertenencias del error **Pe0**, **Pe2**, **Pe2**, **Pe3** y **Pe4** con el primer **Start** hasta que la señal **F5** indique que se tienen listas las 5 pertenencias. La derivada del error **Pd** se calcula una a una al siguiente **Start** controlada por la señal **Next** y **F1**, mientras que la señal **EoD** sirve para controlar si el cálculo de las inferencias (mediante el método del mínimo) se hace con el error o con la derivada del error. La base de reglas se tiene implícita en el

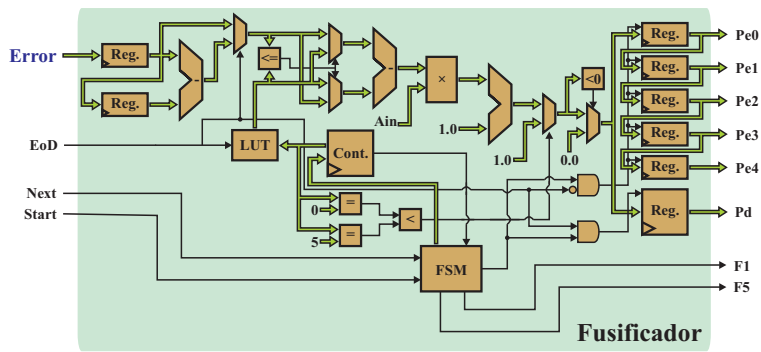


Figura 3.9: Arquitectura hardware del controlador difuso: fusificación

diseño mediante la tabla de consulta de datos (LUT) y el recorrido de la matriz se controla mediante los dos contadores, mismos que son utilizados para la etapa de defusificación mediante el método de centros de gravedad requiriendo la implementación directa de estas ecuaciones, las cual se realiza mediante las unidades MAC (multiplicación-accumulación) presentadas en el diseño, el la cual se requiere de una división, tal y como se observa en la figura 3.10

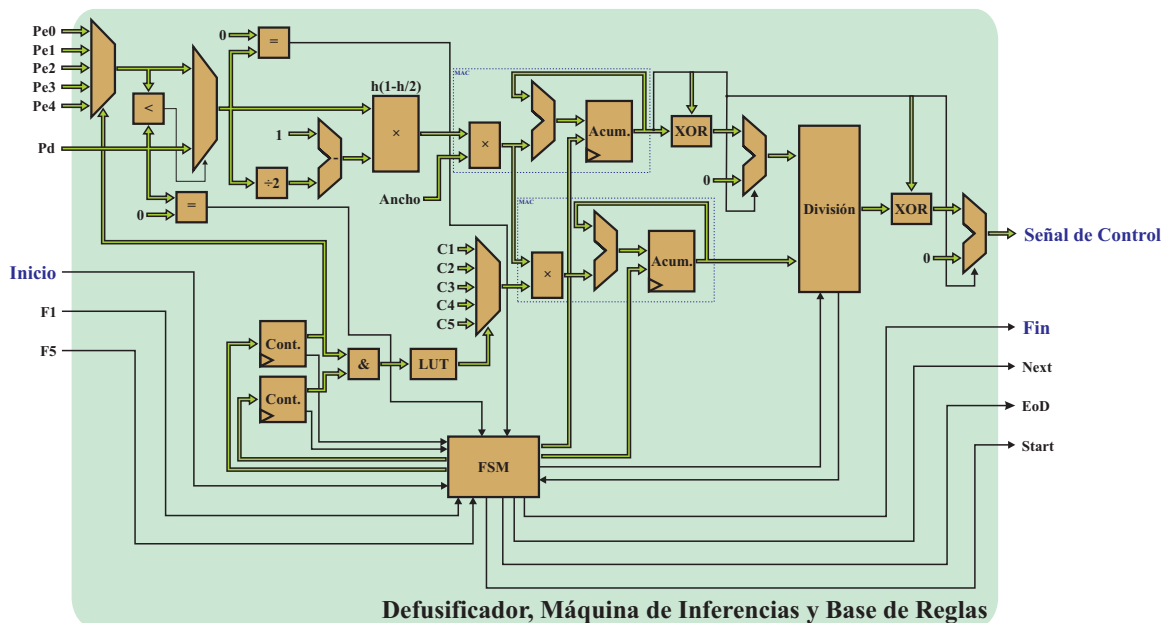


Figura 3.10: Arquitectura hardware del controlador difuso: defusificador, base de reglas y máquina de inferencias

Nótese, también, que las señales **Error**, **Señal de Control**, **Inicio** y **Fin** son las señales presentadas en el IP-*core* del controlador difuso presentado en el esquema de control inteligente en la figura 3.7.

Sintonización

Los métodos de sintonización del módulo de control difuso se efectuaron, esencialmente, de las dos formas que se describen a continuación, basándose en la información proporcionada por la red neuronal sobre la dinámica de la planta.

1. Variación en la base de reglas por lo cual es necesario modificar la tabla de consulta de datos del controlador difuso y así establecer las nuevas reglas que gobiernan el comportamiento del controlador.
2. Variación en las ganancias de los valores de entrada del error y de la derivada del error; es decir, escalar estas señales con el fin de cambiar la respuesta al controlador.

3.3. Puesta del experimento

En la figura 3.11 se presenta el fitotrón diseñado y construido por el grupo de mecatrónica de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro, campus San Juan del Río, sobre el cual se realizaron las pruebas funcionales del sistema de control de humedad con retraso intrínseco al arranque del proceso. La figura 3.12 presenta, por su parte, los elementos que integran el sistema de control de humedad en el fitotrón.



Figura 3.11: Fitotrón sobre el cual se realizaron las pruebas funcionales



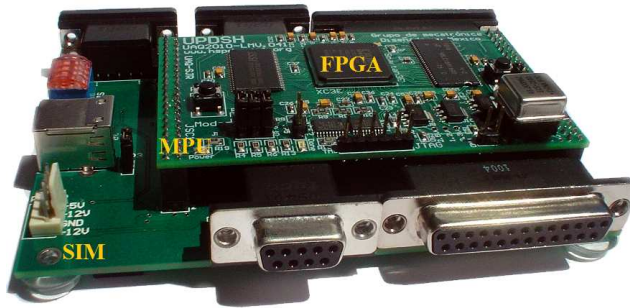
(a) Sensor de humedad



(b) Humidificador



(c) Deshumidificador



(d) Unidad DRC

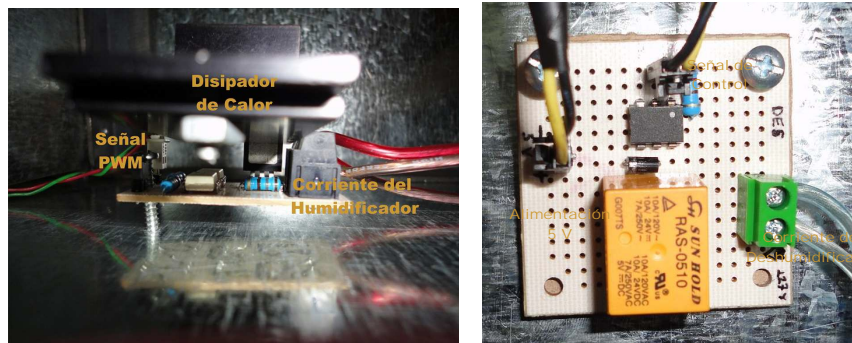
Figura 3.12: Elementos del sistema de control de humedad

3.3.1. Instrumentación del fitotrón

En la figura 3.13 (a) se muestra la ubicación de uno de los sensores de humedad dentro del fitotrón, la figura 3.13 (b) exhibe la instalación del humidificador mientras que la figura 3.13 (c) presenta la instalación del deshumidificador, y en las figuras 3.13 (d) y 3.13 (e) se presenta la tarjeta de opto-acoplamiento de la señal de control PWM para la activación del humidificador y la tarjeta de activación del deshumidificador basada en un relevador, respectivamente.



(a) Instalación de los sensores de humedad (b) Instalación del humidificador (c) Instalación del deshumidificador



(d) Tarjeta de opto-acoplamiento del humidificador (e) Tarjeta de activación del deshumidificador

Figura 3.13: Instrumentación del fitotrón

Capítulo IV

Resultados

4.1. Identificación de la dinámica sistema

Para validar la identificación por medio de la red neuronal¹ definida en la figura 3.4 que fue aplicada a la prueba 2 de la figura 3.3, el resultado de ésta, presentado en la figura 4.1, se comparó con los resultados obtenidos mediante el *System Identification Toolbox*TM de Matlab[®] expuestos en la figura 4.2, aplicados a la misma prueba.

