



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
FACULTAD DE INGENIERÍA

Identificación de Sistemas No Lineales
Mediante Redes Neurodifusas Recurrentes

T E S I S

QUE PARA OBTENER EL GRADO DE:

Doctor en Ingeniería

PRESENTA:

Marcos Angel González Olvera

ASESOR:

Dr. Yu Tang Xu



2009



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Presidente: Dr. Luis Agustín Álvarez-Icaza Longoria

Secretario: Dr. Gerardo René Espinosa Pérez

Vocal: Dr. Yu Tang Xu

1er Suplente: Dr. Wen Yu Liu

2o Suplente: Dr. Héctor Benítez Pérez

Fecha del examen de grado:

“Lo controlable nunca es totalmente real, y lo real nunca es totalmente controlable.” Vladimir Nabokov, escritor.

A:

Mi familia, por su paciencia y apoyo incondicional,
Verónica, por su gran paciencia y nuestros planes presentes y futuros,
La Universidad Nacional Autónoma de México, por haberme dado tanto a
lo largo de mis años de estudio,
Dr. Yu Tang, por haber sido más que mi asesor, un tutor y un amigo en
todo el sentido de la palabra, por 7 años,
Dr. Gerardo Espinosa, por fungir como un tutor *honorario* por los mismos
7 años,
Al Dr. Luis Álvarez-Icaza, Dr. Héctor Benítez, Dr. Leonid Fridman, Dr.
Marco Arteaga, por sus enseñanzas más allá del nivel académico,
A todos los miembros del jurado, por su asesoría y orientación en la
realización de este trabajo.
A CONACyT, CEP-UNAM (antes DGEP) y PAPIIT-UNAM por su
apoyo.

Agradecimientos

Extiendo mi reconocimiento a:

1. **Coordinación de Estudios de Posgrado (CEP, antes Dirección General de Estudios de Posgrado)**, por el apoyo otorgado mediante la beca de doctorado de Agosto 2007 a Enero 2008
2. **Consejo Nacional de Ciencia y Tecnología (CONACyT)**, por el apoyo otorgado mediante la beca de estudios para estudios de posgrado, de Febrero 2008 a Abril 2009.
3. **Proyecto PAPIIT-UNAM**, proyectos IN-106206, IN-120009 y IN-112908.

Índice general

Índice de figuras	VII
Índice de cuadros	XI
1. Introducción	1
1.1. Formulación del problema	1
1.1.1. Identificación de sistemas	4
1.2. Estado del arte	5
1.2.1. Redes neurodifusas	5
1.2.2. Redes neurodifusas recurrentes	6
1.3. Desarrollo del tema de investigación	8
1.3.1. Objetivo de la investigación	8
1.3.2. Principales contribuciones	9
1.3.3. Organización del trabajo de tesis	10
2. Preliminares	13
2.1. Redes neuronales	13
2.2. Sistemas difusos	14
2.3. Redes neurodifusas	18
2.3.1. Aproximadores universales	18
2.4. Redes recurrentes	19
2.5. Métodos existentes de entrenamiento	22
2.5.1. Mínimos cuadrados	22
2.5.2. Filtro de Kalman	24
2.5.3. Algoritmos basados en el gradiente	25
2.6. Algoritmos de agrupamiento	26

ÍNDICE GENERAL

2.6.1. Algoritmo de k-medias (<i>K-Means</i>)	26
2.6.2. Algoritmo difuso de c-medias (<i>Fuzzy C-Means</i>)	27
2.7. Observadores adaptables	28
3. Identificación mediante redes recurrentes entrada-salida	31
3.1. Estructura en tiempo discreto Entrada/Salida	32
3.1.1. Estructura entrada-salida	33
3.1.1.1. Linealización de los parámetros antecedentes	35
3.1.1.2. Algoritmo de inicialización de parámetros	37
3.1.1.3. Análisis de la estabilidad de la red	38
3.1.2. Algoritmos de entrenamiento	38
3.1.2.1. Entrenamiento por mínimos cuadrados	38
3.1.2.2. Entrenamiento por Filtro de Kalman	39
3.1.2.3. Entrenamiento basado en estabilidad en el sentido de Lyapunov	40
3.1.2.4. Ejemplos	42
3.1.3. Estructura Entrada-Salida con restricciones en los parámetros . .	46
3.2. Control basado en predicción un-paso-adelante	51
3.3. Ejemplos	53
3.4. Comentarios generales	54
4. Identificación mediante redes recurrentes en espacio de estados	57
4.1. Estructura de la red: Tiempo continuo	59
4.2. Algoritmo de entrenamiento	60
4.2.1. Casos particulares de regresores	64
4.2.1.1. Regresor como función de y y u	67
4.3. Ejemplos	68
4.3.1. Identificación de una red neuronal recurrente	68
4.3.2. Identificación de un sistema no lineal	70
4.3.2.1. Identificación con partición del espacio en y y u	71
4.3.2.2. Identificación con partición del espacio en y	72
4.3.3. Identificación de un sistema en forma de realimentación de salida	73
4.4. Control basado en la red identificada: Caso particular	76
4.4.1. Estructura neurodifusa para control.	79

4.4.2. Diseño de un control de seguimiento por realimentación de salida basado en el modelo neurodifuso	80
4.4.2.1. Diseño de control con σ conocida.	80
4.4.3. Diseño de control con σ desconocida y \mathbf{b} conocida.	82
4.4.4. Ejemplo	82
4.5. Estructura de la red: Tiempo discreto	84
4.6. Comentarios	87
5. Aplicaciones	89
5.1. Motor de corriente directa	89
5.2. Robot flexible de un eslabón	91
5.3. Celda de tráfico	93
5.3.1. Descripción del problema	93
5.3.2. Desarrollo	96
5.4. Robot planar de 2GDL para servo-visual	102
5.4.1. Descripción del problema	102
5.4.2. Desarrollo	103
5.4.2.1. Resultados de simulación	104
5.4.2.2. Resultados experimentales	105
5.5. Sistema glucosa-insulina en pacientes diabéticos	105
5.5.1. Desarrollo	109
5.5.2. Resultados de simulación	110
6. Conclusiones	117
6.1. Conclusiones	117
6.2. Líneas de investigación abiertas y trabajo futuro	119
6.3. Publicaciones generadas	120
A. Teoremas empleados	123
A.1. Transformación a realimentación de salida no lineal	123
A.2. Control por realimentación de salida dinámica	124
Referencias	127

ÍNDICE GENERAL

Índice de figuras

1.1. Identificación de un sistema mediante una red	9
2.1. Neurona y Red Neuronal	14
2.2. Ejemplos de funciones de membresía	16
2.3. Agrupamiento de datos	26
3.1. Identificación de un sistema lineal	43
3.2. Identificación de un sistema lineal: Evolución de los parámetros	44
3.3. Identificación de un sistema no lineal	45
3.4. Identificación de un sistema no lineal: Evolución de los parámetros	46
3.5. Identificación de un sistema no lineal: Conjuntos difusos resultantes tras el entrenamiento	47
3.6. Restricción en los parámetros	48
3.7. Identificación de un sistema no lineal: Red con restricciones en parámetros.	53
3.8. Identificación de un sistema no lineal: Evolución de los parámetros en una red con restricciones en parámetros	54
3.9. Identificación de un sistema no lineal: Funciones de membresía tras entrenamiento	54
3.10. Control de un sistema no lineal: Red con restricciones en parámetros.	55
4.1. Identificación ideal: Funciones de membresía	69
4.2. Identificación ideal: Evolución de los parámetros durante el entrenamiento	70
4.3. Identificación ideal: Comparación de los estados de ambos sistemas ante una misma entrada	71
4.4. Identificación ideal: Comparación de la salida de ambos sistemas ante una misma entrada	72

ÍNDICE DE FIGURAS

4.5. Identificación de un sistema no lineal: Funciones de membresía	73
4.6. Identificación de un sistema no lineal: Comparación de la evolución de los parámetros.	73
4.7. Identificación de un sistema no lineal: Comparación de la evolución de los estados.	74
4.8. Identificación de un sistema no lineal: Comparación de las señales de salida.	74
4.9. Sistema no lineal en realimentación de salida: Evolución de los parámetros y estados	75
4.10. Sistema no lineal en realimentación de salida: Comparación de la salida	76
4.11. Sistema no lineal en realimentación de salida con $\sigma_i = 0,5$	76
4.12. Esquema de control mediante la red identificada	77
4.13. Sistema no lineal: Evolución de parámetros en el entrenamiento	83
4.14. Sistema no lineal: Resultados de la identificación.	84
4.15. Sistema no lineal: Control	85
5.1. Motor CD: Sistema físico	90
5.2. Motor CD: Señales de excitación	91
5.3. Motor CD: Evolución de los parámetros	92
5.4. Motor CD: Desempeño de la red en paralelo con el sistema	93
5.5. Robot Flexible: Esquema	93
5.6. Robot flexible: Evolución de los parámetros	94
5.7. Robot flexible: Señales de salida	95
5.8. Celda de Tráfico: Relación densidad-flujo vehicular en una autopista . .	96
5.9. Celda de Tráfico: División de una autopista por celdas	96
5.10. Celda de Tráfico: Localización de la autopista i210-West.	97
5.11. Celda de Tráfico: Sección de la autopista i210-West	97
5.12. Celda de Tráfico: Evolución de los parámetros durante el entrenamiento	101
5.13. Sistema de servo-visual: Sistema de servo-visual y robot en el plano de imagen	103
5.14. Sistema de servo visión: Señales de entrada	106
5.15. Sistema de servo visión: Trayectorias del robot	107
5.16. Sistema de servo visión: Funciones de membresía para $y(t) = \phi_1$ y $y(t) = \phi_2$	108

5.17. Sistema de servo visión: Evolución de parámetros y error RMS	108
5.18. Sistema de servo visión: Resultados de modelado para $y(t) = \phi_1$	109
5.19. Sistema de servo visión: Resultados de modelado con datos experimentales	110
5.20. Dinámica de la insulina: Identificación con una entrada (insulina)	112
5.21. Dinámica de la insulina: Identificación con dos entradas (glucosa e in- sulina) y $T_s = 0,2[min]$	113
5.22. Dinámica de la insulina: Identificación con dos entradas (glucosa e in- sulina) y $T_s = 5[min]$	114
5.23. Dinámica de la insulina: Conjuntos difusos obtenidos con c-medias difuso para dos entradas (insulina y glucosa)	115

ÍNDICE DE FIGURAS

Índice de cuadros

5.1. Celda de Tráfico: Matrices diseñadas para cada sección y parámetros consecuentes	98
5.2. Celda de Tráfico: Valores máximos de los datos	99
5.3. Celda de tráfico: Funciones de membresía por sección	100

ÍNDICE DE CUADROS

Capítulo 1

Introducción

1.1. Formulación del problema

En ingeniería se emplea y define el concepto de *sistema* como aquel ente compuesto por varios elementos, donde variables de diversa naturaleza interactúan entre sí para un objetivo común, responden a señales externas (o *señales de entrada*) y generan señales observables, también llamadas *señales de salida*. Éstos se suelen clasificar en *dinámicos* y *estáticos*. Los primeros son aquellos en los que la señal de salida actual depende de valores anteriores de la entrada, mientras que en los segundos no existe dicha dependencia. Una gran cantidad de sistemas físicos estudiados en ingeniería se representan mediante sistemas dinámicos, por lo que el interés del presente trabajo se centrará en ellos.

Para establecer la relación que las variables guardan tanto entre sí como con las señales de entrada y salida se emplean funciones matemáticas, con lo que se establece un *modelo matemático* del sistema, los cuales permiten explicar el funcionamiento de un sistema (análisis), proponer medios que permitan controlarlo (control), así como predecir su desempeño (predicción) y permitir simularlo mediante paquetes de cómputo.

La obtención del modelo depende del problema en específico a resolver. En primer lugar, se suele dividir al sistema en varios sub-sistemas y analizar las propiedades de cada uno ya sea mediante leyes físicas u otras fuentes de conocimiento, para luego sintetizarlos en un modelo final. También se obtienen mediante experimentación directa, donde los datos son analizados y se infiere un modelo. En el primer caso, hablamos del *modelado* de un sistema, mientras que en segundo se habla de la *identificación* del

sistema. Generalmente, para atacar un problema en particular, se recurre a ambos métodos para obtener el modelo final de interés.

Es de tener en cuenta que un modelo es una simplificación o aproximación del sistema físico analizado, ya que el conocimiento matemático y dicho sistema corresponden a espacios diferentes. En la medida en la que se desee una mayor exactitud, el modelo tenderá a tener un mayor nivel de complejidad, por lo que se debe ajustar dicho nivel al que mejor se adapte a las necesidades del problema a resolver. Sin embargo, como lo indica Ljung (1986), por cuestiones prácticas se suele emplear el concepto de *sistema real* al modelo matemático que describe al sistema en los términos anteriormente discutidos.

En lo que respecta a la teoría de control y al tratarse de sistemas en tiempo continuo, el tipo de modelos a emplear suelen expresarse mediante una representación en variables de estado de la forma

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), \quad \mathbf{x} \in \mathbb{R}^n, \quad u \in \mathbb{R}^m, \quad f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n, \quad (1.1)$$

$$y = h(\mathbf{x}), \quad y \in \mathbb{R}^p, \quad h : \mathbb{R}^n \rightarrow \mathbb{R}^p \quad (1.2)$$

Dentro de esta estructura, las variables de estado \mathbf{x} guardan la información de la dinámica del sistema y suelen estar relacionadas con variables que tienen un significado físico (generalmente con aquellos elementos que almacenan energía dentro del sistema). Un buen conocimiento de la función f permite analizar y realizar diseños sobre el sistema físico al cual representa, así como describir el efecto que la entrada u tendrá sobre la salida observada y . De igual modo, los sistemas en tiempo discreto pueden ser expresados de un modo más compacto mediante ecuaciones en diferencias

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k), \quad \mathbf{x} \in \mathbb{R}^n, \quad u \in \mathbb{R}^m, \quad f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n, \quad (1.3)$$

$$y_k = h(\mathbf{x}_k), \quad y \in \mathbb{R}^p, \quad h : \mathbb{R}^n \rightarrow \mathbb{R}^p \quad (1.4)$$

Bajo determinadas condiciones, el sistema es representable mediante una estructura entrada/salida en la forma

$$y_{k+1} = f(y_k, \dots, y_{k-n+1}, u_k, \dots, u_{k-m+1}). \quad (1.5)$$

De forma general, la dinámica del sistema se puede representar mediante un mapeo dinámico entrada-salida

$$y(t) = H(u(t)). \quad (1.6)$$

Sin embargo, es posible que H sea desconocida en forma que genere un modelo adecuado para esta respuesta, y sea necesario recurrir a técnicas de identificación para reconstruirla a partir de mediciones, experimentos y observaciones del sistema mediante el mapeo

$$\hat{y} = \hat{H}(u(t)). \quad (1.7)$$

En la literatura (Nelles, 2001) se suele hacer una distinción con respecto a la cantidad de información disponible para obtener el modelo. Por una parte, se llama *caja negra* a aquel sistema del cual solo se disponen datos de las señales de entrada y salida, sin conocimiento sobre las acciones que se suceden dentro del mismo. Por otro lado, cuando existe un conocimiento parcial de éstas, se suele llamar al sistema *caja gris*, con lo cual una parte del modelo puede estar formado por relaciones matemáticas obtenidas de forma analítica, mientras que otra se puede deducir a partir del análisis de los datos y aproximar mediante expresiones matemáticas, como aproximaciones polinomiales, por serie de Taylor, etc.