De este modo, se obtuvo la ecuación neuronal característica (4.1) que describe la aproximación de la red neuronal en la figura 4.1. Por su parte, las ecuaciones (4.2), (4.3) y (4.4) son los parámetros de los modelos de primero, segundo y tercer orden, respectivamente, de la aproximación usando la herramienta *ident* de ©The MathWorks, Inc., presentados en la figura 4.2.

$$\begin{aligned} s_1 &= 0,6357y(t-2) + 1,2545y(t-1) + 0,0010x(t-2) + 0,0006x(t-1) - 0,1001x(t) + 0,1842 \\ h_1 &= f(s_1) \\ r_1 &= 0,5219h_1 + 0,4386 \\ y(t) &= o_1 = f(r_1) \end{aligned} \tag{4.1}$$

¹El programa para la caracterización de sistemas lineales y no-lineales usando redes neuronales artificiales es presentado en el listado 1 del apéndice A

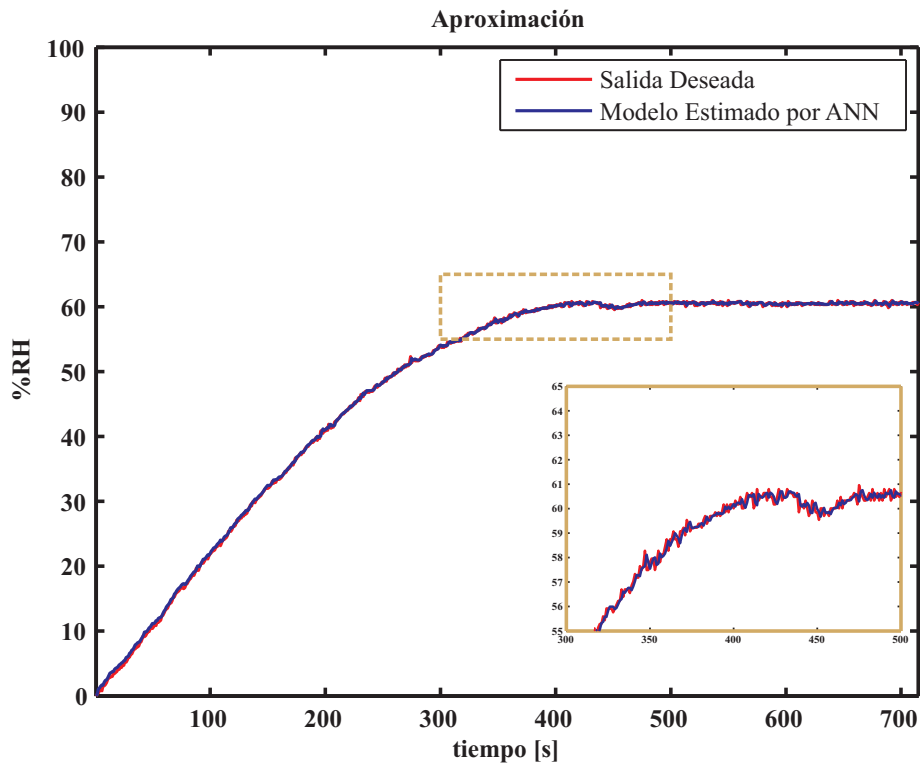


Figura 4.1: Resultado de la identificación con la red neuronal

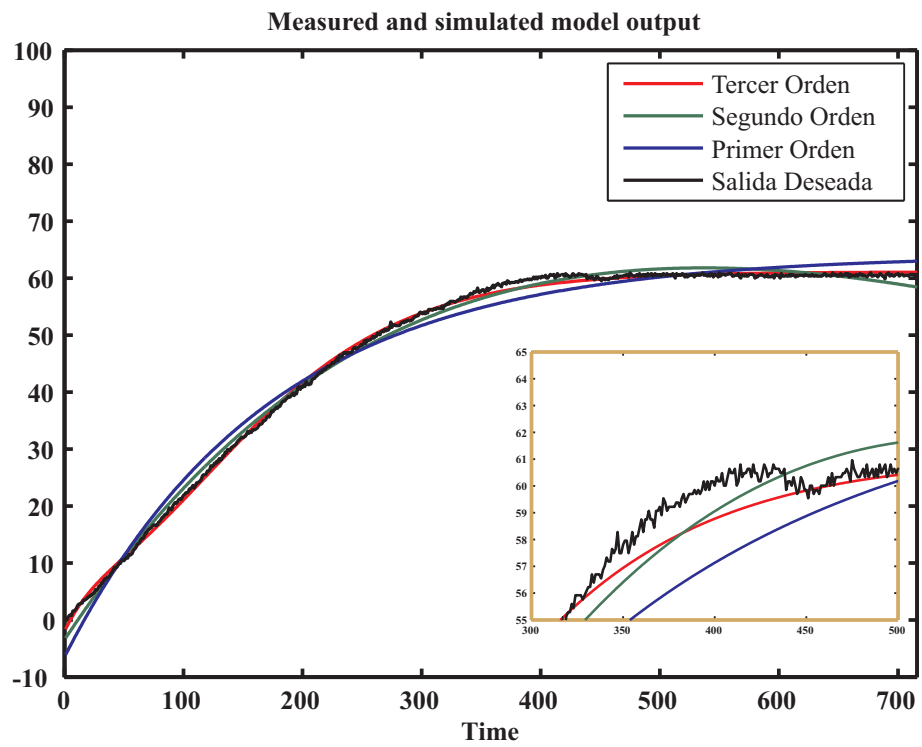


Figura 4.2: Resultado de la identificación con la herramienta ident de Matlab®

$$\frac{Y(s)}{R(s)} = \frac{0,062567}{1 + 172,4304s} \quad (4.2)$$

$$\frac{Y(s)}{R(s)} = \frac{-0,039557}{(1 + 635,77s)(1 + 635,7699s)} \quad (4.3)$$

$$\frac{Y(s)}{R(s)} = \frac{0,059647}{(1 + 60,8552s)(1 + 60,7657s)(1 + 60,8552s)} \quad (4.4)$$

Obsérvese, además, que en la comparativa de la parametrización de la dinámica del sistema de humedad en el fitotrón, el resultado obtenido por medio de la red neuronal tuvo una mejor aproximación a la salida de la planta, obteniendo así un error cuadrático medio (o RMS por las siglas en inglés de *Root Mean Square*) de $8.716224882253846 \times 10^{-06}$ durante la validación del entrenamiento basado en el algoritmo de la retro-propagación del error usando con un coeficiente de aprendizaje $\eta = 0.001$, un coeficiente de momentum $\alpha = 0.8$ y una función de activación lineal (*purelin*) en las neuronas de procesamiento de la capa oculta y la capa de salida de la red; obteniendo, de este modo, los valores de los pesos sinápticos presentados en la tabla 4.1.

Tabla 4.1: Valores de los pesos sinápticos obtenidos durante el entrenamiento de la red neuronal

$w_{i,j}$	$v_{j,k}$
0.6357	0.5219
1.2545	0.4386
0.0010	0
0.0006	0
-0.1001	0
0.1842	0

4.2. Validación de los controladores

4.2.1. Controlador PID

En la figura 4.3 se observa el comportamiento del sistema de control de humedad basado en un controlador PID ante una referencia dada. En la figura, se alcanza a observar, también, como el controlador consigue llevar el porcentaje de humedad relativa al interior del fitotrón a la referencia especificada; sin embargo, las oscilaciones que presenta en un inicio son muy considerables, incluyendo el sobrepaso inicial de 6.26 %RH por arriba de la referencia del 45 %RH, así como el tiempo de asentamiento de casi diez minutos, y un error promedio en estado estacionario de 3.2265 %RH, en el mejor de los casos.

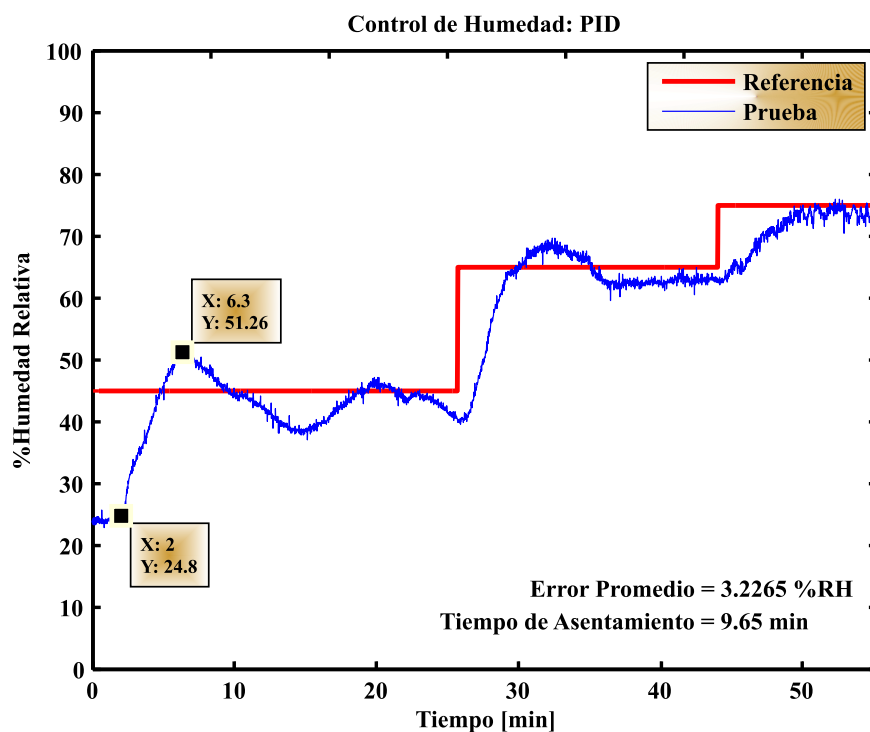


Figura 4.3: Validación del sistema de control clásico

4.2.2. Controlador difuso

Por otra lado, en la figura 4.4 se observa el comportamiento del sistema de control de humedad basado en un controlador difuso ante una referencia marcada. Se puede ver, también, como el controlador consigue llevar el porcentaje de humedad relativa al interior del fitotrón a la referencia especificada disminuyendo considerablemente las oscilaciones que presenta el sistema de control sobre la referencia, y reduciendo el sobrepaso a lo largo del proceso de tal modo que puede ser despreciado. Del mismo modo se puede ver el tiempo de asentamiento se reduce a poco mas de dos minutos, y el error promedio en estado estacionario a lo largo del proceso de control es de 1.2080 %RH.

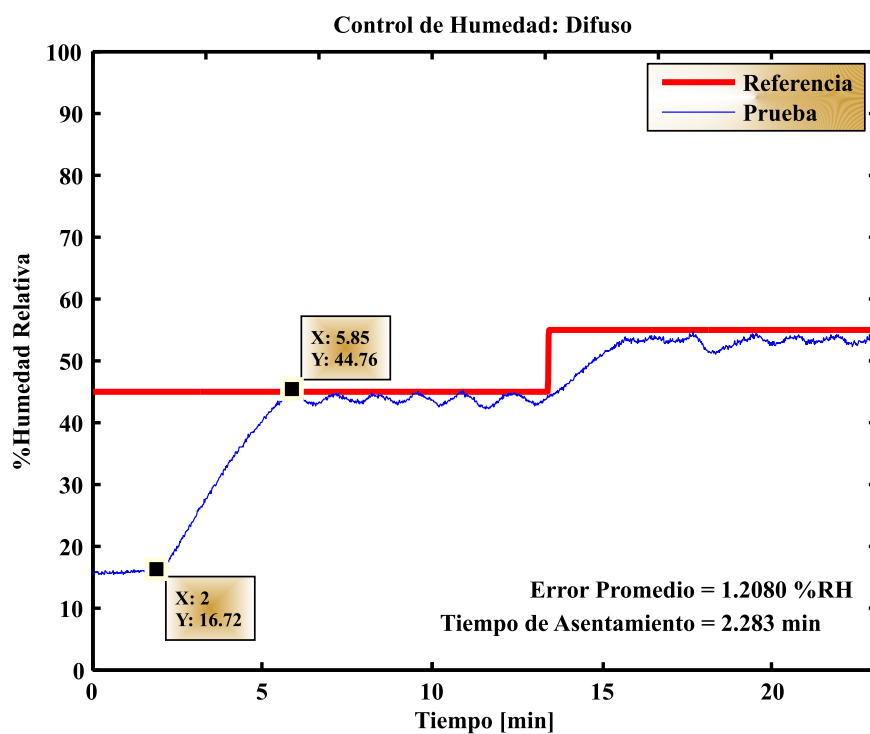


Figura 4.4: Validación del sistema de control difuso

4.3. Validación del sistema de control de humedad en el fitotrón

Por último, en la figura 4.5 se observa el comportamiento final del sistema de control de humedad en el fitotrón basado en un controlador difuso ante una referencia marcada tanto para incrementar el porcentaje de humedad relativa (humidificación) como para decrementar dicho valor (deshumidificación), observándose como el controlador consigue llevar el porcentaje de humedad relativa al interior del fitotrón a la referencia especificada, siendo notorio que la deshumidificación es más lenta; sin embargo, en términos de crecimiento de plantas, es un tiempo despreciable para el propósito de las condiciones climáticas de estudio.

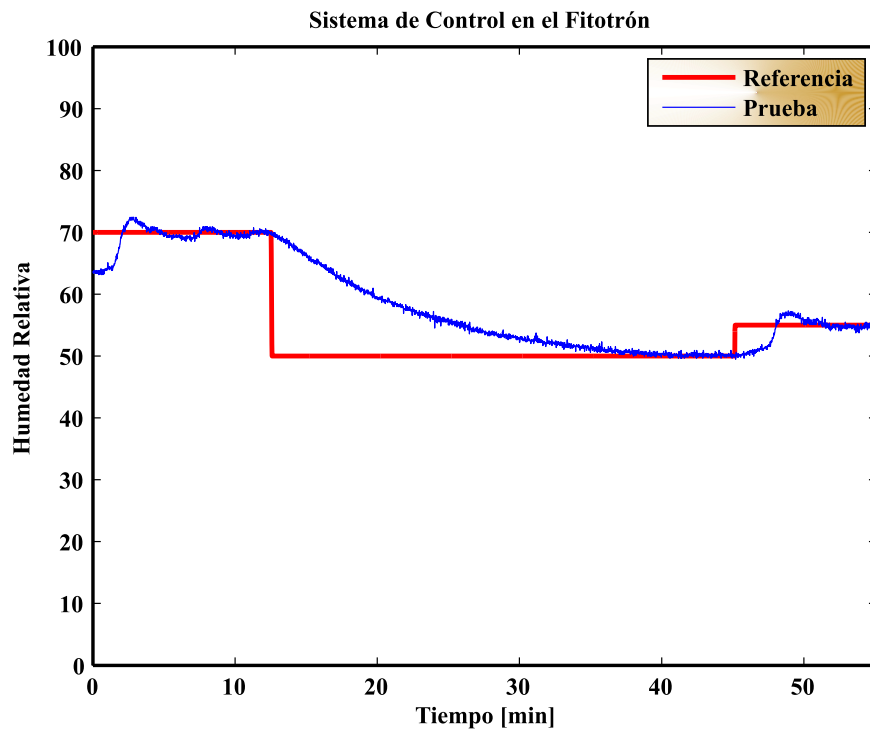


Figura 4.5: Validación del sistema de control de humedad del fitotrón

Capítulo V

Conclusiones

Se desarrolló la instrumentación necesaria para el control de variables continuas en el fitotrón, el cual fue diseñado en la Facultad de Ingeniería de la Universidad Autónoma de Querétaro, campus San Juan del Río, por el grupo de Mecatrónica, determinando el número de sensores y su ubicación dentro del mismo para un óptimo monitoreo de la humedad de manera uniforme.

Se obtuvo el modelo de la dinámica del sistema a partir de métodos lineales y usando redes neuronales artificiales, estimulando de manera externa el sistema al grado de poder parametrizar la curva del comportamiento. En este caso la parametrización con el algoritmo diseñado usando redes neuronales artificiales presentó una mejor aproximación que la herramienta de identificación de sistemas de Matlab[®].

El diseño del sistema de control a nivel hardware y el empleo de plataformas reconfigurables como FPGA's permitieron tener una arquitectura abierta para la expansión del controlador o la integración de nuevos módulos dando una solución tipo SoC, y de bajo costo, a problemas de control de variables ambientales.