Las redes neuronales, sistemas difusos y su combinación conocida como *redes neurodifusas* han mostrado ser herramientas muy útiles para describir funciones matemáticas con diversos grados de complejidad, así como por su capacidad para modificar tanto su estructura como los parámetros involucrados en su definición. Las primeras son representaciones matemáticas de los procesos neuronales que ocurren en los sistemas biológicos, mientras que los segundos se basan en la construcción de modelos mentales seguidos por los seres humanos. Al haber sido exitosamente empleadas en la descripción de sistemas estáticos a partir de datos, han sido también empleadas en la identificación de sistemas dinámicos.

Dentro de este trabajo se ataca el problema de identificación de sistemas dinámicos a partir de los tres tipos de modelado descritos anteriormente. Se considerará para tal efecto un sistema del que se desea obtener su modelo matemático, en el que se cuenta con datos de las señales de entrada y salida con poca información sobre su modelo matemático *real*. De este modo, se propone una estrategia de identificación de sistemas basada en redes neurodifusas, las cuales han sido empleadas con éxito en la identificación de funciones desconocidas a partir de la información entrada-salida disponible.

1.1.1. Identificación de sistemas

En general, el problema de modelado e identificación mediante redes recurrentes se deriva del problema más general de encontrar un mapeo de un sistema del cual no se dispone de información sobre el mismo más que datos entrada/salida.

Existen dos estructuras básicas para el modelado de sistemas: por un lado las estructuras entrada/salida, que buscan representar el comportamiento dinámico de ambas señales con base en un único mapeo dinámico; por el otro, estructuras en variables de estado, donde un conjunto de variables, solución de un conjunto de ecuaciones diferenciales de primer orden simultáneas, conserva la dinámica del sistema, siendo la salida del sistema una función algebraica de dichas variables y la señal (o señales) de entrada.

En el primer caso, los sistemas lineales en el tiempo se modelan mediante la *respuesta al impulso* $g(t)$, donde la señal de salida puede ser obtenida mediante el mapeo $y(t) = H(u(t))$, con $H(u(t)) = \int_0^t g(t-\tau)u(\tau)d\tau$. Es bien sabido que al transformar al dominio de la frecuencia se obtiene una relación ya no integro-diferencial, sino algebraica, llamada *función de transferencia*. En este caso, con $U(s)$, $Y(s)$ y $G(s)$ la transformación al dominio $s = \sigma + j\omega$ de $u(t)$, $y(t)$ y $g(t)$, la relación entre estas (en el caso SISO) se vuelve $Y(s) = G(s)U(s)$, con $G(s)$ típicamente una función producto de una división entre polinomios en s $D(s)$ y $N(s)$ en la forma $G(s) = \frac{N(s)}{D(s)}$. Al hablar de sistemas lineales en tiempo discreto $t = kT_s$, $k = 0, 1, \dots, T_s$ el tiempo de muestreo, se obtienen funciones de transferencia de estructura similar en el dominio de la frecuencia (ahora el dominio $z \in \mathbb{C}$).

Como se puede ver, en el caso lineal la estructura es bien conocida. Si el sistema que se desea identificar y modelar cumple con las condiciones de linealidad, entonces es posible describirlo con una función de transferencia (en el dominio s ó z , según sea el caso), con lo que ahora es necesario determinar un método para calcular los parámetros de la misma. En la literatura existen varios métodos para tal efecto, como los reportados en Nelles (2001). Sin embargo, en el caso de sistemas no lineales el análisis resulta más complejo, ya que no siempre es posible representarlos directamente mediante funciones de transferencia.

El análisis se complica en mayor medida cuando, además de ser un sistema no lineal, la estructura del mismo es poco conocida. En este caso han sido empleadas varias aproximaciones para identificar a dichas funciones: aproximaciones por series

polinomiales (Wiener/Hammerstein o Volterra). Sin embargo, el análisis de estabilidad de este tipo de sistemas suele ser más complejo.

Por otra parte, las redes neuronales y sistemas difusos han probado ser herramientas muy útiles en la identificación de funciones suaves en intervalos cerrados, capaces de lograr un error de aproximación aleatoriamente pequeño (dependiendo de la complejidad de los mismos).

1.2. Estado del arte

1.2.1. Redes neurodifusas

Las redes neuronales artificiales (llamadas simplemente *redes neuronales*) son funciones matemáticas inspiradas en las estructuras neuronales de los organismos biológicos, con lo cual buscan emular sus capacidades de aprendizaje, procesamiento de información, y adaptación. Una de las primeras aproximaciones matemáticas descriptivas de las acciones llevadas a cabo por las neuronas es dada por McCulloch & Pitts (1943). A partir de este trabajo, Rosenblatt (1957) propuso el *perceptrón*, una red neuronal simple. El proceso de lograr que una red *aprenda* a comportarse de una forma deseada es llamada *entrenamiento*, siendo el algoritmo más comúnmente usado el *algoritmo de retropropagación*, propuesto por Werbos (1974) y popularizado gracias al el libro de Rumelhart *et al.* (1986). Estos detonaron el empleo de las redes neuronales en la solución de diversos problemas, como la aproximación de funciones estáticas suaves desconocidas a partir de datos entrada/salida (Cybenko, 1989), su empleo para problemas de identificación y control (Narendra & Parthasarathy, 1990), e incluso en reconocimiento de caracteres (LeCun *et al.*, 1989).

Por otra parte, los sistemas difusos están inspirados en la forma en la que se expresa el conocimiento humano, así como el proceso deductivo asociado al mismo (Zadeh, 1973), mediante reglas en la forma “si X entonces Y”. Para expresar la ambigüedad de términos característica del lenguaje en una forma matemática se recurre a la teoría de la lógica difusa (Zadeh, 1965), donde mediante *conjuntos difusos* definidos a través de *funciones de membresía* se mapea la *pertenencia* de un elemento de un universo a un conjunto mediante valores en el intervalo $[0,1]$. En esta definición, 1 representa la pertenencia total a un conjunto, 0 la no pertenencia, y los valores entre 0 y 1 una pertenencia parcial.

Los sistemas difusos han sido empleados para resolver varios problemas, como el agrupamiento de datos¹, modelado y control de sistemas. Su uso en los primeros fue introducido por Dunn (1973) en el algoritmo *Fuzzy C-Means*, y mejorado por (Bezdek, 1981). Varios ejemplos de control y modelado, especialmente en los casos donde el modelo del sistema es complejo de obtener mediante el análisis de las ecuaciones constitutivas (Chiu, 1994; Passino, 1998; Wang, 1994; Yasunobu & Miyamoto, 1985).

La combinación de la estructura y técnicas empleadas en ambos sistemas (redes neuronales y sistemas difusos) es conocida como *sistema neurodifuso* o *red neurodifusa*. En general, estos términos se refieren a la forma en la cual se crea un sistema difuso a partir de algún método de aprendizaje motivado por la investigación en redes neuronales (Nauck *et al.*, 2005), la aproximación de funciones, clasificación y control. Una red neurodifusa provee de una partición del espacio de entrada que permite una interpretación en forma de reglas “si, entonces”, en contraste con las redes neuronales, donde dicha interpretación resulta ambigua.

Una de las primeras propuestas de controles basados en redes neurodifusas es reportado por Berenji *et al.* (1992). Por su parte, una de las estructuras más difundidas y empleadas para la aproximación de funciones es el modelo ANFIS² propuesto por Jang (1993).

1.2.2. Redes neurodifusas recurrentes

Las estructuras anteriormente descritas no han sido empleadas solamente como funciones estáticas de entrada-salida, sino también para la representación de sistemas dinámicos mediante la realimentación de señales dentro de la propia red, en cuyo caso se les conoce como *redes recurrentes*. En este caso, la dinámica presente debe ser tomada en consideración al proponer esquemas de aprendizaje para la identificación de sistemas dinámicos.

Una de las primeras aproximaciones es la red de Hopfield (1982), cuyo primer uso fue el aprendizaje de trayectorias mediante una memoria de un único bit. Sin embargo, se considera (Nauck *et al.*, 2005) que la primer red neurodifusa *recurrente* con una estrategia de aprendizaje fue desarrollada por Gorrini & Bersini (1994), donde la red

¹Esto es conocido por su nombre en inglés *clustering*.

²Por sus siglas en inglés: *Adaptive Neuro-Fuzzy Inference System*, Sistema de Inferencia Neurodifuso Adaptable

es entrenada empleando un algoritmo de retropropagación adaptado, conocido como *retropropagación recurrente en tiempo real*¹. En este sentido, Narendra & Parthasarathy (1990) propone el empleo de redes recurrentes con realimentación en las diversas capas para propósitos de identificación de sistemas y control. Por otra parte, Weibel *et al.* (1989) ha empleado redes recurrentes con retrasos de tiempo, y Williams & Zipser (1989) han propuesto redes completamente recurrentes para identificación (donde todas las capas contienen realimentación de sus propias señales de salida).

Por otra parte, se han propuesto redes globalmente estáticas y localmente recurrentes para identificación (Back & Tsoi, 1990; Wan, 1990) que emplean neuronas con filtros lineales. Igualmente, existen algoritmos que toman en cuenta la propia dinámica de la red, como la *retropropagación a través del tiempo* (Rumelhart *et al.*, 1986) y el *aprendizaje recurrente en tiempo real* (Williams & Zipser, 1989) mencionado anteriormente. Sin embargo, estos algoritmos tienden a ser lentos y no se garantiza su estabilidad como lo indica Mastorocostas & Theocharis (2002). Para sobreponerse a estos problemas se han propuesto algoritmos basados en aproximaciones lineales como mínimos cuadrados recursivos (Johansson, 1993; Passino, 1998) y el filtro de Kalman (Haykin, 2001), así como algoritmos elipsoidales vistos como una generalización del filtro de Kalman Rubio & Yu (2007) donde se requiere que las perturbaciones sean acotadas, en contraste con el filtro de Kalman que requiere de perturbaciones gaussianas.

En la literatura se han reportado diversas estructuras, las cuales en general dependen del problema específico a atacar. Por ejemplo, Mastorocostas & Theocharis (2002); Poznyak *et al.* (2006); Yu & Li (2004) han propuesto redes con una estructura en espacio de estados para identificación, observación y control, con algoritmos de entrenamiento estables. Sin embargo, dependen de la medición de los estados para lograr el entrenamiento de la red, los cuales no pueden ser medidos en determinadas circunstancias o donde solo existen mediciones entrada-salida del sistema.

En este sentido Juang & Lin (1999) proponen una estructura basada en redes neurodifusas con representación en espacio de estados, que únicamente requiere de la medición de datos entrada-salida y genera en forma auto-organizada cada regla y el número de estados. No obstante, el algoritmo propuesto carece de un análisis de estabilidad y las reglas carecen de alguna forma de interpretación, lo que provoca que se pierdan algunos

¹Conocido también por su nombre en inglés *Real-Time Recurrent Learning Algorithm*

de los beneficios de la parte difusa de la red y alguna forma de interpretación de las reglas.

1.3. Desarrollo del tema de investigación

1.3.1. Objetivo de la investigación

El presente trabajo tiene como objetivo principal el proponer una metodología de identificación para una clase de sistemas no lineales mediante el empleo de redes neurodifusas recurrentes, en tiempo discreto y tiempo continuo, mediante el empleo de datos entrada/salida, basado en cierta información estructural sobre los mismos.

Por una parte se considera la estructura discreta al poder modelarse la dinámica de un sistema mediante ecuaciones en diferencias dependientes de retrasos tanto de la señal de entrada como de salida. Bajo esta perspectiva se propone estudiar una estructura entrada-salida con una representación similar que, basada en redes neurodifusas recurrentes en tiempo discreto con un algoritmo convergente, cuente con un algoritmo de entrenamiento convergente y estable y que permita identificar la dinámica de (1.5).

Por otro lado, se estudian las condiciones bajo las cuales un sistema en tiempo continuo en la forma (1.2) puede ser identificado por una red neurodifusa recurrente que posea una estructura en variables de estado, así como que el algoritmo con el cual se entrenen los parámetros sea convergente bajo un análisis de estabilidad basado en la teoría de control. De esta misma forma, se analizará el caso donde la representación en variables de estado se lleve a cabo en tiempo discreto con el fin de identificar sistemas en la forma (1.4).

En todos estos casos se estudia si la red obtenida puede ser empleada como un modelo del sistema y considerarse como una aproximación aceptable de la dinámica del mismo, como se muestra en la Fig. 1.1.

Los objetivos particulares se enlistan a continuación:

- Obtener una metodología de identificación de sistemas no lineales en tiempo discreto, empleando una red neurodifusa recurrente con una estructura entrada/salida; donde el algoritmo de entrenamiento considere la sintonización tanto de los parámetros lineales como de los no lineales de la estructura, así como la estabilidad y convergencia del algoritmo de entrenamiento.

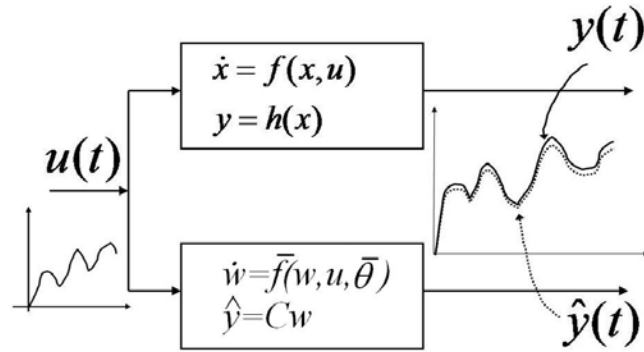


Figura 1.1: Identificación de un sistema mediante una red

- Obtener una metodología de identificación de sistemas no lineales en tiempo continuo, empleando una red neurodifusa recurrente con una estructura en variables de estado, donde se sintonicen los parámetros lineales de la red. El algoritmo de entrenamiento debe contar con un análisis de estabilidad y convergencia.
- Proponer estrategias de control basadas en las estructuras identificadas.
- Validar los resultados teóricos mediante su aplicación tanto en sistemas académicos como en sistemas físicos con datos obtenidos de forma experimental.

1.3.2. Principales contribuciones

Las principales contribuciones al estado del arte se enumeran a continuación, junto con las publicaciones donde han sido reportadas. En todos los casos, la identificación del sistema es llevada a cabo mediante mediciones entrada/salida, sin necesidad de realizar la medición directa de los estados físicos del sistema.

- Proponer una estructura entrada/salida para identificación de sistemas no lineales en tiempo discreto, con un algoritmo de aprendizaje (para parámetros lineales y no lineales) basado en el gradiente (Gonzalez Olvera & Tang, 2007).
- Con base en la estructura anterior, planteamiento de un aprendizaje basado en la linealización de los parámetros, con aprendizaje basado en algoritmos lineales (mínimos cuadrados, filtro de Kalman), así como propuestas de control basados en dicho esquema (Gonzalez-Olvera & Tang, 2007, 2008b).

- Mejoramiento de la estructura anterior mediante la limitación del rango de variación de los parámetros durante el entrenamiento, donde dicha información se encuentra directamente codificada dentro de la red (Gonzalez-Olvera & Tang, 2008a).
- Planteamiento de una red en tiempo continuo para identificación de una clase de sistemas no lineales, con una estructura en variables de estado, con un algoritmo de identificación inspirado en observadores adaptables (Gonzalez-Olvera & Tang, 2009a,b).
- Empleo de la red en tiempo continuo para el diseño de control de una clase de sistemas no lineales (Gonzalez-Olvera & Tang, 2009c).
- Empleo de la red en tiempo continuo para la identificación de diversos sistemas no lineales, a saber: sistema de tráfico vehicular (Gonzalez-Olvera & Tang, 2009e), robot de dos grados de libertad en un sistema de servo visión Gonzalez-Olvera *et al.* (2009a), sistema insulina-glucosa en pacientes diabéticos (Gonzalez-Olvera *et al.*, 2009b), entre otros.
- Aplicaciones derivadas de la investigación, como el uso de observadores adaptables en el problema de observación de velocidad de motores de corriente directa, en conjunto con estimación paramétrica; así como control basado en la red identificada (Berrospe *et al.*, 2009; Gonzalez-Olvera & Tang, 2008c).