El diseño de los controladores presentó el funcionamiento apropiado que se requiere en aplicaciones de bioingeniería. Sin embargo, el controlador difuso mostró un mejor desempeño en el control de la humedad del fitotrón, observando

un tiempo de asentamiento menor, una estabilidad óptima y una precisión adecuada a lo largo del proceso de control para realizar pruebas en el área de biosistemas. Además, ambos controladores cumplieron con los rangos de operación que se requiere dentro del área, proporcionando así una herramienta de investigación apta para sus necesidades al grupo de Biosistemas de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro.

Se realizó un panorama de control clásico aplicado a modelos que presentan un retraso considerable exclusivamente al inicio del proceso, de tal manera que se pudo compensar el efecto de éste sin verse reflejado en la retroalimentación del mismo.

Bibliografía

- Acidey, D. H.; Hinton, G. E. and Sejnowski, T. J. 1985. "A Learning Algorithm for Boltzmann Machine". *Cognitive Science* 9 pp. 147-169.
- Aguado, A. 2000. "Temas de Identificación y Control Adaptable". Instituto de Cibernética, Matemática y Física. ISBN: 959-7056-11-9.
- Antsaklis, P. J. and Passino, K. M. 1993. "An Introduction to Intelligencet and Autonomous Control" (Preface and Overview), Kluwer Academic Publishers, USA.
- Betzabeth, I. P. 2007. "Lógica difusa para principiantes. Teoría y práctica". Publicaciones UCAB, ISBN: 9789802445240.
- Buckley, J.; Hayashi, Y. and Czogala, E. 1992. "On the equivalence of neural networks and fuzzy expert systems". IJCNN'92, Baltimore, USA.
- Carpenter, G.; Groosberg, S.; Markuzon, N. and Reybolds, J. 1992. "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional map".
- Castañeda Miranda R., Ventura Ramos E., Peniche Vera R. del R. y Herrera Ruiz G. 2006. "Fuzzy Greenhouse Climate Control System based on a Field Programmable Gate Array". *Biosystems Engineering*. 94 (2): 165-177.
- Contreras, L. M. 2008. "Monitoreo de vibraciones para detección de fallas mecánicas". Tesis de Maestria. Universidad de Guanajuato. Junio.

- Downs, R. J. 1980. "The Botanical Review". Publisher: Springer New York; Volume 46, Number 4 / octubre; pp. 447-489. ISSN: 0006-8101.
- Enríquez González, E. 2006. "Sistema de control de temperatura aplicado a maquinaria de inyección de plástico". Tesis de Maestría. Universidad Autónoma de Querétaro – Facultad de Ingeniería.
- Freeman, J. y Skapura, D. 1991. "REDES NEURONALES. Algoritmos, aplicaciones y técnicas de programación". Addison-Wesley Iberoamericana S.A., U.S.A.
- Galán, R. 1992. La Inteligencia Artificial desde la perspectiva del Control Automático. En el libro: "La Inteligencia Artificial y el Control en Tiempo Real". Ed. Repsol Colección Ensayo.
- Galán, R.; Jiménez, A.; Sanz, R.; Matía, F. 2000. "Control Inteligente". Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial. ISSN (Versión impresa): 1137-3601. año 4, número 010. pp. 43-48.
- Gómez Meléndez, D. J. 2005. "Control *fuzzy*-PID aplicado a un sistema de riego localizado para invernadero". Tesis de Maestría. Universidad Autónoma de Querétaro – Facultad de Ingeniería.
- Haykin, S. 1998. "Neural Networks: A comprehensive Foundation". Segunda Edición, PrenticeHall.
- Hebb, D. O. 1949. "The Organization of Behavior: A neuropsychological theory". New York: John Wiley and Sons.
- Hernández Miranda D. I., 2005, "Métodos de identificación y control adaptable aplicados a una tarjeta controladora de movimiento", Tesis maestría, Universidad Autónoma de Querétaro – Facultad de Ingeniería.
- Honeywell: Golden Valley. 2010. "HIH-4000 RH Sensor Data Sheet". MN, USA.

- Hopfield, J. 1982. “Neural networks and physical systems with emergent collective computational abilities”. *Proceedings of the National Academy of Sciences of the USA*, 9(2554). 81, 3088-3092.
- Kohonen, T. 1988. “Self-organization and associative memory”. Springer Verlag.
- Jang, J. 1992. “self-learning fuzzy controllers based on temporal back propagation”. *IEEE Transactions on Neural Networks*, 3(5):714–723.
- Jang, J. and Sun, c. (1995) “Neuro-fuzzy modeling and control”. *Proceedings of the IEEE*, 83(3):378–406.
- Lang. A. 1962. “Phytotron design criteria–biological principles”. *Prod. Symp. Eng. Aspects of Environmental Control for Plant Growth*, C.S.I.R.O., Australia: 3–19.
-] Lin, C. T. and Lee, C. S. 1996. “Neural Fuzzy Systems”. Prentice–Hall PTR.
- Martín del Brío, B. 2002. “Redes Neuronales y Sistemas Difusos”. Ed. Alfaomega.
- Mas, J. F. and Flores, J. J. 2008. “The application of artificial neural networks to the analysis of remotely sensed data”. *International Journal of Remote Sensing*, 29: 3, pp. 617 – 663.
- Mayr, O. 1970. «The Origins of Feedback Control», *Scientific American*, 223, 4, October, pp. 110-118.
- McCulloch, W. S. and Pitts, W. 1943. “A logical calculus of ideas immanent in nervous activity”. *Bulletin of Mathematical Biophysics*, 5:115–133.
- Millán Almaraz J. R., Romero Troncoso R. de J., Guevara González R. G., Contreras Medina L. M., Carrillo Serrano R. V, Osornio Ríos R. A., Duarte Galván C., Ríos Alcaraz M. e I. Torres Pacheco. 2010. “FPGA-based Fused Smart Sensor for Real-Time Plant-Transpiration Dynamic Estimation”. *Sensors*. 10(9): 8316-8331.

- Morales Velázquez, L. 2007. “Unidad USB de control de posición y generación de perfiles para un intercambiador automático de herramientas”. Tesis de Maestría. Universidad Autónoma de Querétaro – Facultad de Ingeniería.
- Morales Velázquez L., Romero Troncoso R. De J., Osornio Ríos R. A., Herrera Ruiz G., y Cabal Yépez E. 2010. “Open–architecture system based on a reconfigurable hardware–software multi-agent plataform for CNC machines” *Journal of Systems Architecture*. 56(2010), pp. 407-418.
- Nanayakkara, T.; Sahin, F. and Jamshidi, M. 2010. “Intelligent Control Systems With an Introduction to System Of Systems Enegineering”. CRC Press (Taylor & Francis Group). ISBN: 978–1–4200–7924–1 (Hardback).
- Navas González, R. de J. 2000. “Diseño y Aplicación de Controladores Neuro–Difusos Integrados en Modo Mixto de Alta Complejidad”. Tesis Doctoral. Universidad de Málaga. Diciembre.
- Nie, J. and Linkens, D. 1995. “Fuzzy–Neural Control”. Prentice–Hall.
- Ogata, K. 2003. “Modern Control Engineering”. Fourth edition. Prentice Hall.
- Omidvar, O; Elliot, D. L. 1997. “Neuronal Systems For Control”; Academic Press, USA. (Ref. ITESO, 629.89 OMI)
- Osornio Ríos R. A., Romero Troncoso R. de J., Herrera Ruiz G. y Castañeda Miranda R. 2008. “The application of reconfigurable logic to high speed CNC milling machines controllers”. *Control Engineering Practice*. Vol. 16 No. 6: 674-684
- Pal, S. and Mitra, S. 1992. “Multilayer perceptron, fuzzy sets, and classification”. *IEEE Transactions on Neural Networks*, 3(5): 683–697.
- Pardo, F. y Boluda, J. 2003. “VHDL: Lenguaje para síntesis y modelado de circuitos”. Segunda edición. Ra-Ma. SBN: 9701504437

- Parker, S. 2002. "Dictionary of Scientific and Technical Terms". McGraw-Hill Professional; Sixth Edition. ISBN: 007042313X.
- Rojas Molina, A. 2003. "Control difuso para el sistema de riego de un invernadero". Tesis de Maestría. Universidad Autónoma de Querétaro – Facultad de Ingeniería.
- Romero Troncoso, R. J. 2004. "Procesamiento de señales para la detección de ruptura a herramientas en sistemas de manufactura por control numérico computarizado". Tesis Doctoral. Universidad Autónoma de Querétaro – Facultad de Ingeniería.
- Rook, D. A. 1968. "Phytotronics In Forestry". New Zealand Journal of Forestry.
- Sinnott, E. W., 1960. "Plant Morphogenesis". McGraw-Hill, London: p. 329.
- Stair, R. M. and Reynolds, G. W. 2000. "Principios de Sistemas de Información". 4ta Edición, Thompson. ISBN: 9789687529974.
- Wang, L. and Mendel, J. 1992. "Fuzzy basis functions, universal approximators and orthogonal least-squares learning". *IEEE Transactions on Neural Networks*, 3(5):807–814.
- Went, F. W. 1962. "Phytotronics". Proc. Symp. Plant Sci., Campbell Soup Co., Camden, New Jersey: 149-61.
- Zadeh, L. A. 1965. "Fuzzy Sets, Information and Control".