1.3.3. Organización del trabajo de tesis

El trabajo de tesis se divide en la siguiente forma. En el capítulo 2 se presenta la teoría básica a emplearse en el desarrollo del tema de investigación, comenzando con el planteamiento básico de las redes neuronales y sistemas difusos en forma estática. Se justifica el empleo de dichas estructuras como aproximadores universales de funciones suaves en regiones cerradas, y se introduce el concepto de *recurrencia* de las mismas. A continuación, se describen algunos de los algoritmos empleados para su entrenamiento, como el filtro de Kalman y algoritmos basados en el gradiente, y se comenta la teoría de observadores adaptables. Finalmente, se discuten algunos algoritmos donde los sistemas difusos son empleados como mecanismos de agrupamiento de datos y algunas aplicaciones reportadas en la literatura.

A continuación, en los capítulos 3 y 4 se describen las estructuras propuestas para atacar el problema de identificación mediante redes neurodifusas recurrentes, en tiempo discreto y tiempo continuo, en variables de estado y en representación entrada-salida. En concreto, en el capítulo 3 se describe la estructura en tiempo discreto entrada-salida donde, a partir de datos se entrena a una red donde los parámetros cuentan con restricciones no lineales que no les permiten abandonar su zona de operación, evitando el empleo de algoritmos de proyección. Se muestra la identificación de diversos sistemas académicos para probar su efectividad. En un apartado se comenta la aplicación de esta red para problemas de control mediante equivalencia cierta y predicción un-paso-adelante, ya que este es un objetivo inmediato tras la identificación del sistema, así por la dificultad de controlar al sistema por otros métodos.

En el capítulo 4, por su parte, se presenta una estructura en tiempo continuo para la identificación de una clase de sistemas no lineales, con un algoritmo basado en la teoría de observadores adaptables, y se lleva a cabo el análisis de estabilidad y convergencia. Para probar la capacidad de identificación de la red, se identifican varios sistemas no lineales académicos. Finalmente, se emplea la red como modelo del sistema y se propone una solución al problema de control mediante una modificación de la estructura propuesta para una clase de sistemas, donde se comentan las líneas de investigación abiertas y trabajo futuro a realizar en el área. Con base en esta estructura, se analiza una versión en tiempo discreto también en variables de estado.

En el capítulo 5 se muestran diversas aplicaciones de la red para identificar sistemas físicos: un motor de corriente directa, un robot flexible de un eslabón, un sistema de tráfico vehicular, un robot de dos grados de libertad montado en un sistema de servo visión y un sistema biológico de glucosa-insulina en pacientes diabéticos. En tres de estos casos (motor, tráfico y servo visión), los datos necesarios para la identificación fueron obtenidos de forma experimental.

Finalmente, en el capítulo 6 se comentan las conclusiones a las que lleva el trabajo de tesis, los principales resultados obtenidos, los productos de investigación generados y las líneas de investigación abiertas.

1. INTRODUCCIÓN

Capítulo 2

Preeliminaries

En este capítulo se presentan varios elementos de teoría empleados a lo largo del trabajo de tesis. En primer lugar, se discute la estructura general de las redes neuronales, los sistemas difusos, las redes neurodifusas, su empleo como aproximadores universales de funciones en dominios cerrados y algunos de los algoritmos de entrenamiento existentes. Se menciona también el empleo de sistemas difusos como algoritmos de agrupamiento. A continuación, se menciona el concepto de *red recurrente*, algunas de las estructuras más importantes mencionadas en la literatura y los métodos de entrenamiento empleados. Finalmente se discute la teoría de observadores adaptables, la cual será empleada en capítulos posteriores.

2.1. Redes neuronales

Las redes neuronales *artificiales* son estructuras matemáticas inspiradas en las redes neuronales biológicas, cuyo principal propósito es el emular (de manera simplificada) la capacidad de aprendizaje, adaptabilidad y procesamiento de información de estas últimas a partir de elementos muy sencillos (neuronas) conectadas en paralelo. En la literatura, es común eliminar el adjetivo *artificial* y referirse a estas estructuras simplemente como *redes neuronales*.

De este modo, las redes neuronales han sido empleadas como herramientas matemáticas que, a partir de datos, modelen las relaciones entre las señales de entrada y salida de un cierto proceso, sistema o datos.

El elemento fundamental de una red neuronal es la *neurona*, mostrada en la Fig. 2.1.a, la cual es un elemento que, a partir de un vector de entradas $\mathbf{x} \in \mathfrak{R}^n$ y uno de parámetros $\mathbf{b} \in \mathfrak{R}^m$, realiza una función matemática simple $f(\mathbf{x}, \mathbf{b})$ la cual entrega en la salida y . La función $f(\cdot, \cdot)$ puede ser de distinta naturaleza: por ejemplo, $f(\mathbf{x}, \mathbf{b}) = \sum_{i=1}^n x_i$ (lineal), $f(\mathbf{x}, \mathbf{b}) = \sum_{i=1}^n b_i x_i$ (lineal), $f(\mathbf{x}, \mathbf{b}) = \tanh(\sum_{i=1}^n b_i x_i + \sum_{i=n+1}^m b_i)$ (no lineal, con $m > n$) o $f(\mathbf{x}, \mathbf{b}) = \sum_{i=1}^n b_i \tanh(x_i)$.

Una red neuronal se compone de varias neuronas conectadas entre sí, como se muestra en la Fig. 2.1.b. Con un mayor número de neuronas, es posible crear mapeos $f : \mathbf{x} \rightarrow y$ más complejos mediante la manipulación tanto de las funciones como de los parámetros de cada neurona y de este modo emplearlas para identificar, aproximar y modelar procesos no lineales.

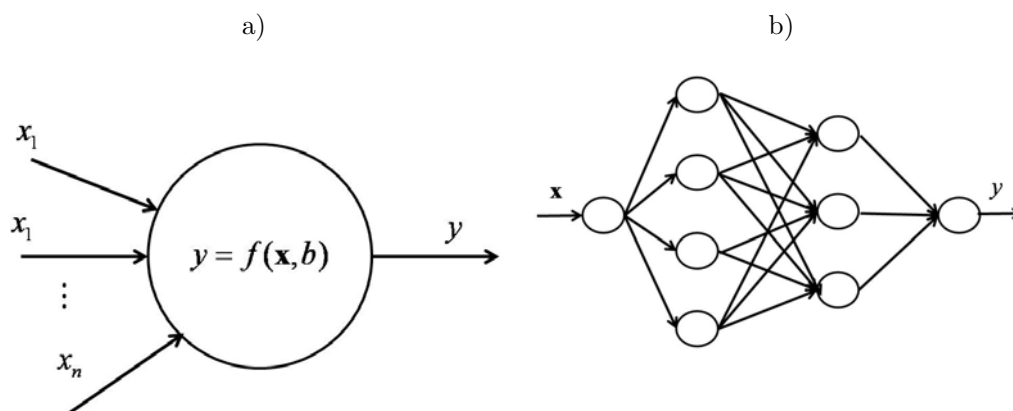


Figura 2.1: a) Neurona, b) Red Neuronal

Para tal efecto, una red debe ser *entrenada*; esto es, debe encontrarse un método que permita modificar tanto su estructura (las conexiones entre neuronas) como sus parámetros, el cual es conocido como *algoritmo de aprendizaje* o *algoritmo de entrenamiento*.

2.2. Sistemas difusos

Los sistemas difusos están basados en la lógica difusa, creada por Zadeh (1965), la cual es una extensión de la lógica de Boole. La lógica difusa extiende el sentido de *pertenencia* de un elemento a un conjunto, al permitir que pertenezca a éste en forma

parcial, a diferencia de lo que sucede con la lógica de Boole, donde la pertenencia o no pertenencia son absolutas. De este modo, la pertenencia o no de un elemento \mathbf{x} a un conjunto X de un universo \mathcal{U} , $X \subseteq \mathcal{U}$ se puede expresar mediante un par de números reales. Por ejemplo, en lógica Booleana se puede definir a la *función de pertenencia* de \mathbf{x} a X mediante un mapeo $\mu_X(\mathbf{x}) : X \rightarrow \mu$, $\mu = \{\bar{\mu} \in \mathfrak{R} \mid \bar{\mu} \in \{0, 1\}\}$, donde

$$\mu_X(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in X \\ 0, & \mathbf{x} \notin X. \end{cases} \quad (2.1)$$

En lógica difusa, la pertenencia a un conjunto se define como un número real, de modo que el mapeo se determina mediante $\mu = \{\bar{\mu} \in \mathfrak{R} \mid \bar{\mu} \in [0, 1]\}$, que se interpreta como el *grado de pertenencia* del elemento \mathbf{x} a X , de modo que puede pertenecer en forma *parcial* a este conjunto. Entre más cercano sea el valor de μ a cero, \mathbf{x} pertenecerá en menor grado a X , e inversamente cuando el grado de pertenencia está cercano a la unidad.

Este tipo de lógica resulta muy conveniente para dar un sentido matemático a las expresiones del lenguaje humano, donde se da cabida a expresiones ambiguas e inciertas. Por ejemplo, al indicar cómo se debe de detener un automóvil en movimiento al acercarse a una señal de alto, se emplean expresiones o *reglas* en la forma:

Si	la velocidad es alta y el punto donde deseamos detenernos está lejos
entonces	se debe frenar con suavidad.
Si	la velocidad es media y el punto donde deseamos detenernos está cercano
entonces	se debe frenar con firmeza.
	⋮
Si	la velocidad es alta y el punto donde deseamos detenernos está cercano
entonces	se debe frenar al máximo

Desde el punto de vista humano, si la velocidad del auto se sitúa entre alta y media y el punto a detenerse se encuentra en un punto cercano, entonces se razonará que se debe llegar a un punto de frenado entre el máximo y con firmeza, intensidad que variará en la medida que los valores de velocidad y distancia se modifiquen a lo largo de la acción de frenado.

Si se desea transmitir este conocimiento a un sistema de cómputo, dicho lenguaje será de difícil interpretación, ya que éstas operan con lógica booleana y funciones matemáticas. El principal problema es la interpretación de las *variables lingüísticas* velocidad y distancia en términos de las *etiquetas lingüísticas* alta, media, lejano, cercano.

Combinando los conceptos de lógica difusa, es posible interpretar la pertenencia de un elemento a un conjunto a partir de una función de pertenencia descrita mediante una función matemática, como las mostradas en la Fig. 2.2.

Nombre	Función	Representación
Triangular	$f(x) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-c}, & x \in [a, b] \\ -\frac{x-b}{b-c}, & x \in [b, c] \\ 0, & x > c \end{cases}$	
Sigmoide	$f(x) = \begin{cases} 0, & x < a \\ 2 \left(\frac{x-a}{c-a} \right)^2, & x \in (a, b] \\ 1 - 2 \left(\frac{x-a}{c-a} \right)^2, & x \in (b, c] \\ 1, & x > c \end{cases}$	
Gaussiana	$f(x) = e^{-\sigma^2(x-\mu)^2}$	

Figura 2.2: Ejemplos de funciones de membresía

De este modo, cada etiqueta lingüística se puede asociar con un conjunto difuso, creándose una *base de conocimiento* mediante reglas en la forma *Si... entonces...* Aquella parte asociada con el condicional *si* se conoce como *antecedente*, mientras que a su consecuencia *entonces* se le conoce como *consecuente*.

Mediante operaciones lógicas definidas para la lógica difusa, es posible combinar los grados de pertenencia de los elementos antecedentes en cada regla y obtener el *nivel de activación* de cada una. Con respecto al ejemplo anterior, es necesario determinar el *grado de verdad* de la *velocidad alta* y *punto lejano* de acuerdo con el grado de pertenencia de la velocidad al conjunto asociado con la etiqueta lingüística *velocidad alta*, así como de la distancia con *lejano*.

Este proceso se lleva a cabo mediante operaciones que generalizan las operaciones de unión o intersección de conjuntos con base en los valores de membresía de sus elementos. En particular, para un sistema con n entradas $x_i, i = 1, \dots, n$, la operación conjunción

x_1 es $A_{i,1}$ Y ... Y x_n es $A_{i,n}$ se conoce como *t-norma*. Al ser este el nivel de activación de cada regla R_i , se le describe mediante $R_i(x_1, \dots, x_n) = A_{i1}(x_1) \wedge A_{in}(x_n)$, y puede ser definida matemáticamente de diversas formas, las cuales son ampliamente presentadas en Kosko (1992); Nelles (2001).

Para el caso del presente trabajo, se tomará como t-norma al producto entre las funciones de membresía de la parte antecedente, es decir $R_i(x_1, \dots, x_n) = \prod_i A_{i1}(x_1)$.

Una vez obtenido el nivel de activación de cada regla, se lleva a cabo el proceso de *agregación*, el cual consiste en combinar la parte consecuente de las mismas. Dependiendo de la estructura definida en la parte consecuente de las reglas, los sistemas difusos se pueden dividir en dos grandes grupos: sistemas tipo Mamdani y sistemas tipo Takagi-Sugeno, con procesos de agregación diferentes entre sí. Los primeros consideran reglas en la forma

$$R_i: \text{Si } x_1 \text{ es } A_{i1} \text{ y } \dots \text{ y } x_n \text{ es } A_{in} \text{ , entonces } y \text{ es } B,$$

donde tanto A como B son conjuntos difusos; mientras que los segundos están formados por reglas en la forma

$$R_i: \text{Si } x_1 \text{ es } A_{i1} \text{ y } \dots \text{ y } x_n \text{ es } A_{in} \text{ , entonces } y = f_i(x_1, \dots, x_2).$$

Al proceso de obtener la salida del un sistema difuso tras la *agregación* se le conoce como *desdifusión*. En el caso de los sistemas Mamdani, que consideran conjuntos difusos como salida, se emplean métodos diversos como el cálculo del centroide del proceso de agregación, cálculo de promedio de máximos, etc., los cuales suelen ser complejos de representar con funciones matemáticas compactas. Se puede encontrar una relación de éstos en Driankov *et al.* (1996).

Por su parte, los sistemas difusos tipo Takagi-Sugeno, al considerar funciones en la parte consecuente de cada regla difusa, permiten calcular la salida mediante el promedio ponderado

$$y = \frac{\sum_i R_i f_i(x_1, \dots, x_n)}{\sum_i R_i}. \quad (2.2)$$

Esta relación se puede escribir de un modo más compacto mediante

$$\varphi_i = \frac{R_i}{\sum_i R_i}, \quad (2.3)$$

$$y = \sum_i \varphi_i f_i. \quad (2.4)$$

2.3. Redes neurodifusas

Los modelos expresados en la forma Takagi-Sugeno han sido comúnmente empleados para problemas de modelado y control (Tanaka & Wang, 2001) gracias a que permiten tratarlos como redes neuronales. En general, el término *neurodifuso* se refiere, de acuerdo con Nauck *et al.* (2005), a la combinación de técnicas propias de los sistemas difusos y las redes neuronales. De forma estricta, comenzaron siendo sistemas difusos a los cuales se les aplicaban los algoritmos de aprendizaje diseñados para redes neuronales al analizárseles como una estructura neuronal. Uno de los ejemplos más populares es el algoritmo/estructura ANFIS propuesta por Jang (1993). Sin embargo, a lo largo de la literatura se ha conservado el nombre *red neurodifusa* para todos aquellos sistemas difusos que son entrenados a partir de datos (Nelles, 2001).