Apéndice A

Códigos Fuentes

Se anexa un CD con los códigos de la descripción del sistema de control *neurodifuso* de humedad en el fitotrón desarrollado en el presente trabajo, así como la implementación de la red neuronal para la parametrización de la respuesta del sistema.

También se incluye la versión en formato electrónico de éste trabajo.

Apéndice B

Productos Obtenidos

Los productos obtenidos del trabajo fueron los siguientes:

- ◇ Participación en el 7° Congreso Internacional de Ingeniería realizado del 6 al 8 de Abril de 2011 en la ciudad de Santiago de Querétaro, México, con la ponencia “Control de humedad en un fitotrón mediante un controlador PID basado en FPGA”.
- ◇ Artículos publicados:
 - i) Razo–Montes C. A., Morales–Velázquez L., Osornio–Ríos R. A., López–Cajún C. S., Torres–Pacheco I. y González–Aguirre M. A., 2011, “*Control de humedad en un fitotrón mediante un controlador PID basado en FPGA*”, Memorias del 7° Congreso Internacional de Ingeniería.
 - ii) Razo–Montes C. A., Jaen–Cuellar A. Y., Muñoz–Barrón B., Morales–Hernández L. A., Osornio–Ríos R. A., 2011, “*Método para la detección de rostros en imágenes con modelo de color RGB basado en segmentación y operaciones elementales*”, Memorias del 7° Congreso Internacional de Ingeniería.

Control de humedad en un fitotrón mediante un controlador PID basado en FPGA

Humidity control in a phytotron by a PID controller based on FPGA

Carlos Alfredo Razo-Montes¹, Luis Morales-Velázquez², Roque Alfredo Osornio-Rios², Carlos Santiago López-Cajún², Irineo Torres-Pacheco³ y Marco Antonio González-Aguirre⁴

¹ Estudiante de la Maestría en Instrumentación y Control Automático, Facultad de Ingeniería, Universidad Autónoma de Querétaro, Campus San Juan del Río, ² HSPdigital – CA Mecatrónica, Facultad de Ingeniería, Universidad Autónoma de Querétaro, Campus San Juan del Río, ³ CA Ingeniería de Biosistemas, División de Investigación y Posgrado, Facultad de Ingeniería, Universidad Autónoma de Querétaro, ⁴ Estudiante de la Maestría en Mecatrónica, Facultad de Ingeniería, Universidad Autónoma de Querétaro, Campus San Juan del Río.

RESUMEN. En este trabajo se presenta un panorama de control aplicado a modelos matemáticos que presentan un retraso intrínseco y considerable en el arranque del sistema por medio de un controlador clásico de tres tiempos PID (Proporcional-Integral-Derivativo) que procesa la información en tiempo real con un alta resolución, implementado en una plataforma hardware FPGA (*Field Programmable Gate Array*, Arreglo de Compuertas Programables en Campo). El controlador fue aplicado en un fitotrón diseñado por el grupo de Mecatrónica de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro, campus San Juan del Río, donde fue necesario compensar el retraso que presentó el actuador únicamente al arranque del proceso, demostrando, a través del control de humedad dentro del fitotrón, la funcionalidad del mismo.

Palabras clave: Humedad, Controlador PID, Retraso Inherente, Fitotrón, FPGA.

1. INTRODUCCIÓN

Desde que se intentó dar respuesta a cuestiones sobre el origen y la evolución de las especies en la tierra, con trabajos que se remontan a la época de Charles R. Darwin (1809 - 1882), las condiciones adversas de la naturaleza han sido el principal reto. Hoy día, un espacio donde se tenga la capacidad de manipular las condiciones ambientales, ejerciendo un control de éstas, ha sido la necesidad primordial de los diversos campos de estudio de la biotecnología. Actualmente, el avance tecnológico ha permitido construir y acondicionar de manera artificial lugares para el cultivo de plantas fuera de su ambiente y clima habitual, como es el caso de los invernaderos donde es posible manipular variables como temperatura, humedad, iluminación, entre otras; y pese a que en éstos se obtienen mejores resultados que de la forma tradicional, la imprecisión que los caracteriza hace imposible realizar un estudio más sofisticado. En los últimos años se han desarrollado equipos que son usados en laboratorios y centros de investigación con el fin de simular distintas posibilidades climáticas para evaluar el comportamiento del objeto a ensayar bajo dichas condiciones ambientales. Lo que se ha denominado fitotrón, o cámara de ambiente controlado, permite controlar parámetros como temperatura, humedad e intensidad luminosa, así como también en algunos casos presión, nivel de CO₂, entre otras, según las especificaciones del ensayo de una manera mucho más precisa. Sin embargo, éstas últimas presentan una serie de inconvenientes, tales como: costos muy elevados y gastos de instalación y mantenimiento exagerados por parte de los proveedores oscilando, por ejemplo, entre los \$160,813.12 (CIENTEC, 2010) y los \$978,363.50 (Biotécnica del Bajío, 2010); se genera una dependencia tecnológica con el proveedor en cuanto al uso del software de control; y, en muchos casos el sistema de control se limita a un simple controlador ON/OFF en el cual no se considera la dinámica del sistema y, para este tipo de aplicaciones, los resultados son extremadamente imprecisos; por mencionar algunos.

El uso de controladores ha tenido diversas áreas de aplicaciones, algunas de éstas se puede ver en el trabajo de Osornio *et al.* (2008) que incluye, entre otros, un módulo de un controlador PID dentro de un sistema de control para servomotores de alta velocidad basado en FPGA con aplicaciones en CNC (del inglés *Computed Numerical Control*, Control Numérico Computarizado); o en el trabajo de Moreno *et al.* (2010) donde se construyó una plataforma de microposicionamiento para dos ejes basada en FPGA, y además se realizó una comparativa entre el

controlador desarrollado y uno comercial. Asimismo, la vicisitud climática en la que se ha visto inmersa la agricultura a nivel mundial durante las últimas décadas ha originado una tendencia en la producción de cultivos que revela cada vez más el uso inminente de tecnologías de control capaces de aumentar la productividad para cubrir las demandas que la sociedad impone. Ejemplo de esto se ve en el trabajo de Castañeda *et al.* (2006) con el desarrollo de un sistema de control inteligente basado en FPGA para invernaderos aplicado en el entorno agro-industrial con una alta relación desempeño *versus* costo que genera independencia tecnológica, o el presentado por Gómez (2005) de un sistema de control que se encarga de la inyección de fertilizantes. Sin embargo, en el estudio detallado que intenta obtener modelos de crecimiento o la patología de los seres vivos, es fundamental, disponer del ambiente apropiado que se requiere para caracterizar el desarrollo de las especies, ya que éstas dependen en su totalidad de la biosfera donde se hallan inmersas y debido al espacio exagerado que abarca un invernadero es imposible monitorear las variables ambientales de una manera homogénea del mismo modo que se complica obtener el modelo del sistema de manera más precisa, por lo que éstos no alcanzan a cubrir las especificaciones técnicas que el área de la bioingeniería demanda, surgiendo la necesidad del uso de fitotrones en la investigación que se realiza en estas áreas.

La contribución del presente trabajo es el desarrollo de un sistema de control de humedad en un fitotrón, basado en tecnología FPGA de bajo costo, que considera el retraso intrínseco y cuantioso de los actuadores de humidificación en el modelo de la planta al arranque del sistema, para así obtener una aproximación real del comportamiento del sistema y una respuesta más adecuada, ya que el porcentaje de humedad relativa repercute de manera directa en la función fisiológica más importante de las plantas: la transpiración (Millán *et al.*, 2010); y que permite, además, la integración de diversos módulos de control para otras variables del fitotrón, dando una solución del tipo SoC (del inglés *System on a Chip*, Sistema en un solo Encapsulado), y proporcionar así una herramienta de investigación, acorde a sus necesidades, al grupo de Biosistemas de la Universidad Autónoma de Querétaro.

2. DESARROLLO DEL SISTEMA DE CONTROL

2.1. Dinámica del Sistema

En la naturaleza, todo sistema físico tiende a presentar retardo en su respuesta al ser estimulado de manera externa debido, por ejemplo, a la distancia física y el lugar de medición de la variable, una demora en los actuadores, o bien, por cualquier otra circunstancia. En muchos de los casos este retraso puede ser despreciado al momento de modelar el sistema y obtener una aproximación adecuada, ya que el retraso es un tiempo extremadamente corto respecto a la dinámica misma del sistema. Sin embargo, habrá sistemas donde el tiempo de retardo no se puede excluir de manera directa en el modelado del mismo, tal es el caso de la respuesta mostrada en la figura 2.1, que representó la planta a controlar en el sistema de control de humedad dentro del fitotrón, en la cual es fácil apreciar un tiempo considerable de retardo, de aproximadamente dos minutos.

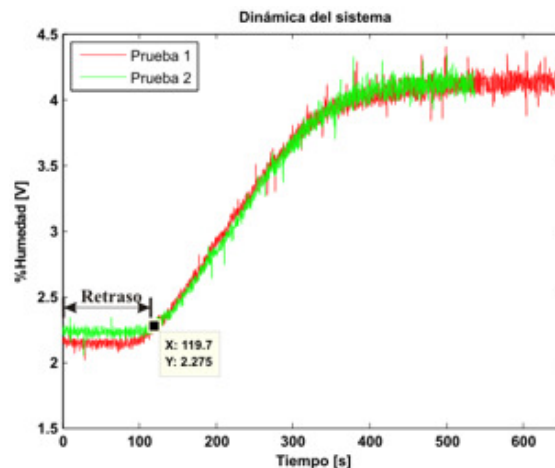


Figura 2.1: Respuesta del sistema a controlar con retraso intrínseco de 2 minutos

Existen diversos panoramas de control donde es posible compensar un retardo constante externo en la retroalimentación del proceso, el más comúnmente usado es el predictor de Smith debido principalmente a su simplicidad (Palmor, 1996). Pero en el caso de la respuesta del sistema de control de humedad, sólo se tiene un retardo al arranque del sistema, consecuencia del tiempo que le lleva al humidificador el comenzar a vaporizar; después, este retraso se reduce de manera considerable de tal manera que se puede despreciar; por lo tanto fue necesario usar otro método de compensación para el retardo presentado únicamente al arranque del proceso.

2.2. Descripción general del sistema de control

A grandes rasgos, la figura 2.2 muestra la descripción general a bloques del sistema, en la cual se pueden observar tres elementos constructores principales, los cuales se describen a continuación:

1. **Elementos sensor y actuador dentro del fitotrón:** Los sensores de humedad son del tipo capacitivos de la compañía Honeywell, modelo HIH-4000-002 que operan en un rango de 0 a 95% de humedad relativa (RH por las siglas en inglés de *Relative Humidity*) y una exactitud de 2.5% (Honeywell, 2010); en tanto que el actuador de humedad, o humidificador, es de la marca BIONAIRE modelo BWM401-LA013 de vapor tibio.
2. **Unidad DRC:** Se utilizó un controlador distribuido reconfigurable (o DRC por las siglas en inglés de *Distributed Reconfigurable Controller*), el cual es una plataforma de hardware que se resume en la unión de una unidad de microprocesamiento (MPU por las siglas en inglés de *Micro-processing Unit*) basada en tecnología FPGA, la cual está dedicada exclusivamente a las tareas de procesamiento de datos, con un sistema para interconexión de medios (o SIM por sus siglas en inglés, *System for Interconnection of Media*), el cual es una tarjeta que contiene todos los elementos necesarios para conectar el MPU con el mundo real como convertidores ADC (*analog-to-digital converter*) o DAC (*digital-to-analog converter*); conformando, entonces, una unidad completa de procesamiento en tiempo real con capacidad de monitoreo y control (Morales *et al.*, 2010) desarrollada por el grupo de Mecatrónica de la Universidad Autónoma de Querétaro, campus San Juan del Río – Facultad de Ingeniería. En el bloque FPGA se integran los módulos del sistema de control con retardo intrínseco al arranque del sistema para el procesamiento digital de las señales en hardware (o HSP por las siglas en inglés de *Hardware Signal Processing*), donde se tiene como entrada las señales de humedad adquiridas, muestreadas y convertidas por el módulo MPU y de salida una señal de control modulada por ancho de pulsos (o PWM por las siglas en inglés de *Pulse-Width Modulation*) para el control del actuador de humedad.
3. **Acondicionamiento de señal:** Este bloque incluye la etapa de amplificación de la señal analógica proveniente de los sensores de humedad para ser ajustadas a los rangos de operación del módulo DRC, y el optoacoplamiento de la señal PWM para la activación del actuador.

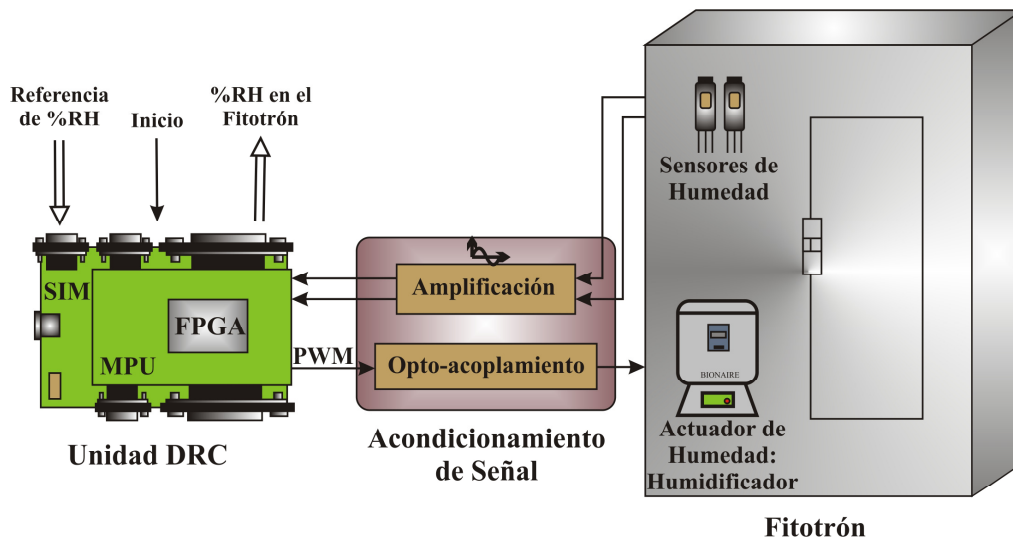


Figura 2.2: Diagrama general a bloques del sistema

2.2.1 Modelo hardware del sistema de control implementado en el FPGA

En la figura 2.3 se muestra el diagrama a bloques del sistema digital del control de humedad que considera el retraso del arranque del sistema. Se observan, también, el bloque digital **ADC** que se encarga de controlar el convertidor ADC en la unidad SIM y entregar listas al FPGA de la unidad MPU las señales digitales provenientes de los sensores, el bloque **PID** que describe el controlador de tres tiempos, el bloque **PWM** que genera la señal de control, el bloque **Retraso** donde se lleva la cuenta del retraso del proceso y el bloque **FSM** en el cual se hace el control de las señales digitales de tal modo que el sistema pueda compensar el retraso de la dinámica de la planta. En el diseño, la identificación de la planta a controlar así como la sintonización del controlador PID se hacen de manera externa al proceso.