En particular, las redes neurodifusas han sido empleadas exitosamente como aproximadores de funciones no lineales. Sin embargo, a diferencia de otro tipo de estructuras cuentan con la ventaja de permitir la interpretación de los resultados obtenidos en cada regla: Cada regla aproxima una sección del espacio definida por la parte antecedente de la regla R_i a partir de las funciones f_i . Es necesario tener en cuenta que este proceso de *interpretación* no es directo ni necesariamente automático, ya que depende del número de reglas, la forma en la cual se realiza la partición del espacio, etc. En Nauck *et al.* (2005) se da un panorama general del desarrollo de las redes neurodifusas hasta el año 2005.

2.3.1. Aproximadores universales

De acuerdo con el *Teorema Universal de Aproximación* (Wang, 1994), dado un sistema difuso o una red neuronal de n entradas con funciones de membresía gaussianas representado por

$$f(\mathbf{u}) = \frac{\sum_{l=1}^M \bar{y}^l \prod_{i=1}^n a_i^l e^{-(x_i - \mu_{i,l})^2 \sigma_{i,l}^2}}{\sum_{l=1}^M \prod_{i=1}^n a_i^l e^{-(x_i - \mu_{i,l})^2 \sigma_{i,l}^2}} \quad (2.5)$$

donde $x \in \mathbb{R}^n$, M el número de reglas, $\mu_{i,l}$ y $\sigma_{i,l}$ el centro y ancho de las funciones gaussianas que representan a los conjuntos difusos de la entrada i y la regla l respecti-

vamente, a_i^l es el peso que se le da a cada conjunto difuso, es posible aproximar cualquier función suave en un conjunto compacto, de acuerdo con el siguiente teorema:

Teorema 2.1 (Teorema Universal de Aproximación). *Para cualquier función real y continua g definida en un conjunto compacto $U \subset \mathbb{R}^n$ y un número $\varepsilon \in \mathbb{R}$, $\varepsilon > 0$ existe un sistema difuso en la forma (2.5) tal que*

$$\sup_{\bar{w} \in U} |f(\bar{w}) - g(\bar{w})| < \varepsilon \quad (2.6)$$

La demostración de este teorema se describe en Wang (1994), en forma más explícita en Tanaka & Wang (2001), basada en el trabajo de Cybenko (1989) y en el Teorema de Stone-Weierstrass, donde se demuestra que con una combinación de varias funciones sigmoideas se puede aproximar la clase de funciones requeridas por el teorema. Un corolario del teorema permite definir a la norma empleada:

Corolario 2.1 (Wang (1994)). *Para cualquier función $g \in \mathcal{L}_2(U)$ ¹ y $\varepsilon > 0$ existe un sistema difuso f en la forma (2.5) tal que*

$$\left(\int_U |f(\bar{w}) - g(\bar{w})|^2 d\bar{w} \right)^{\frac{1}{2}} < \varepsilon \quad (2.7)$$

donde $U \subset \mathbb{R}^n$ es compacto, $\mathcal{L}(U) = [g : U \rightarrow \mathbb{R} | \int_U |g(\bar{w}^2)| d\bar{w} < \infty]$ con las integrales en el sentido de Lebesgue.

Este teorema y su corolario indican que una red aproxima con precisión arbitraria una función tal que cumpla con las características indicadas, aunque se debe notar que no hace referencia al grado de complejidad que el sistema difuso ha de tener ni el cómo obtener dichas funciones; razón por la cual se debe entender a este teorema y su corolario como una justificación del uso de sistemas difusos para atacar el problema de identificación de funciones no lineales.

2.4. Redes recurrentes

Cuando la salida de un modelo no solo depende de las entradas en el tiempo actual sino además de valores anteriores de éstas, se habla del concepto de *modelos dinámicos*,

¹Un espacio \mathcal{L}_2 se define aquel que contiene a todas las funciones u tales que cumplan con que su norma $\|u\|_{\mathcal{L}_2} = \sqrt{\int_0^\infty u(t)^T u(t) dt} < \infty$

modelos recurrentes, o simplemente *recurrencia*, lo que les hace propicios para analizar sistemas dinámicos.

De acuerdo con Nauck *et al.* (2005), una de las primeras estructuras neurodifusas propuestas junto con un método de entrenamiento, fue presentada por Gorrini & Bersini (1994), donde se emplea una estructura tipo Takagi-Sugeno con parte consecuente constante, donde las variables internas (equivalentes a los estados) deben ser elegidas manualmente, y sin un método sistemático (a no ser aleatorio) para entrenar a la parte antecedente. De igual modo, la estructura carece de una facilidad de interpretación de los parámetros obtenidos.

Entre las principales estructuras empleadas para identificar sistemas dinámicos mediante neurodifusas recurrentes se encuentran las siguientes:

- Theocharis & Vachtsevanos (1996a). Esta es una red recurrente con realimentación externa, en forma de Takagi-Sugeno, con parte consecuente basada en funciones polinomiales. Esta estructura es empleada como identificador de sistemas no lineales y como filtro adaptable para atenuación de ruido. Por su parte, el artículo publicado por Theocharis & Vachtsevanos (1996b) propone un aprendizaje tanto de estructura (primer aprendizaje) como de parámetros (segunda fase).
- Zhang & Morris (1999). Propone una estructura neurodifusa en tiempo discreto para el modelado de procesos mediante una red recurrente con reglas que consideran sistemas lineales locales, con un entrenamiento basado en el algoritmo de Levenberg-Marquardt. Se propone además una estrategia de control basada en la red entrenada, donde se proponen controles locales para cada sistema.
- Lin (1996). Esta estructura es muy parecida a la propuesta más tarde por Lin & Hsu (2001), donde la dinámica se lleva a cabo en la primer capa de la red mediante una realimentación de primer orden a funciones gaussianas. Sin embargo, la principal contribución es la propuesta de esquemas de control directo e indirecto que emplean la red, así como un análisis de convergencia del algoritmo. Sin embargo, no se puede considerar que la red sea completamente recurrente, ya que la realimentación se restringe a la primera capa. Asimismo, es difícil interpretar los resultados obtenidos tras el entrenamiento.

- Juang & Lin (1999). Esta red está igualmente basada en redes neurodifusas recurrentes, con un algoritmo de entrenamiento fundamentado en la aplicación del gradiente en forma temporal (provocando un algoritmo de entrenamiento igualmente recurrente). A diferencia de Lin (1996), la recurrencia se da desde las últimas capas de salida de la red (parte consecuente) hasta la parte antecedente de la misma. En versiones posteriores (Juang, 2002), incorpora un aprendizaje de parámetros y estructura basado en algoritmos genéticos. Sin embargo, pese a ser una estructura neurodifusa, no es posible realizar una interpretación sobre la partición del espacio proporcionada por los conjuntos difusos o un análisis de la estabilidad del algoritmo.
- Nürnberger & Kruse (2002). Se propone una mayor complejidad para las redes mediante un modelo neurodifuso cuyo objetivo es optimizar bases de reglas recurrentes en estructuras jerárquicas. Sin embargo, no logra realizarlo sin intervención del usuario o un experto. En Nürnbergger (2001) se propone igualmente una red jerárquica, donde se imponen además restricciones a los parámetros de modo que no queden zonas sin cubrir.
- Poznyak *et al.* (2001). En este libro se propone una estructura de red recurrente en tiempo continuo y discreto en forma de variables de estado para identificar sistemas donde los estados sean directamente medibles, con los cuales se logra entrenamiento de la red. Diversos trabajos derivados (Rubio & Yu, 2007) consideran algoritmos estables y convergentes, bajo suposiciones de acotamiento de las señales de perturbación; e incluso se proponen condiciones relativamente generales (Poznyak *et al.*, 2006) para las ganancias de aprendizaje y su aplicación a sistemas físicos. Una vez entrenada, la red permite realizar el diseño de observadores y controladores basados en la estructura. Sin embargo, sigue requiriendo de la medición de estados para lograr el entrenamiento de la red (previo a su utilización donde no se requiere ya dicha medición).
- Gonzalez-Olvera & Tang (2008a). En este trabajo se propone una red entrada/salida en tiempo discreto, similar a la propuesta por Zhang & Morris (1999), para identificación y control de sistemas por medio de una red neurodifusa recurrente. Sin embargo, se sugiere además la modificación del algoritmo para entrenar

de forma estable los parámetros antecedentes. En modificaciones posteriores, se proponen restricciones no lineales (Gonzalez-Olvera & Tang, 2009d) de modo que no se permita que zonas queden descubiertas.

Se pueden encontrar diversas propuestas de identificación control mediante estructuras inspiradas en redes neuronales, sistemas difusos y redes neurodifusas en el libro Ruano (2005). En el caso de estructuras donde se emplean sistemas difusos con reglas que involucran sistemas lineales locales, la estabilidad y existencia de puntos de equilibrio se describe en Kempf & Adamy (2003) y Kosko (1992).

2.5. Métodos existentes de entrenamiento

Existen varios métodos de entrenamiento para sistemas difusos y redes neuronales. El más popular de ellos es quizás es *algoritmo de retropropagación* propuesto por Werbos (1974), que no es más que la aplicación de un algoritmo tipo gradiente de una función de costo del error de identificación con respecto a los parámetros. Sin embargo, este algoritmo ha sido modificado para lograr hacerlo más rápido y convergente mediante métodos conjugados de gradiente, cuasi-Newton, modificando en forma dinámica la ganancia de adaptación, al añadir términos de momento, etcétera (Hagan & Menhaj, 1994). Se emplean igualmente métodos basados en sistemas linealmente parametrizables, como mínimos cuadrados o el filtro de Kalman, y en general este tipo de métodos se distingue por no hacer mayores suposiciones sobre la topología de la función de costo, a no ser el suponer que es una función suave. Sin embargo, se sabe que estos algoritmos presentan problemas al aprender dinámicas de *largo plazo* (Bengio *et al.*, 1994).

2.5.1. Mínimos cuadrados

Supóngase que se cuenta con mediciones de la señal de salida y y entrada \mathbf{x} de un cierto proceso y se desea obtener una función, basada en redes neurodifusas, que describa su relación. Si la estructura difusa o neuronal es linealmente parametrizable, entonces la salida se expresa mediante

$$y = \phi(\mathbf{x})\theta + e, \tag{2.8}$$

donde $\mathbf{x} \in \mathfrak{R}^n$ es el vector de entradas, $\phi : \mathfrak{R}^n \rightarrow \mathfrak{R}^p$ el *regresor* de funciones conocidas, $\theta \in \mathfrak{R}^p$ el vector de parámetros a conocer. Sea una serie de N mediciones de la señal de entrada y salida:

$$\Phi = [\phi(X(1)) \quad \dots \quad \phi(X(N))]^T = [\phi(1) \quad \dots \quad \phi(N)]^T, \quad (2.9)$$

$$Y = [Y(1) \quad \dots \quad Y(N)]^T, \quad (2.10)$$

de modo que es posible representar al error entre la señal y su cálculo mediante la red $\hat{Y} = \Phi(\mathbf{x})\hat{\theta}$ mediante

$$E = Y - \Phi\hat{\theta}. \quad (2.11)$$

Varios algoritmos permiten calcular la aproximación $\hat{\theta}$ de modo que se minimice una cierta función de costo. Algoritmos típicamente empleados para este propósito son los mínimos cuadrados, el filtro de Kalman y algoritmos elipsoidales.

La función de costo más comúnmente usada es la suma de los errores al cuadrado

$$J(\theta) = \frac{1}{2}E^T E = \frac{1}{2} (Y - \Phi\hat{\theta})^T (Y - \Phi\hat{\theta}), \quad (2.12)$$

De esta forma, se puede determinar que el valor de los parámetros que minimiza a la función de costo (2.12) es

$$\hat{\theta} = (\Phi^T \Phi)^{-1} \Phi^T f. \quad (2.13)$$

Dado que el cálculo de la matriz $(\Phi^T \Phi)^{-1}$ puede llevar a singularidades, suele emplearse el método recursivo para mínimos cuadrados, el cual considera incluso su utilización en sistemas en tiempo real mediante la expresión de adaptación de parámetros:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \bar{\gamma}(k)e(k), \quad (2.14)$$

$$e_k = y_k - \hat{y}_k = y_k - \phi_k^T \hat{\theta}_{k-1}, \quad (2.15)$$

$$\gamma(k) = \frac{P_{k-1} \phi_k}{\phi_k^T P_{k-1} \phi_k + \lambda}, \quad (2.16)$$

$$P_k = (I - \gamma_k \phi_k^T P_{k-1}). \quad (2.17)$$

Por lo general, se suele iniciar el valor de P con $P(0) = \alpha I$, donde $\alpha \gg 1$. Este método puede ser aplicado para modificar los parámetros en línea, cuando estos varían con el tiempo. Por ello, se incluye al término λ , el cual es llamado *factor de olvido*, y

que permite que los datos más recientes tengan mayor peso que los antiguos, con la desventaja de volver al algoritmo más sensible a perturbaciones.

2.5.2. Filtro de Kalman

El algoritmo anterior puede ser diseñado de modo que se tome en cuenta cierta información sobre las perturbaciones y ruido en las señales medidas. Por ello, si se asume que el sistema original está perturbado por señales de ruido gaussiano, no correlacionado y blanco de media cero ν_k en la forma

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \nu_k \quad (2.18)$$

$$y_k = \phi_k \theta_k + e_k \quad (2.19)$$

donde se sabe que la varianza de ν_k está dada por $\mathcal{E}\{\nu\nu^T\} = R_1$, y la de e_k por $\mathcal{E}\{e^2\} = R_2$, entonces el cálculo de θ_k basado en el filtro de Kalman está dado por

$$\hat{\theta}_k = \hat{\theta}_{k+1} + K_k e_k, \quad (2.20)$$

$$K_k = \frac{P_{k-1} \phi_k}{R_2 + \phi_k^T P_{k-1} \phi_k}, \quad (2.21)$$

$$e_k = y_k - \phi_k^T \hat{\theta}_{k-1}, \quad (2.22)$$

$$P_k = P_{k-1} - \frac{P_{k-1} \phi_k \phi_k^T P_{k-1}}{R_2 + \phi_k^T P_{k-1} \phi_k} + R_1. \quad (2.23)$$

donde típicamente se relaciona a R_2 con el llamado *factor de olvido*, el cual es empleado en aplicaciones en tiempo real para dar más peso a los datos más recientes y *desvanecer* el efecto de los más antiguos. Generalmente se emplea $R_2 = 1$ cuando no se desea dicho efecto, y se decrementa (generalmente en valores no menores de $R_2 = 0,9$) conforme se necesite incrementar esta acción.

Es claro que estas ecuaciones son un caso más general que aquellas del método de los mínimos cuadrados. Sin embargo, aunque este algoritmo ha sido ampliamente utilizado en una gran cantidad de aplicaciones gracias a su efectividad y rechazo a perturbaciones, cuando $\phi_k = 0$ se tiene un crecimiento exponencial de P_k , además que esta misma matriz no tiende a cero si $k \rightarrow \infty$ (Johansson, 1993). En general, el análisis de estabilidad del filtro de Kalman ha sido objeto de estudio, y en general se admite que su análisis tiende a ser de gran dificultad (Rubio & Yu, 2007). Sin embargo, si las

perturbaciones son acotadas, es posible obtener algoritmos elipsoidales (Rubio & Yu, 2007) con análisis de estabilidad y convergencia.