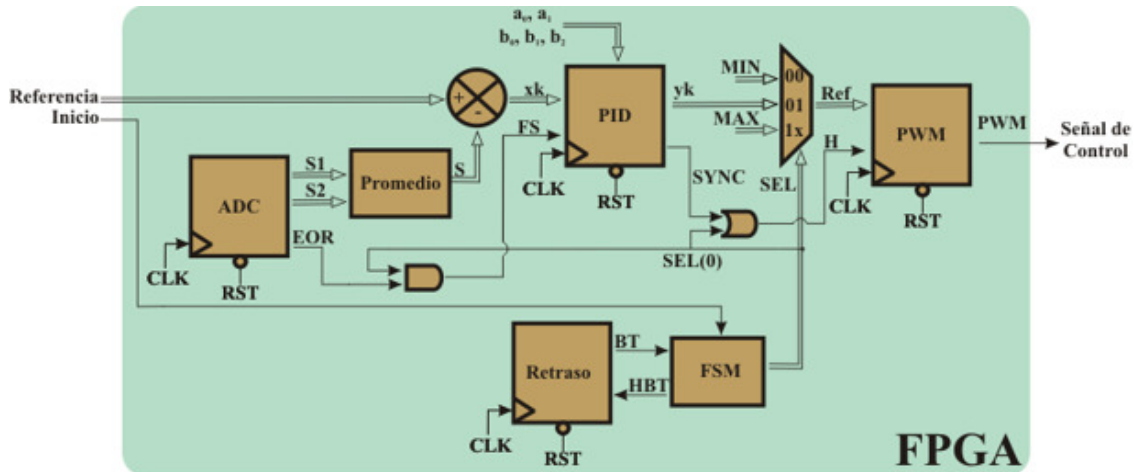


Figura 2.3: Arquitectura hardware del sistema de control

2.3. Desarrollo del Controlador

Uno de los controladores más utilizados en el control de procesos industriales, es el denominado controlador de tres términos o controlador PID (Dorf y Bishop, 2005). En sistemas digitales, un controlador digital es un caso particular de filtro digital, que se distingue por la forma en cómo se seleccionan las constantes del filtro. En teoría de control, a este proceso de ajuste de constantes se le conoce como sintonización del controlador (Ogata, 1996).

2.3.1 Implementación del controlador PID en el FPGA

El diseño del controlador de tres términos se hizo de manera genérica, de manera tal que permita, además, programar las leyes de control derivadas del control PID: P (Proporcional), PI (Proporcional-Integral), PD (Proporcional-Derivativo); por lo tanto, el diseño se limitó al desarrollo de la estructura digital para implementar la ecuación (2.1), la cual representa la ecuación en tiempo discreto del controlador PID pero expresada en su ecuación en diferencias. En la tabla 2. 1 se presentan las leyes de control de tres tiempos y sus derivaciones para tiempo continuo y discreto, así como los coeficientes que relacionan a las ganancias para cada ley de control con la ecuación (2.1).

$$y(k) = a_0x(k) + a_1x(k-1) + a_2x(k-2) + b_1y(k-1) + b_2y(k-2) \quad (2.1)$$

Se puede observar, también, en la ecuación (2.1) una relación directa con la ecuación de un filtro de respuesta infinita al impulso (IIR, por las siglas en inglés de *Infinite Impulse Response*). Sin embargo, los métodos de sintonización para los controladores presentados en la tabla 2.1 son los propuestos por Hernández (2005).

La figura 2.4 presenta la arquitectura hardware del controlador digital de tres tiempos descrito en la ecuación (2.1). Las operaciones del controlador digital involucran, básicamente, sumas acumuladas y multiplicaciones por lo que fue necesario inferir una unidad de multiplicación – acumulación (*multiplication-accumulation*, MAC), cuyo funcionamiento se resume en multiplicar dos números, sumar el resultado de la multiplicación al acumulador y almacenar el resultado de la suma en el acumulador; el limitador antes de la salida de la MAC es indispensable para evitar errores de salida y la máquina de estados (FSM, *Finite State Machine*) se encarga de controlar la MAC y el contador señala el índice del término que se está calculando. La entrada \mathbf{xk} debe desplazarse cada periodo de muestreo \mathbf{FS} al siguiente registro. El formato de entrada \mathbf{xk} está dado por $e_x . f_x$, el formato de la salida \mathbf{yk} por $e_y . f_y$ y el formato de coeficientes por $e_c . f_c$. Es importante señalar que por la forma que está descrita la unidad MAC, los cálculos y ajustes de punto decimal se realizan de manera automática.

Tabla 2.1: Tabla de equivalencias de controladores

Tipo de Controlador	F.T. Continua	F.T. Discreta	Equivalencias
P	$u(t) = K_p e(t)$	$G(z) = a_0$	$a_0 = K_p$
PI	$u(t) = K_p e(t) + K_i \int_0^t e(t) dt$	$G(z) = \frac{a_0 + a_1 z^{-1}}{2 - 2z^{-1}}$	$a_0 = K_i + 2K_p$ $a_1 = K_i - 2K_p$
PD	$u(t) = K_p e(t) + K_d \frac{de(t)}{dt}$	$G(z) = \frac{a_0 + a_1 z^{-1}}{1 - 1z^{-1}}$	$a_0 = K_p + 2K_d$ $a_1 = K_p - 2K_d$
PID	$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt}$	$G(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2}}$	$a_0 = K_p + K_i + K_d$ $a_1 = -(K_p + 2K_d)$ $a_2 = K_d$ $b_1 = 1$ $b_2 = 0$

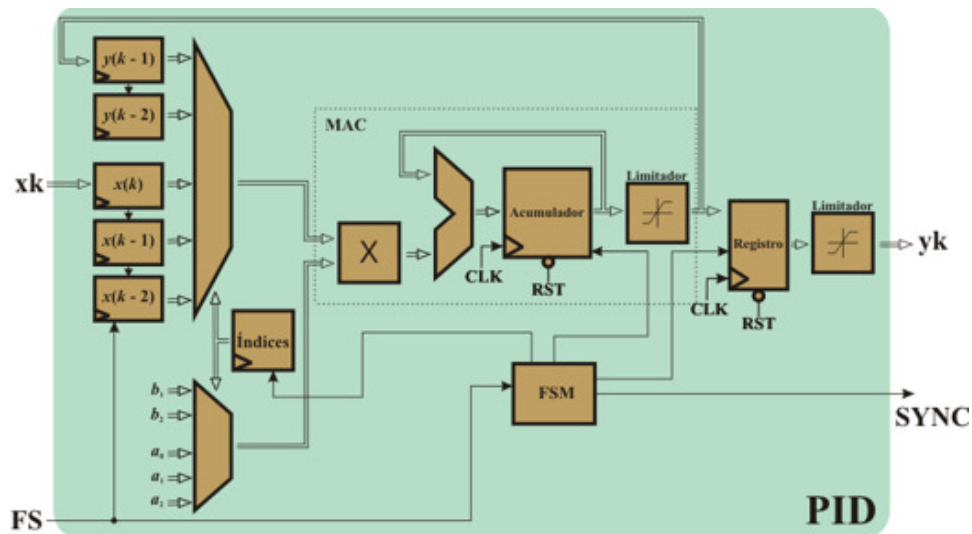


Figura 2.4: Arquitectura hardware del controlador PID

3. EXPERIMENTACIÓN Y RESULTADOS

3.1. Puesta del experimento

En la figura 3.1 se presenta el fitotrón diseñado y construido por el grupo de mecatrónica de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro, campus San Juan del Río, sobre el cual se realizaron las pruebas funcionales del sistema de control de humedad con retraso intrínseco al arranque del proceso.



Figura 3.1: Fitotrón

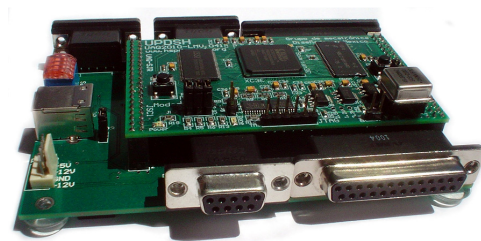
La figura 3.2 presenta los elementos descritos en la sección 2.2 que integran el sistema de control de humedad; mientras que la figura 3.3 (a) muestra la instalación del humidificador, la figura 3.3 (b) exhibe la ubicación de uno de los sensores de humedad dentro del fitotrón y, en la figura 3.3 (c) se presenta la tarjeta de opto-acoplamiento de la señal de control PWM para la activación del humidificador.