Las aplicaciones del filtro de Kalman a redes neuronales han probado su utilidad en una gran variedad de aplicaciones (Haykin, 2001). Su empleo busca proveer con una aproximación al estimado de la varianza bajo una estructura lineal en los parámetros. Bajo algunas condiciones, el entrenamiento basado en el filtro de Kalman ha sido analizado como estable (Guo, 1990). Sin embargo, la estabilidad de los esquemas basados en el filtro de Kalman se vuelve difícil de analizar (a diferencia del análisis para el caso particular de mínimos cuadrados recursivos), como ya ha sido analizado por Hagner *et al.* (2000) y de Jesús Rubio & Yu (2007).

2.5.3. Algoritmos basados en el gradiente

Los algoritmos basados en el gradiente son los más comunes, dada su relativa facilidad de implementación incluso para sistemas sin una parametrización lineal, es decir, donde la relación entre la salida medida y el regresor se define como

$$y_k = \phi(\mathbf{x}_k, \theta_k) + e, \quad (2.24)$$

con e el error de aproximación. El principio básico de operación es el modificar el vector de parámetros calculados $\hat{\theta}_k$ en forma proporcional a una *ganancia de adaptación* η_k en la dirección de mayor inclinación dada por el gradiente de la función de costo $G_k = \frac{\partial J_k}{\partial \theta}$ y una dirección P_{k-1} mediante

$$\hat{\theta}_k = \hat{\theta}_{k-1} - \eta_{k-1} P_{k-1}, \quad P_{k-1} = R_{k-1} G_{k-1}, \quad (2.25)$$

donde R_{k-1} es la escala y rotación de la dirección de búsqueda de forma que la función de costo decrezca en cada paso de iteración. Se pueden hacer diversas elecciones de η_k y R_k , así como en la forma de calcular u aproximar G_k . Diversas elecciones conducen a desempeños diferentes (mayor rapidez, mayor precisión), aunque determinar la estabilidad de estos algoritmos no suele ser sencillo. Además, suele existir un compromiso entre la velocidad de convergencia y el nivel de precisión logrado. Un resumen de las diferentes elecciones posibles se encuentra en Nelles (2001). Como se mencionó anteriormente, este método es la base para el algoritmo de retropropagación propuesto por Werbos (1974).

2.6. Algoritmos de agrupamiento

Las técnicas de agrupamiento se refieren a la búsqueda de grupos de datos con características similares mediante una *medición de similitud*. Existen algoritmos donde previo al agrupamiento se define el número de grupos; mientras que otros generan los grupos directamente tras el análisis de los datos, a partir de una *medida de granularidad* dada por el usuario.

Los algoritmos clásicos de agrupamiento de datos consideran conjuntos con fronteras bien definidas (un dato medido puede o no pertenecer a un determinado grupo), como se muestra en la Fig. 2.3. Técnicas más modernas consideran el empleo de conjuntos difusos, donde cada dato puede tener un *grado de membresía* a diversos grupos. Asimismo, es posible emplear algoritmos jerárquicos de agrupamiento, donde dentro de un mismo grupo de datos se pueden volver a dividir en subgrupos, y así sucesivamente. Generalmente, los grupos se definen a partir de pocos datos. En el caso de grupos circulares, se definen con el centro c_i y el radio r_i del mismo.

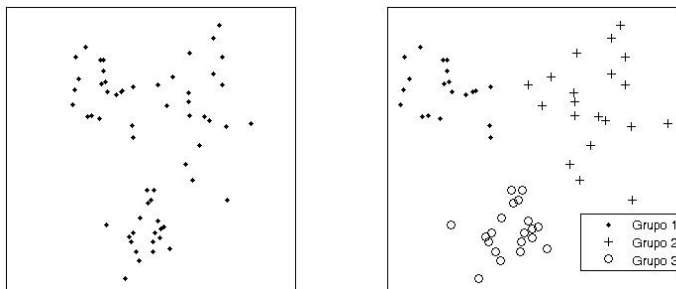


Figura 2.3: Agrupamiento de datos

2.6.1. Algoritmo de k-medias (*K-Means*)

Dado un conjunto de datos $X = [X_1 \ \dots \ X_N]$ y de centros c_i , con un número predeterminado de grupos, este algoritmo busca minimizar la función de costo

$$J = \sum_{j=1}^C \sum_{i \in S_j} \|X_i - c_j\|^2 = \sum_{j=1}^C \sum_{i=1}^N \mu_{ij} \|X_i - c_j\|^2 \quad (2.26)$$

donde C es el número de grupos, S_j cada grupo (compuesto por los índices de los elementos X_i pertenecientes al conjunto S_j) y c_j los centros de éstos. La segunda ecuación es una expresión compacta, donde $\mu_{ij} = 1$ si $X_i \in S_j$, y $\mu_{ij} = 0$ si $X_i \notin S_j$.

Existen varios métodos para minimizar esta función de costo manipulado a los centros c_j , entre las que se encuentran el algoritmo de MacQueen (se eligen valores iniciales para los grupos, quizás en forma aleatoria) y el Gustafson-Kessel (cada grupo cuenta con su propia función de distancia). Una descripción más detallada de este tipo de algoritmos se encuentra reportada por Anderberg (1973).

2.6.2. Algoritmo difuso de c-medias (*Fuzzy C-Means*)

Este algoritmo es una versión difusa el algoritmo de k-medias descrito anteriormente, donde la función de costo continúa siendo

$$J = \sum_{j=1}^C \sum_{i=1}^N \mu_{ij} \|X_i - C_j\|^2, \quad (2.27)$$

con la diferencia de que el grado de pertenencia μ_{ij} del dato X_i al grupo S_j puede variar entre $[0, 1]$, donde 1 corresponde al centro de los conjuntos difusos, y decrece medida que los datos se alejan de los mismos. Estas funciones de pertenencia han sido definidas mediante

$$\mu_{ij} = \frac{1}{\sum_{l=1}^C \left(\frac{D_{ij,R}^2}{D_{il,R}^2} \right)^{\frac{1}{\gamma-1}}} \quad (2.28)$$

donde R es una matriz de rotación dentro de la definición de distancia $D_{ij,R}^2 = (X_i - c_j)^T R (X_i - c_j)$, y γ permite definir el grado de decaimiento de la función (cercano a 1 para grupos fácilmente separables, valores grandes para grupos con gran grado de superposición). El algoritmo para minimizar esta función de costo, descrito en Bezdek (1981), consiste en que a partir de una elección arbitraria de los centros, se calculan las distancias de cada elemento X_i a los grupos C_j , y el centroide de éstos últimos (ponderado por los grados de membresía), el cual será empleado para modificar los centros. Si los cambios son mayores que un grado ϵ definido previamente, se continúa con el procedimiento, hasta que los cambios no sean mayores que dicho ϵ .

Algunas variaciones de este método han sido popularizadas gracias a su inclusión dentro del paquete de cómputo *Matlab*, en particular el algoritmo de identificación de modelos difusos basado en cálculo de grupos propuesto por Chiu (1994), donde los grupos son definidos por funciones de membresía gaussianas y se permite la creación o eliminación de grupos (que son interpretados ahora como la parte antecedente de reglas difusas). Este tipo de algoritmos son conocidos también como *mapas autoorganizados* (Kohonen, 2001). Más detalles sobre estos algoritmos se encuentran reportados por Bezdek (1981); Dunn (1973).

2.7. Observadores adaptables

Considérese un sistema de una entrada-una salida descrito por

$$\dot{\mathbf{x}} = f(\mathbf{x}, u), \quad (2.29)$$

$$y = h(\mathbf{x}). \quad (2.30)$$

donde $u, y \in \mathfrak{R}$, $f : \mathfrak{R}^n \times \mathfrak{R} \rightarrow \mathfrak{R}^n$, $h : \mathfrak{R}^n \mathfrak{R}$ son funciones suaves, con f Lipschitz y (f, h) observables dentro de una región de interés. Cuando no es posible medir todos los estados \mathbf{x} del sistema y solo pueden medirse las salida y y las entrada u , es posible diseñar un algoritmo que, con base en esta información, pueda *reconstruir* a \mathbf{x} mediante $\hat{\mathbf{x}}$ (siempre y cuando el sistema sea *observable*).

En el caso lineal, sabemos que $f(\mathbf{x}, u) = A\mathbf{x} + Bu$ y $h(\mathbf{z}, u) = C\mathbf{x} + Du$, de modo que es posible transformar mediante $\mathbf{z} = T\mathbf{x}$ al sistema en su *forma canónica observable*

$$\dot{\mathbf{z}} = \underbrace{\begin{bmatrix} \mathbf{0}_{n-1 \times 1} & I_{(n-1) \times (n-1)} \\ \mathbf{0}_{1 \times n} \end{bmatrix}}_{A_0} \mathbf{z} + \underbrace{\begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}}_{\mathbf{a}} y + \underbrace{\begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}}_{\bar{B}} u, \quad (2.31)$$

$$= A_0 \mathbf{z} + \mathbf{a}y + \bar{\mathbf{b}}u,$$

$$y = \underbrace{\begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \end{bmatrix}}_{C_0} \mathbf{z} + Du. \quad (2.32)$$

Es sabido, a partir de la teoría de sistemas lineales, que si $A_0 - \mathbf{a}C_0 = TAT^{-1}$ y $\bar{\mathbf{b}} = TB$ son conocidos, entonces se puede diseñar un observador para \mathbf{z} mediante

$$\dot{\hat{\mathbf{z}}} = A_o \hat{\mathbf{z}} + \mathbf{a}y + \mathbf{b}u + \mathbf{k}(y - C_0 \hat{\mathbf{z}}), \quad (2.33)$$

donde $\mathbf{k} = [k_1 \ \dots \ k_n]^T$ con el polinomio $\lambda^n + (k_n + a_n)\lambda^{n-1} + \dots + (k_2 + a_2)\lambda + (k_1 + a_1)$ Hurwitz por diseño. De este modo, se garantiza que la dinámica del error de observación $\tilde{\mathbf{z}} = \mathbf{z} - \hat{\mathbf{z}}$ tiende a cero exponencialmente.

Como se indica en Marino & Tomei (1996), es natural el buscar una misma estructura de observación para sistemas no lineales como (2.30), donde una transformación $\mathbf{z} = T(\mathbf{x})$ permita obtener el sistema

$$\dot{\mathbf{z}} = A_0 \mathbf{z} + \gamma(y) + \psi(y)u, \quad (2.34)$$

$$y = C_0 \mathbf{z}, \quad (2.35)$$

al partir del cual se diseñe un observador en forma similar al anterior, de modo que su estructura esté dada por

$$\dot{\hat{\mathbf{z}}} = A_0 \hat{\mathbf{z}} + \gamma(y) + \phi(y)u + k(y - C_0 \hat{\mathbf{z}}) \quad (2.36)$$

$$\hat{\mathbf{x}} = T^{-1}(\hat{\mathbf{z}}). \quad (2.37)$$

Nótese que para diseñar este tipo de observadores para sistemas no lineales es necesario determinar la existencia de la transformación, su naturaleza, así como el perfecto conocimiento de $\gamma(y)$ y $\psi(y)$. Es posible que estas funciones dependan a su vez de parámetros desconocidos $\theta = [\theta_\gamma^T \ \theta_\psi^T]^T$ en la forma $\gamma = \gamma(y, \theta_\gamma)$ y $\psi = \psi(y, \theta_\psi)$, los cuales sea necesario calcular. Los observadores que son diseñados para obtener un cálculo de $\hat{\mathbf{x}}$ en la presencia de parámetros desconocidos son llamados *observadores adaptables*.

En este campo ha sido desarrollada una gran cantidad de literatura al respecto, desde los trabajos con sistemas lineales de Kreisselmeier (1977); Lüders & Narendra (1973), el libro de Ioannou & Sun (1996), hasta estructuras no lineales con parametrización lineal (Bastin & Gevers (1988); Marino & Tomei (1996); Zhang (2005), o para clases particulares de sistemas no lineales sin necesidad de transformaciones de linealización Xu & Zhang (2004). Incluso han sido propuestas estructuras para observación estructuras basadas en redes neuronales, como Poznyak *et al.* (2006).

En particular, en Zhang (2009) se propone un observador adaptable para un sistema no lineal linealmente parametrizable, con un cálculo conjunto de los estados $\hat{\mathbf{x}}$ y los parámetros θ con base en un algoritmo desacoplado, el cual será analizado en el Capítulo 4.

Capítulo 3

Identificación mediante redes recurrentes entrada-salida

En este capítulo se presentan dos resultados que proponen una solución al problema de identificación de sistemas no lineales parcialmente desconocidos, basados en redes neurodifusas recurrentes. En primer lugar, se muestra una estructura entrada/salida en tiempo discreto capaz de identificar y modelar el comportamiento dinámico de un sistema a partir de mediciones de entrada y salida del mismo, donde se aprovecha el conocimiento de técnicas lineales en el entrenamiento de los parámetros lineales y no lineales de la red. Se mostrarán diversos algoritmos y modificaciones a la estructura de la red, así como algunas aplicaciones a sistemas no lineales.

En segundo lugar se muestra una estructura en tiempo continuo en forma de variables de estado para el mismo propósito, donde el algoritmo de entrenamiento se basa ahora en la teoría de observadores adaptables. Para este caso, se delimita la identificación y modelado a una clase de sistemas no lineales que puedan ser representados mediante una realimentación de salida. Para realizar una validación de la estructura y algoritmo de entrenamiento se muestra el modelado e identificación de diversos sistemas no lineales.

3.1. Estructura en tiempo discreto Entrada/Salida

Considérese el problema de identificar un sistema SISO¹ cuyo modelo en tiempo discreto, con tiempo de muestreo T_s , puede ser representado en tiempo k mediante el mapeo f_d ,

$$y_{k+1} = f_d(y_k, \dots, y_{k-n+1}, u_k, \dots, u_{k-m+1}) \quad (3.1)$$

donde la señal de entrada es u_k y la de salida es y_k .

El objetivo principal es el encontrar un esquema mediante el cual se logre modelar e identificar al sistema contando con información limitada del mismo, a saber:

Suposición 3.1. *El sistema no lineal (3.1) es tal que:*

- I) *El sistema es BIBO².*
- II) *La función que modela al sistema mediante una ecuación en diferencias f_d es suave con respecto a sus argumentos, y el sistema tiene solución única para cada secuencia de entrada $\{u[0], u[1], \dots, u[k], \dots\}$.*
- III) *Únicamente se tiene acceso a mediciones de y_k y el conocimiento de la señal de entrada u_k ,*
- IV) *No se dispone de la medición de las variables de estado reales de la planta,*
- V) *Es posible que los estados de la red dependan de alguna transformación de los de la planta, ($\mathbf{z} = T(\mathbf{x})$) (este caso se analizará en el Capítulo 4),*
- VI) *Se cuenta con N mediciones de la entrada y salida de un sistema dinámico, tomadas con un tiempo de muestreo constante T .*
- VII) *El orden n del sistema es conocido a priori, así como los m retrasos en la entrada.*

En particular, la suposición 3.1.III implica que únicamente es posible medir a la salida y entrada del sistema y no es posible medir o conocer los estados del mismo. Es de notar que los estados de la red no tienen por qué ser los estados del sistema físico, si bien ambos representan a la dinámica del sistema.

¹Sistema Una-Entrada-Una-Salida, por sus siglas en inglés SISO: *Single-Input-Single-Output*.

²Entrada-Acotada-Salida-Acotada, por sus siglas en inglés BIBO: *Bounded Input, Bounded Output*.

En consecuencia, al no disponerse de los estados, pero sí de n retrasos de la señal de salida y k de la de entrada, los primeros pueden ser tomados como los *estados* del sistema disponibles para medición. A continuación se propone una estructura basada en redes neurodifusas recurrentes capaz de identificar un sistema bajo estas suposiciones.

3.1.1. Estructura entrada-salida

El sistema (3.1) es un sistema que puede ser visto como un mapeo dinámico entrada/salida $y_k = G(u_k)$. Al suponer que no contamos con información específica de la naturaleza de dicho mapeo, el objetivo particular es encontrar una estructura no lineal represente al mapeo dinámico del sistema anterior atendiendo a las suposiciones. Mediante una estructura general $\hat{y}_h = \hat{G}(u_k)$ que genere una respuesta similar a la del sistema original.

Haciendo uso del teorema de Aproximación Universal (Wang, 1994) mencionado en la sección 2.3.1, se propone una estructura basada en redes neurodifusas recurrentes en tiempo discreto, donde cada regla se define por n_R reglas en la forma:

R_i : Si \hat{y}_k es A_i entonces

$$\begin{aligned} \hat{y}_{k+1} &= \underbrace{\begin{bmatrix} c_{i1} & \dots & c_{in} \end{bmatrix}}_{\mathbf{c}_i^T} \begin{bmatrix} \hat{y}_k \\ \vdots \\ \hat{y}_{k-n+1} \end{bmatrix} + \underbrace{\begin{bmatrix} h_{i1} & \dots & h_{im} \end{bmatrix}}_{\mathbf{h}_i^T} \begin{bmatrix} u_k \\ \vdots \\ u_{k-m+1} \end{bmatrix}, \\ &= \mathbf{c}_i^T \begin{bmatrix} \hat{y}_k \\ \vdots \\ \hat{y}_{k-n+1} \end{bmatrix} + \mathbf{h}_i^T \begin{bmatrix} u_k \\ \vdots \\ u_{k-m+1} \end{bmatrix}, \end{aligned} \quad (3.2)$$

donde \hat{y} es la estimación de la salida del sistema, $A_i(x) = \exp(-\sigma_i^2(x - \zeta_i)^2)$ es la función de membresía del conjunto difuso A_i . En lo sucesivo el proceso de desfusión se define mediante el promedio ponderado.

Con las consideraciones anteriores, el sistema se reescribe con base en una red neurodifusa parcialmente recurrente en la forma

$$\begin{aligned} \hat{y}_{k+1} &= (\varphi_k^1 \mathbf{c}_1^T + \dots + \varphi_k^n \mathbf{c}_{n_R}^T) \mathbf{z}_k + (\varphi_k^1 \mathbf{h}_1^T + \dots + \varphi_k^n \mathbf{h}_{n_R}^T) \xi_k, \\ &= (\varphi_k^T \otimes \mathbf{z}_k) \text{vect}(\mathbf{C}) + (\varphi_k^T \otimes \xi_k) \text{vect}(\mathbf{H}), \end{aligned} \quad (3.3)$$

donde \otimes es el producto matricial de Kronecker, y las demás señales y funciones se definen como

$$\mathbf{z}_k = [\hat{y}_k \ \hat{y}_{k-1} \ \dots \ \hat{y}_{k-n+1}]^T, \quad (3.4)$$

$$\xi_k = A_0 \xi_{k-1} + B_0 u_k, \quad (3.5)$$

$$R_k^i = e^{-\sigma_i^2 (\hat{y}_k - \varsigma_i)^2}, \quad (3.6)$$

$$\varphi_k^i = \frac{R_k^i}{\sum_j R_k^j}, \quad (3.7)$$

$$\mathbf{C} = [\mathbf{c}_1 \ \dots \ \mathbf{c}_n]^T, \quad (3.8)$$

$$\mathbf{H} = [\mathbf{h}_1 \ \dots \ \mathbf{h}_m]^T, \quad (3.9)$$

$$A_0 = \begin{bmatrix} 0_{1 \times m} & \\ I_{(m-1) \times (n-1)} & 0_{(m-1) \times 1} \end{bmatrix}, \quad (3.10)$$

$$B_0 = \begin{bmatrix} 1 \\ 0_{(m-1) \times (m-1)} \end{bmatrix}, \quad (3.11)$$

donde los parámetros a determinar son

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_n^T \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{h}_1^T \\ \vdots \\ \mathbf{h}_m^T \end{bmatrix}, \quad \sigma = \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_n \end{bmatrix}, \quad \varsigma = \begin{bmatrix} \varsigma_1 \\ \vdots \\ \varsigma_n \end{bmatrix},$$

con las funciones

$$R_k^i = A_i(y_k) \quad (\text{activación de la } i\text{-ésima regla}),$$

$$\varphi_k^i = \frac{\mu_{R_k^i}}{\sum_j \mu_{R_k^j}} \quad (\text{membresía normalizada}),$$

Nótese que las ecuaciones (3.5), (3.10), (3.11), representan a un sistema en variables de estado cuya función principal es obtener los m retrasos de la entrada u_k en forma del vector $\xi_k = [u_k \ \dots \ u_{k-m+1}]^T$.

De esta forma, la aproximación al sistema real H , cuya salida es y_{k+1} se expresa como

$$\begin{aligned} y_{k+1} &= f_1(\mathbf{C}, \sigma, \varsigma, \mathbf{z}_k) + f_2(\mathbf{H}, \sigma, \varsigma, \xi_k) + \bar{\zeta}_k \\ &= f(\mathbf{C}, \mathbf{H}, \sigma, \varsigma, \mathbf{z}_k, \xi_k) + \bar{\zeta}_k \end{aligned} \quad (3.12)$$

Donde el error de aproximación $\bar{\zeta}_k$ es mínimo ($\bar{\zeta}_k = \min_k \{\bar{\zeta}_k\}$) si se encuentran valores de \mathbf{C}^T , \mathbf{H}^T , σ y ς tales que

$$\begin{aligned}
y_{k+1} &= f_1(\mathbf{C}^*, \sigma^*, \varsigma^*, \mathbf{z}_k) + f_2(\mathbf{H}^*, \sigma^*, \varsigma^*, \xi_k) + \zeta_k, \\
&= f(\mathbf{C}^*, \mathbf{H}^*, \sigma^*, \varsigma^*, \mathbf{z}_k, \xi_k) + \zeta_k.
\end{aligned} \tag{3.13}$$

3.1.1.1. Linealización de los parámetros antecedentes

Al observar la estructura (3.3) se observa que los parámetros consecuentes (esto es, las matrices \mathbf{C} y \mathbf{H}) aparecen en forma lineal con respecto a la estructura, si no se desea además identificar a los parámetros antecedentes σ y ς . En este caso, en la literatura existe una gran cantidad de algoritmos diseñados que pueden ser de utilidad (Nelles, 2001): desde algoritmos basados en el gradiente, pasando por mínimos cuadrados, filtro de Kalman, etc. Sin embargo, la modificación de los parámetros antecedentes (no lineales, en este caso) es deseable si se desea mejorar el desempeño de identificación de la red, ya que ello implica sintonizar la localización de los conjuntos difusos.

En el capítulo anterior se comentaron algunos algoritmos existentes para modificar los parámetros no lineales del sistema. Sin embargo, una de sus principales desventajas es la falta de un análisis de estabilidad, lo cual compromete la calidad del entrenamiento final.

Sin embargo, siguiendo un procedimiento similar al desarrollado por Yu & Li (2004), es posible encontrar una *linealización* de la red (3.3) en torno a valores *subóptimos* de los parámetros (aquellos que no minimizan el error $y_k - \hat{y}_k$, pero se encuentran “cerca” del valor ideal). Visto de esta forma, dado el vector de parámetros

$$\theta = [\text{vect}(\mathbf{C})^T \quad \text{vect}(\mathbf{H})^T \quad \sigma^T \quad \varsigma^T] \tag{3.14}$$

y la función de estos $f = f(\theta^*)$, se emplea una expansión en series de Taylor en torno a θ en la forma

$$f = f(\theta^*) = f(\theta) + (\theta^* - \theta)^T \left. \frac{\partial f}{\partial \theta} \right|_{\theta^* = \theta} + R(\theta, \theta^*), \tag{3.15}$$

donde $R(\theta, \theta^*)$ representa a los elementos de orden mayor dentro de la expansión en la serie de Taylor. Dado que se cuenta con el conocimiento directo de $y_k \dots, y_{k-n+1}$, se considera que $\mathbf{z}_k = [y_k \dots y_{k-n+1}]^T$ durante el entrenamiento.

Al aplicar el mismo desarrollo en la ecuación (3.13), y dado que la función f es lineal con respecto a los parámetros consecuentes \mathbf{C} , \mathbf{H} , finalmente se obtiene

$$\begin{aligned}
 f(\mathbf{C}^*, \mathbf{H}^*, \sigma^*, \varsigma^*, \mathbf{z}_k, \xi_k) &= f(\mathbf{C}, \mathbf{H}, \sigma, \varsigma, \mathbf{z}_k, \xi_k) \\
 &+ f(\mathbf{C}^* - \mathbf{C}, \mathbf{H}^* - \mathbf{H}, \sigma, \varsigma, \mathbf{z}_k, \xi_k) \\
 &+ (\sigma^* - \sigma)^T \left. \frac{\partial f}{\partial \sigma^*} \right|_{\sigma, \varsigma, \mathbf{c}} + (\varsigma^* - \varsigma)^T \left. \frac{\partial f}{\partial \varsigma^*} \right|_{\sigma, \varsigma, \mathbf{c}} + R(3.16)
 \end{aligned}$$

Como $y_{k+1} = f(\mathbf{C}^*, \mathbf{H}^*, \sigma^*, \varsigma^*, \mathbf{z}_k, \xi_k) + \zeta_k$ y $\hat{y}_{k+1} = f(\mathbf{C}, \mathbf{H}, \sigma, \varsigma, \mathbf{z}_k, \xi_k)$, al restar ambos elementos se obtiene

$$\begin{aligned}
 \tilde{y}_{k+1} &\triangleq y_{k+1} - \hat{y}_{k+1} = \tilde{\sigma}^T \mathbf{g}_k + \tilde{\varsigma} \mathbf{h}_k \\
 &+ \varphi^T (\mathbf{C}^* - \mathbf{C}) \mathbf{z}_k + \varphi^T (\mathbf{H}^* - \mathbf{H}) \xi_k + \vartheta_k
 \end{aligned} \tag{3.17}$$

donde $\vartheta_k = R - \zeta_k$, $\mathbf{g}_k = \left. \frac{\partial f}{\partial \sigma^*} \right|_{\sigma, \varsigma}$, $\mathbf{h}_k = \left. \frac{\partial f}{\partial \varsigma^*} \right|_{\sigma, \varsigma}$.

Al reescribir las ecuaciones, obtenemos el modelo del error de aproximación linealizado con respecto a los parámetros

$$\tilde{y}_{k+1} = \tilde{\theta}^T \mathbf{G}_k + \vartheta_k = \tilde{\theta}_k^T \mathbf{G}_k + \vartheta_k, \tag{3.18}$$

donde

$$\tilde{\theta} \triangleq \theta^* - \theta, \quad \theta = \begin{bmatrix} \text{vect}(\mathbf{C}) \\ \text{vect}(\mathbf{H}) \\ \sigma \\ \varsigma \end{bmatrix}, \quad \mathbf{G}_k = \begin{bmatrix} \varphi \otimes \mathbf{z}_k \\ \varphi \otimes \xi_k \\ \mathbf{g}_k \\ \mathbf{h}_k \end{bmatrix}. \tag{3.19}$$

Las ecuaciones (3.20-3.22) expresan los cálculos de las derivadas parciales con respecto a σ y ς :

$$\left. \frac{\partial f_1}{\partial \sigma^*} \right|_{\sigma^*=\sigma} = \frac{\partial \varphi^T}{\partial \sigma} \mathbf{Cz}(k) \quad (3.20)$$

$$\begin{aligned} \frac{\partial \varphi^T}{\partial \sigma} &= 2 \begin{bmatrix} \frac{\sigma_1(y_k - \varsigma_1)^2}{(\sum_j \mu_j)} \mu_1 \\ \vdots \\ \frac{\sigma_n(y_k - \varsigma_n)^2}{(\sum_j \mu_j)} \mu_n \end{bmatrix} \otimes [\mu_1 \ \dots \ \mu_n] - \\ &2 \text{diag} \left(\left[\frac{\mu_1}{(\sum_l \mu_l)} (\sigma_1 (y_k - \varsigma_1)^2) \dots \frac{\mu_n}{(\sum_l \mu_l)} (\sigma_n (y_k - \varsigma_n)^2) \right] \right) \end{aligned} \quad (3.21)$$

$$\left. \frac{\partial f}{\partial \varsigma^*} \right|_{\varsigma^*=\varsigma} = \frac{\partial \varphi^T}{\partial \sigma} \mathbf{Cz}(k) \quad (3.22)$$

$$\begin{aligned} \frac{\partial \varphi^T}{\partial \varsigma} &= -2 \begin{bmatrix} \frac{\sigma_1^2(y_k - \varsigma_1)}{(\sum_j \mu_j)} \mu_1 \\ \vdots \\ \frac{\sigma_n^2(y_k - \varsigma_n)}{(\sum_j \mu_j)} \mu_n \end{bmatrix} \otimes [\mu_1 \ \dots \ \mu_n] + \\ &2 \text{diag} \left(\left[\frac{\mu_1}{(\sum_l \mu_l)} (\sigma_1^2 (y_k - \varsigma_1)) \dots \frac{\mu_n}{(\sum_l \mu_l)} (\sigma_n^2 (y_k - \varsigma_n)) \right] \right) \end{aligned} \quad (3.23)$$

3.1.1.2. Algoritmo de inicialización de parámetros

Al realizar la linealización es conveniente inicializar a los parámetros cerca de un mínimo local, lo cual tiene como objetivo el lograr que la diferencia entre $\theta^* - \theta$ sea lo más reducida posible para reducir la magnitud de los elementos de mayor orden R en la expansión en series de Taylor (3.16).

El algoritmo empleado es el mismo empleado que en Gonzalez Olvera & Tang (2007), el cual se basa en los siguientes pasos:

1. Entrenar un sistema difuso en forma estática, cuya salida sea la dada por las salidas conocidas en y_k , sus n retrasos, y la entrada u_k y m retrasos. De preferencia, emplear el método ANFIS (Jang, 1993).
2. El sistema entrenado tendrá $n + m$ conjuntos difusos en cada entrada (relacionados con las señales de entrada y salida, respectivamente), así como n_R reglas.
3. Inicializar la red con n_R reglas y n "estados".

4. Extraer los parámetros relacionados con el conjunto difuso correspondiente a y_{k-1} e igualarlos con aquellos de la red IOCRenn
5. Dado que ANFIS genera un sistema difuso con funciones lineales en cada regla, extraer los parámetros consecuentes e igualarlos con aquellos de \mathbf{C} y \mathbf{H} , siguiendo el orden lógico. Esto es, el algoritmo ANFIS genera reglas en la forma

$$R_i: \text{ Si } y_k \text{ es } A_{y,1,i} \text{ y } \dots \text{ y } y_{k-n} \text{ es } A_{y,n-1,i} \text{ y } u_k \text{ es } B_{u,1,i} \text{ y } \dots \text{ y } u_{k-m} \text{ es } A_{u,m-1,i} \\ \text{ entonces} \\ \hat{y}_{k+1} = \alpha$$

3.1.1.3. Análisis de la estabilidad de la red

El análisis de estabilidad de la red puede ser llevado a cabo al transformarlo en una representación en variables de estado. Para tal efecto, considerando $u_k \equiv 0$, se obtiene

$$\mathbf{z}_{k+1} = \sum_i \varphi_k^i \begin{bmatrix} \mathbf{c}_i^T \\ I_{(n-1) \times (n-1)} & 0_{(n-1) \times 1} \end{bmatrix} \mathbf{z}_k. \quad (3.24)$$

De esta forma, se emplea el análisis de estabilidad mencionado en Kosko (1992):

Teorema 3.1. *Sea un sistema difuso con reglas lineales en variables de estado con D_i como la matriz del sistema en cada regla. Es condición suficiente para que el sistema sea asintóticamente estable en el sentido de Lyapunov que $\exists P = P^T > 0$ tal el conjunto de desigualdades*

$$D_i P D_i - P < 0 \quad (3.25)$$

se cumpla $\forall i = 1, \dots, n_R$

La prueba de este teorema se puede encontrar en Kosko (1992).