(a) Humidificador



(b) Sensor de Humedad



(c) Unidad DRC

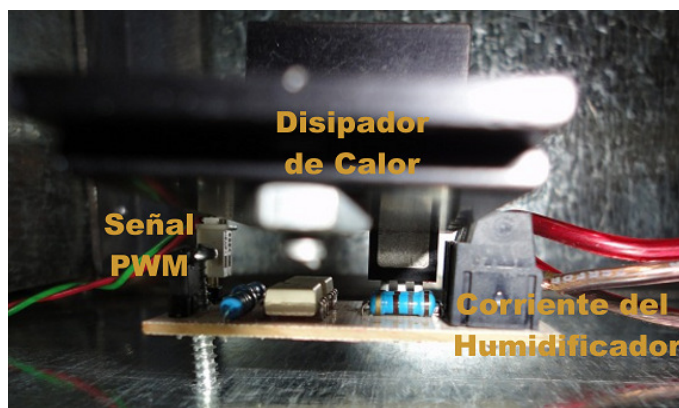
Figura 3.2: Elementos del sistema de control de humedad en el fitotrón



(a) Instalación del humidificador



(b) Instalación de los sensores de humedad



(c) Tarjeta de opto-acoplamiento del humidificador

Figura 3.3: Instalación del actuador, los sensores y la tarjeta de opto-acoplamiento de la señal de control en el fitotrón

3.2. Pruebas y Resultados

La funcionalidad del sistema de control de humedad para un fitotrón queda demostrada en la figura 3.4, donde claramente se logra observar como el controlador logra llevar a la referencia el porcentaje de humedad relativa en el interior de la cámara. En la figura 3.5 se observa, además, cómo el controlador consigue estabilizar el porcentaje de humedad relativa luego de haber sido ésta sometida a una perturbación al ventilar el interior de la cámara para extraer la humedad.

Ambas figuras, a su vez, ostentan el retraso intrínseco del proceso de 120 segundos, y un error promedio en estado estacionario de 1.1832% de humedad relativa para la prueba presentada en la figura 3.4 y 1.5627% para la prueba que se presenta en la figura 3.5, después de la perturbación.

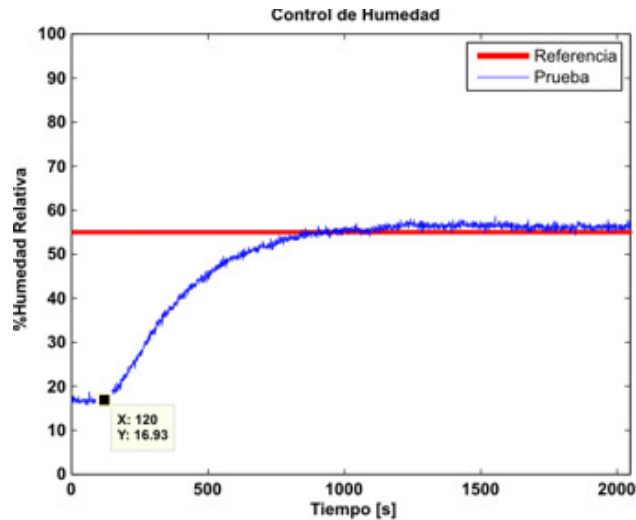


Figura 3.4: Validación del sistema de control de humedad ante una referencia del 55% de RH

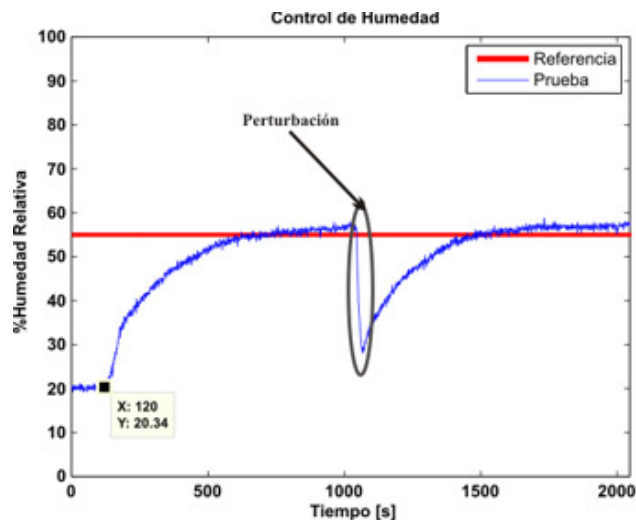


Figura 3.5: Sistema sometido a una perturbación durante el proceso de control

4. CONCLUSIONES

En el presente trabajo se realizó un panorama de control aplicado a modelos que presentan un retraso considerable exclusivamente al inicio del proceso, de tal manera que se pudo compensar el efecto de éste sin verse reflejado en la retroalimentación del mismo.

El sistema de control presentó el desempeño apropiado que se requiere para aplicaciones en el área de Biosistemas, ya que el error promedio no superó el 2% de RH. Es importante remarcar que las lecturas de los sensores de humedad presentan ciertas oscilaciones debidas, básicamente, a la continua circulación de aire en el interior de la cámara, por lo que se presentan instantes en los que llegan ventosas a los límites de los sensores con una densidad ya sea mayor o menor en porcentaje de humedad relativa; y, aunque la variación es mínima, la resolución del sensor alcanza a percibir las. Sin embargo, se puede ver cómo se logra alcanzar la referencia.

El diseño del sistema de control a nivel hardware y el empleo de plataformas reconfigurables como los FPGAs permiten tener una arquitectura abierta para la expansión del controlador o la integración de nuevos módulos y dar una solución tipo SoC de bajo costo a problemas de control de variables ambientales con relación a los equipos comerciales disponibles en el mercado.

Con el diseño, construcción, instrumentación y control por parte del grupo de Mecatrónica de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro, campus San Juan del Río, de un fitotrón que cumple con los rangos de operación, estabilidad y precisión que se requiere dentro de la bioingeniería, se le pudo proporcionar una herramienta de investigación apta para sus necesidades al grupo de Biosistemas de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro.

5. REFERENCIAS

Biotécnica del Bajío S.A. de C.V., 2010, *No. Cotización 10-0249 de la cámara bioclimática marca LUMISTELL modelo MAC-560*, se entiende el precio en pesos (16% de IVA incluido), Septiembre 29.

Castañeda Miranda R., Ventura Ramos E., Peniche Vera R. del R. y Herrera Ruiz G., 2006, *Fuzzy Greenhouse Climate Control System based on a Field Programmable Gate Array*, Biosystems Engineering, 94 (2): 165-177.

CIENTEC, S.A. de C.V., 2010, *Quotation Number EE0092 del equipo PTS Incubadora Serie ACI modelo ACI-450*, precio en pesos (16% de IVA incluido), Octubre 13.

Dorf R. C. y Bishop R. H., 2005, *Sistemas de Control Moderno*, PEARSON Prentice Hall, ISBN: 84-205-4401-9

Gómez Meléndez D. J., 2005, *Control fuzzy-PID aplicado a un sistema de riego localizado para invernadero*, Tesis de maestría, Universidad Autónoma de Querétaro – Facultad de Ingeniería.

Hernández Miranda D. I., 2005, *Métodos de identificación y control adaptable aplicados a una tarjeta controladora de movimiento*, Tesis maestría, Universidad Autónoma de Querétaro - Facultad de Ingeniería.

Honeywell: Golden Valley, 2010, *HIH-4000 RH Sensor Data Sheet*, MN, USA.

Millán Almaraz J. R., Romero Troncoso R. de J., Guevara González R. G., Contreras Medina L. M., Carrillo Serrano R. V., Osornio Ríos R. A., Duarte Galván C., Ríos Alcaraz M. e I. Torres Pacheco, 2010, *FPGA-based Fused Smart Sensor for Real-Time Plant-Transpiration Dynamic Estimation*, Sensors, 10(9): 8316-8331.

Morales Velázquez L., Romero Troncoso R. de J., Osornio Ríos R. A., Herrera Ruiz G., y Cabal Yépez E., 2010, *Open-architecture system based on a reconfigurable hardware-software multi-agent platform for CNC machines*, Journal of Systems Architecture, 56(2010): 407-418.

Moreno Tapia S. V., Romero Troncoso R. de J. y Osornio Ríos R. A., 2010, *Sistema de control para microposicionamiento basado en FPGA*, Memorias 6° Congreso Internacional de Ingeniería, Universidad Autónoma de Querétaro: 44-52.

Ogata K, 1996, *Modern control engineering*, 3a ed, Prentice Hall, ISBN: 13-227307-1

Osornio Ríos R. A., Romero Troncoso R. de J., Herrera Ruiz G. y Castañeda Miranda R., 2008, *The application of reconfigurable logic to high speed CNC milling machines controllers*, Control Engineering Practice, Vol. 16 No. 6: 674-684

Palmor Z. J., 1996, *The Control Handbook. Time Delay Compensation: Smith Predictor and its Modifications*. CRC Press and IEEE Press, Boca Raton.