3.1.2. Algoritmos de entrenamiento

3.1.2.1. Entrenamiento por mínimos cuadrados

Al estar el sistema inicializado con parámetros *cercanos* a un mínimo local, se hacen las siguientes suposiciones:

Suposición 3.2. *La red neurodifusa no lineal es inicializada cerca de un mínimo local tal que la diferencia entre $\theta^* - \theta$ sea lo suficientemente pequeña como para considerar a los residuos R pequeños.*

De esta forma, el sistema linealizado (3.18) optimizado puede ser reescrito como

$$\tilde{\theta}_{k+1} = \tilde{\theta}_k \quad (3.26)$$

$$\tilde{y}_k = \mathbf{G}_k \tilde{\theta}_k^* + \vartheta_k \quad (3.27)$$

Dado que no se cuenta con la medición directa de $\tilde{\theta}$, se emplea el método de mínimos cuadrados (Johansson, 1993) para determinar los parámetros, mediante la ley de adaptación

$$\hat{\theta}_k = \hat{\theta}_{k-1} + K_k (y_k - \mathbf{G}_k \hat{\theta}_{k-1}) \quad (3.28)$$

$$K_k = \frac{P_{k-1}}{R_2 + \mathbf{G}_k P_{k-1} \mathbf{G}_k^T} \quad (3.29)$$

$$P_k = P_{k-1} - \frac{P_{k-1} \mathbf{G}_k^T \mathbf{G}_k P_{k-1}}{1 + \mathbf{G}_k P_{k-1} \mathbf{G}_k} \quad (3.30)$$

De esta forma, se logra un entrenamiento estable, ya que se puede probar (Johansson, 1993) que, empleando la función cuadrática de Lyapunov

$$V_k = \frac{1}{2} \tilde{\theta}_k^T P_k^{-1} \tilde{\theta}_k \quad (3.31)$$

se obtiene que el cambio de esta función es

$$\Delta V_k = V_{k+1} - V_k = \frac{1}{2} \|\vartheta_k\|_2^2 - \frac{1}{2} \frac{\mathbf{G}_k^T P_{k-1} \mathbf{G}_k}{1 + \varphi_k^T P_{k-1} \varphi_k} \tilde{y}_k^2 \quad (3.32)$$

Por lo tanto la región de convergencia del error de identificación es función de la magnitud $\|\vartheta_k\| = \|\zeta_k - R_k\|$.

3.1.2.2. Entrenamiento por Filtro de Kalman

Al considerar de nueva cuenta al sistema nominal

$$\theta_{k+1}^* = \theta_k^* \quad (3.33)$$

$$y_k = \mathbf{G}_k \theta_k^* + \vartheta_k \quad (3.34)$$

con $\vartheta_k = \zeta_k - R_k$ vista ahora como una perturbación. Dado que no hay un conocimiento directo de θ^* , y el efecto de las señales ζ_k y R_k debe ser tomado en cuenta, se puede

considerar ahora un algoritmo de entrenamiento basado en el filtro de Kalman. Las ecuaciones que definen al filtro de Kalman para este caso están dadas por

$$\hat{\theta}_k = \hat{\theta}_{k-1} + K_k(y_k - \mathbf{G}_k \hat{\theta}_{k-1}) \quad (3.35)$$

$$K_k = \frac{P_{k-1} \mathbf{G}_k^T}{R_2 + \mathbf{G}_k P_{k-1} \mathbf{G}_k^T} \quad (3.36)$$

$$P_k = P_{k-1} - \frac{P_{k-1} \mathbf{G}_k^T \mathbf{G}_k P_{k-1}}{R_2 + \mathbf{G}_k P_{k-1} \mathbf{G}_k^T} + R_1 \quad (3.37)$$

con K_k la matriz de ganancia, P_k la *matriz de covarianza*, R_2 el *factor de olvido*, y $R_1 = R_1^T$ la matriz que de covarianza del ruido del proceso ϑ_k , el cual se supone es ruido blanco gaussiano con media cero.

Teorema 3.2. *Si se emplea al algoritmo de mínimos cuadrados para entrenar a un sistema con perturbación ϑ_k , entonces el error de parámetros $\tilde{\theta}_k$ permanece acotado mientras*

$$e_k^2 > (1 + \mathbf{G}_k P_{k-1} \mathbf{G}_k) \vartheta_k^2 \quad (3.38)$$

Demostración. En primer lugar, se define la función cuadrática con $P_k = P_k^T > 0$

$$V_k = \frac{1}{2} \tilde{\theta}_k^T P_k^{-1} \tilde{\theta}_k. \quad (3.39)$$

Al calcular el cambio $\Delta V_k = V_k - V_{k-1}$

$$\Delta V_k = \frac{1}{2} \tilde{\theta}_k^T P_k^{-1} \tilde{\theta}_k - \frac{1}{2} \tilde{\theta}_{k-1}^T P_{k-1}^{-1} \tilde{\theta}_{k-1} \quad (3.40)$$

$$= \frac{1}{2} \left(\tilde{\theta}_{k-1} + P_k \mathbf{G}_k^T e_k \right)^T P_k^{-1} \left(\tilde{\theta}_{k-1} + P_k \mathbf{G}_k^T e_k \right) - \frac{1}{2} \tilde{\theta}_{k-1}^T P_{k-1}^{-1} \tilde{\theta}_{k-1}. \quad (3.41)$$

Tomando en cuenta que $P_k^{-1} = \mathbf{G}_k \mathbf{G}_k^T + P_{k-1}^{-1}$, $P_k = P_k - \frac{P_{k-1} \mathbf{G}_k^T \mathbf{G}_k P_{k-1}}{1 + \mathbf{G}_k P_{k-1} \mathbf{G}_k^T}$,

$$\Delta V_k = \frac{1}{2} \vartheta_k^2 - \frac{1}{2} \frac{e_k^2}{1 + \mathbf{G}_k P_{k-1} \mathbf{G}_k^T} \quad (3.42)$$

De lo cual se observa que mientras $\|e_k\|_2^2 > (1 + \mathbf{G}_k P_{k-1} \mathbf{G}_k) \|\vartheta_k\|_2^2$, el error de parámetros decrece, y por lo tanto todas las señales permanen acotadas. \square

3.1.2.3. Entrenamiento basado en estabilidad en el sentido de Lyapunov

En general, los algoritmos basados en el filtro de Kalman o en mínimos cuadrados recursivos son un tanto más demandantes en términos computacionales que los métodos

basados en el gradiente. En primer lugar, definamos el error cuadrático del error de identificación como

$$V_k \triangleq \frac{1}{2} \tilde{\theta}_k^T \tilde{\theta}_k. \quad (3.43)$$

Al calcular el cambio de tiempo k a $k+1$, obtenemos

$$\Delta V_k = \frac{1}{2} (V_{k+1} - V_k) = \frac{1}{2} \left(\tilde{\theta}_{k+1}^T \tilde{\theta}_{k+1} - \tilde{\theta}_k^T \tilde{\theta}_k \right), \quad (3.44)$$

que puede ser reescrito como

$$\Delta V_k = \frac{1}{2} \left(\tilde{\theta}_{k+1} - \tilde{\theta}_k \right)^T \left(\tilde{\theta}_{k+1} + \tilde{\theta}_k \right) \quad (3.45)$$

$$= \frac{1}{2} \Delta \tilde{\theta}_k^T \left(\tilde{\theta}_{k+1} - \tilde{\theta}_k + \tilde{\theta}_k + \tilde{\theta}_k \right) \quad (3.46)$$

$$= \frac{1}{2} \Delta \tilde{\theta}_k^T \Delta \tilde{\theta}_k + \Delta \tilde{\theta}_k^T \tilde{\theta}_k = \frac{1}{2} \left\| \Delta \tilde{\theta}_k \right\|_2^2 + \Delta \tilde{\theta}_k^T \tilde{\theta}_k \quad (3.47)$$

Como $\Delta \tilde{\theta}_k = \left(\hat{\theta}_{k+1} - \theta^* \right) - \left(\hat{\theta}_k - \theta^* \right) = \Delta \hat{\theta}_k$, la función de actualización de variables puede escribirse como

$$\Delta \hat{\theta}_k = -\mathbf{G}_k \frac{\alpha_k}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} e_k. \quad (3.48)$$

donde $\alpha_k \in \mathfrak{R}^+$ es una ganancia positiva posiblemente variante en tiempo y $\varepsilon > 0$. De esta forma, al ser el error de modelado $e_k = y_k - \hat{y}_k = G_k \theta^* + \vartheta_k - G_k \hat{\theta} = G_k \tilde{\theta}_k + \vartheta_k$,

$$\Delta V_k = \frac{1}{2} \left(\frac{\alpha_k^2 e_k^2}{\left(2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon \right)^2} \mathbf{G}_k^T \mathbf{G}_k \right) - \frac{\alpha_k e_{k-1}}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} \mathbf{G}_k^T \tilde{\theta}_k \quad (3.49)$$

$$= \frac{1}{2} \left(\frac{\alpha_k^2 e_k^2}{\left(2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon \right)^2} \mathbf{G}_k^T \mathbf{G}_k \right) - \frac{\alpha_k}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} e_k^2 + \frac{\alpha_k e_k}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} \vartheta_k \quad (3.50)$$

$$= \frac{1}{2} \frac{\alpha_k e_k^2}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} \left(\frac{\alpha_k \left\| \mathbf{G}_k \right\|_2^2}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} - 1 \right) + \frac{\alpha_k e_k}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} \vartheta_k. \quad (3.51)$$

Por lo tanto, mientras que $\varepsilon > 0$, la ganancia sea

$$\alpha_k \leq \left(2 + \frac{\varepsilon}{\left\| \mathbf{G}_k \right\|_2^2} \right), \quad (3.52)$$

y la perturbación cumpla con

$$\left\| \vartheta_k \right\| \leq \frac{\left\| e_k \right\|}{2} \left\| 1 - \frac{\alpha_k \left\| G_k \right\|^2}{2 \left\| \mathbf{G}_k \right\|_2^2 + \varepsilon} \right\| \quad (3.53)$$

se garantiza la estabilidad del entrenamiento y la estabilidad práctica del sistema (cota final).

3.1.2.4. Ejemplos

Para analizar la capacidad de identificación de la red se prueba con un sistema lineal y uno no lineal. Al estar la red compuesta por una consecución de sistemas lineales, se espera que el error de modelado tienda prácticamente a cero y con ello se evalúe la efectividad del algoritmo de entrenamiento. En el segundo caso, al existir teóricamente un error de modelado ε se espera que la red logre modificar sus parámetros de modo tal que el error de identificación sea lo más pequeño posible.

En ambos casos se emplea el algoritmo de inicialización descrito en la sección 3.1.1.2 para situar a los parámetros cerca de algún mínimo local y evitar en lo posible la interferencia de R en el algoritmo de identificación. En ambos casos, los sistemas fueron entrenados y probados empleando una señal pseudoaleatoria cuyos valores varían en $[-1,02, 1,02]$. En el primer caso se emplearon 4,000 datos, mientras que en el segundo fueron empleados 8,000.

Identificación de un sistema lineal

El sistema a identificar está dado por la función de transferencia

$$H(z) = \frac{(z^{-1} - 0,5)(z^{-1} + 0,5)}{(z^{-1} + 0,9)(z^{-1} - 0,8 \pm 0,2i)} \quad (3.54)$$

En este caso, se entrena una red con $m = 2$ y $n = 3$, con 4 reglas, para analizar la efectividad de identificación de la red. El algoritmo de entrenamiento elegido es el de mínimos cuadrados, tomando una linealización para la identificación de los parámetros no lineales. Los resultados obtenidos tras la simulación de ambos sistemas corriendo en paralelo se muestran en la Figura 3.1. Por su parte, la evolución de los parámetros internos σ y ζ , así como el valor de las raíces de cada polinomio asociado a la regla de cada sistema lineal se muestra en la Fig. 3.2.

Identificación de un sistema no lineal

Para probar la capacidad de identificación de la red se emplea el sistema propuesto por (Narendra & Parthasarathy, 1990), típicamente empleado como sistema de

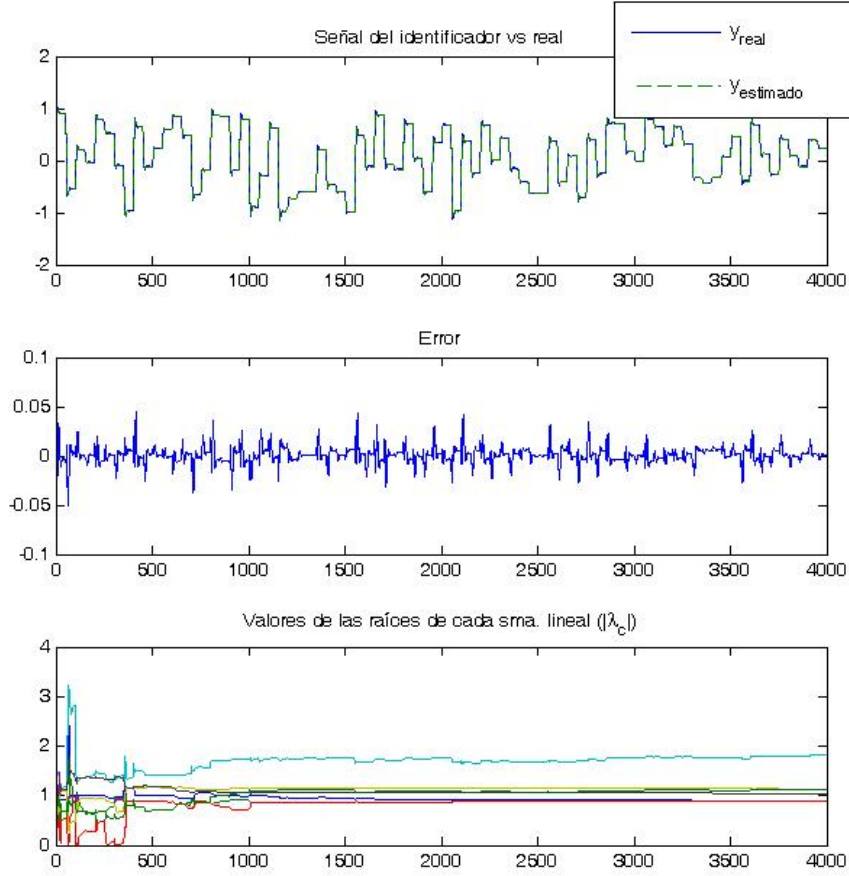


Figura 3.1: Identificación de un sistema lineal

comparación, que se define por la ecuación en diferencias

$$\begin{aligned}
 y_k &= f_N(y_{k-1}, y_{k-2}, y_{k-3}, u_k, u_{k-1}), \\
 f(x_1, x_2, x_3, x_4, x_5) &= \frac{x_1 x_2 x_3 x_5 (x_3 - 1) + x_4}{1 + x_3^2 + x_2^2}.
 \end{aligned}
 \tag{3.55}$$

En el algoritmo se emplearon igualmente $n = 3$ y $m = 2$ bajo el método de mínimos cuadrados y linealización de los parámetros no lineales. Los resultados obtenidos se muestran en la Figura 3.3. Por su parte, en la Fig. 3.4 se muestra la evolución de los parámetros internos σ y ζ , así como el valor de las raíces de cada polinomio asociado a la regla de cada sistema lineal. El valor RMS obtenido para esta identificación es de 0,1177

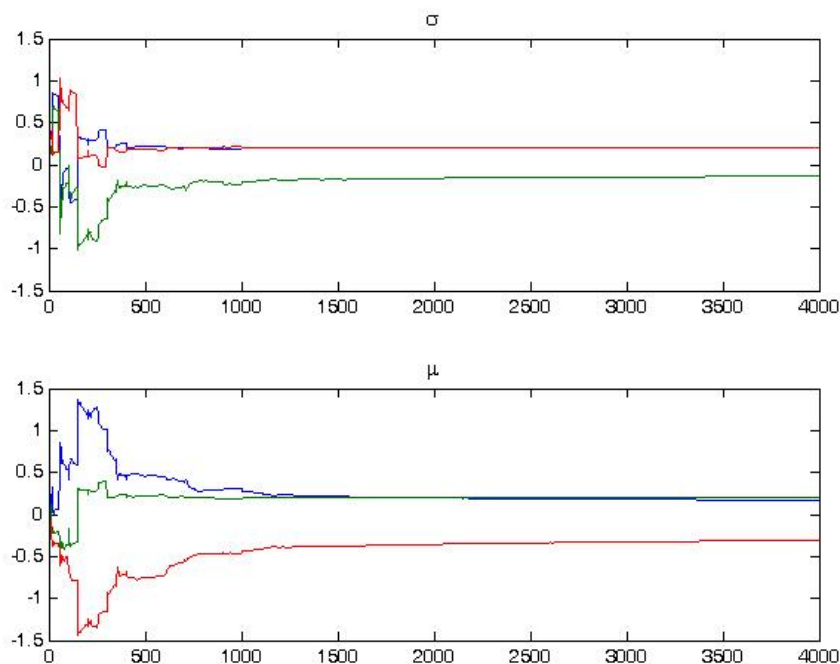


Figura 3.2: Identificación de un sistema lineal: Evolución de los parámetros

y se muestra en la Fig. 3.3, donde además se aprecia que existen errores relativamente grandes al existir cambios drásticos en la señal de entrada. Los valores obtenidos por otros autores son: 0.0418 en el trabajo de Gonzalez Olvera & Tang (2007), y de 0.0084 en los resultados reportados por Juang (2002).

De los ejemplos anteriores se aprecia que la estructura, basada en redes recurrentes en tiempo discreto, es capaz de identificar el desempeño entrada/salida de sistemas no lineales. Una de las ventajas mostradas por la estructura radica en que ésta permite el empleo de teorías ya desarrolladas para la estabilidad de sistemas difusos representados en variables de estado. A pesar de que la aproximación tiene resultados con menor precisión a comparación de los obtenidos por otros autores, el acceso a los parámetros es más sencillo y se cuenta con un análisis de estabilidad del sistema. La estructura en forma de red neurodifusa permite interpretar los de los parámetros obtenidos, ya que los parámetros σ y ζ indican la zona del espacio donde actúan, mientras que \mathbf{C} y \mathbf{H} modelan sistemas lineales locales.

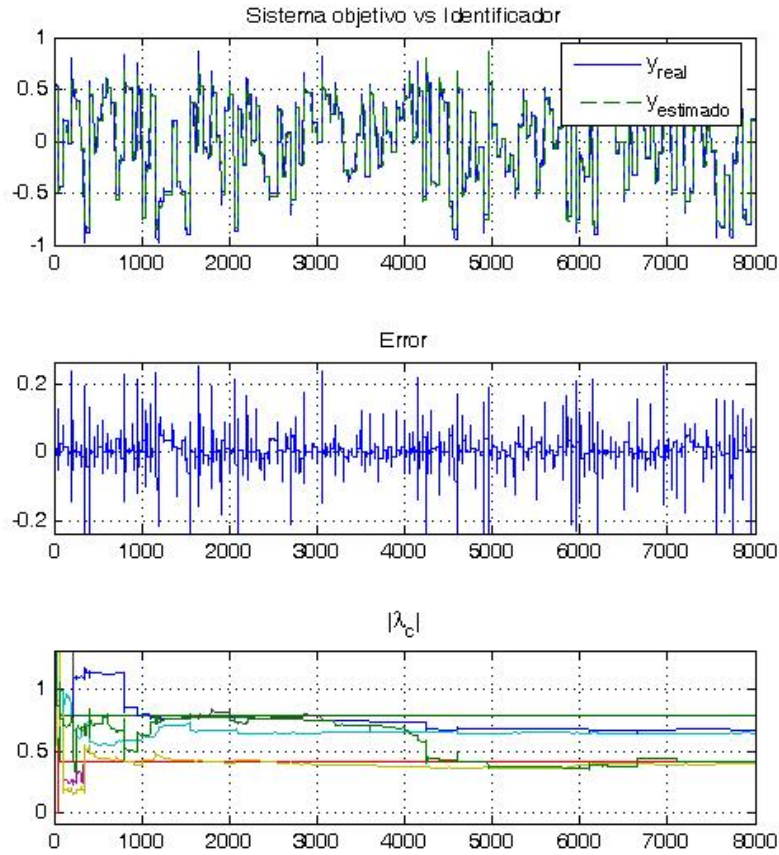


Figura 3.3: Identificación de un sistema no lineal

Por otra parte, dada la propia naturaleza del algoritmo empleado, se logra una prueba de convergencia en los parámetros y se comprueba que el método es robusto ante perturbaciones estocásticas. Sin embargo, no existe garantía alguna de que los conjuntos difusos obtenidos se sitúen dentro de la zona de operación de la red, como ocurre con este ejemplo. Esto se puede explicar por el hecho de que la linealización para algún o algunos ς_i puede ser válida más allá de los límites de las señales, como ocurre con los conjuntos difusos obtenidos tras el entrenamiento que se muestran en la Fig. 3.5. Otra explicación probable es la probable no unicidad de soluciones, donde alguna de ellas puede provocar este efecto. Es por ello que en la siguiente sección se

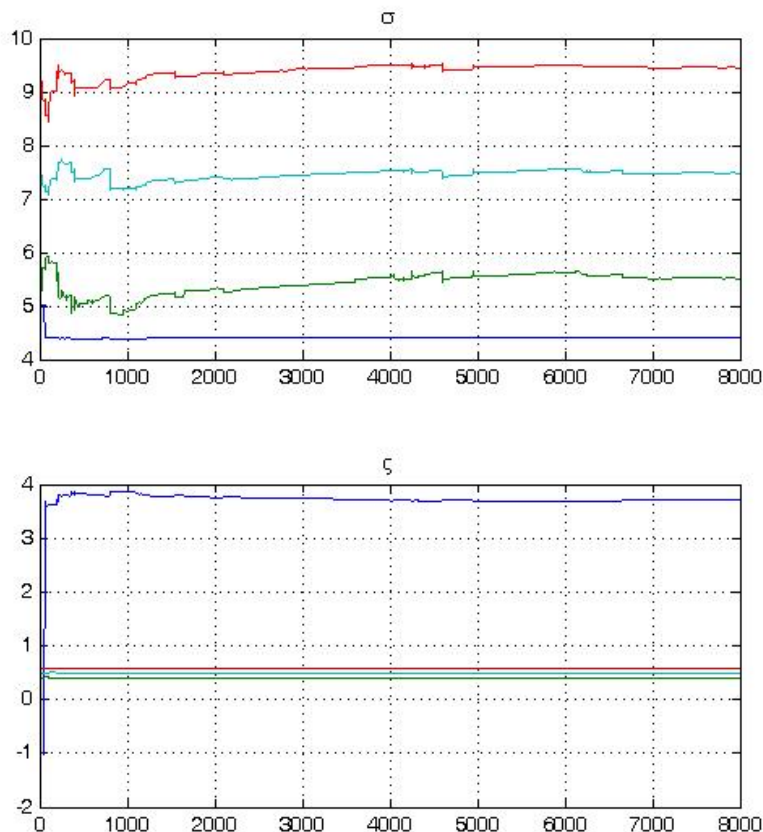


Figura 3.4: Identificación de un sistema no lineal: Evolución de los parámetros

abordará este problema y una propuesta de solución.

3.1.3. Estructura Entrada-Salida con restricciones en los parámetros

La estructura descrita en la sección 3.1.1 considera el problema de identificación de un sistema y prueba la estabilidad durante el entrenamiento de la parte consecuente (o lineal) y antecedente (no lineal) de la red neurodifusa. Sin embargo al no haber restricción respecto a los valores que pueden alcanzar los parámetros, es posible que algunos de ellos tiendan a valores demasiado grandes, lo que es particularmente importante en el caso de aquellos de la parte antecedente. Esto se señala en la Fig. 3.6.a, donde

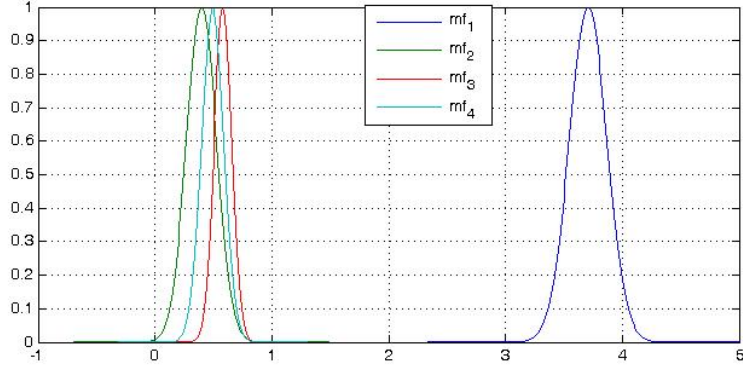


Figura 3.5: Identificación de un sistema no lineal: Conjuntos difusos resultantes tras el entrenamiento

es posible que durante el entrenamiento uno de los conjuntos difusos resulte definido fuera del intervalo donde se desempeña la señal de salida $y(t)$. De esta forma, es posible que la regla no resulte activada en momento alguno del entrenamiento; o que se dejen zonas sin cubrir y que por lo tanto existan singularidades en la definición del promedio ponderado.

En esta sección se explorará un método para evitar este tipo de situaciones, el cual no se referirá al algoritmo de entrenamiento sino a la misma estructura. El objetivo será el lograr que los conjuntos de membresía no salgan de una zona dada de operación, como se aprecia en la Fig. 3.6.b.

El diseño se centrará en evitar estos problemas mediante un diseño de una red recurrente entrada/salida tal que permita el entrenamiento tanto de los parámetros antecedentes como de los consecuentes, manteniendo la relativa simplicidad del esquema anteriormente propuesto. De esta forma, la estructura responderá a una modificación de la estructura mostrada en la Sección 3.1 y reportada en Gonzalez-Olvera & Tang (2008a).

La estructura considerada cuenta prácticamente la misma estructura que la vista en la sección 3.1.1 y consta de una red neurodifusa con n_R reglas, donde la i -ésima regla ($i = 1, 2, \dots, n_R$) se define mediante

R_i : Si \hat{y}_k es A_i entonces

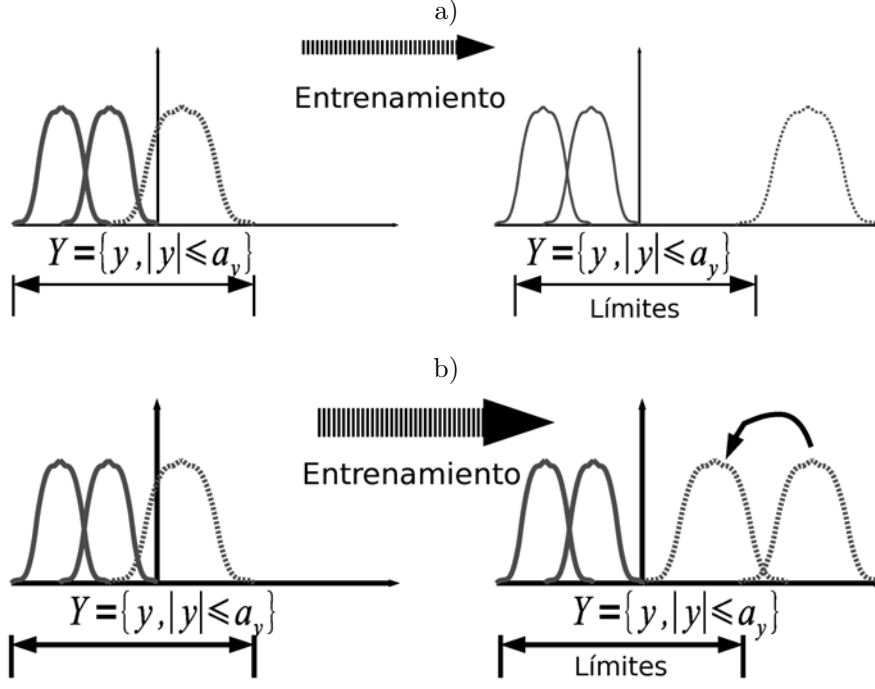


Figura 3.6: Restricción en los parámetros

$$\begin{aligned}
 \hat{y}_{k+1} &= [c_{i1} \dots c_{in}] \begin{bmatrix} \hat{y}_k \\ \vdots \\ \hat{y}_{k-n+1} \end{bmatrix} + [h_{i1} \dots h_{im}] \begin{bmatrix} u_k \\ \vdots \\ u_{k-m+1} \end{bmatrix} \\
 &= \mathbf{c}_i^T \begin{bmatrix} \hat{y}_k \\ \vdots \\ \hat{y}_{k-n+1} \end{bmatrix} + \mathbf{h}_i^T \begin{bmatrix} u_k \\ \vdots \\ u_{k-m+1} \end{bmatrix}
 \end{aligned} \tag{3.56}$$

donde nuevamente \hat{y}_k es el valor estimado de la señal de salida, $A_i(x) = \exp(-\sigma_i^2(x - \zeta_i)^2)$ son nuevamente las funciones de membresía y la desfusión se define mediante un promedio ponderado. Las reglas solamente emplean a \hat{y}_k para la partición difusa del espacio, mientras que y_{k-1} , u_k y sus términos subsecuentes son señales empleadas en la capa de desfusión.

Al reescribir en forma vectorial las ecuaciones anteriores, se obtiene nuevamente

una representación compacta

$$\begin{aligned}\hat{y}_{k+1} &= (\varphi_k^1 \mathbf{c}_1^T + \dots + \varphi_k^n \mathbf{C}_{n_R}^T) \mathbf{z}_k + (\varphi_k^1 \mathbf{h}_1^T + \dots + \varphi_k^n \mathbf{h}_{n_R}^T) \xi_k, \\ &= (\varphi_k^T \otimes \mathbf{z}_k) \text{vect}(\mathbf{C}) + (\varphi_k^T \otimes \xi_k) \text{vect}(\mathbf{H}),\end{aligned}\quad (3.57)$$

con las señales definidas en forma similar que en la sección anterior, donde los parámetros a identificar nuevamente son

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_n^T \end{bmatrix}, \mathbf{H} = \begin{bmatrix} \mathbf{h}_1^T \\ \vdots \\ \mathbf{h}_n^T \end{bmatrix}, \sigma = \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_n \end{bmatrix}, \varsigma = \begin{bmatrix} \varsigma_1 \\ \vdots \\ \varsigma_n \end{bmatrix}.$$

Consideremos la función sigmoideal ϕ , la cual actuará como una función *restrictiva*. Es decir, una función que impedirá que los parámetros se alejen más allá de un valor previamente definido. Concretamente, actúa sobre los parámetros antecedentes y evita que por una parte el centro ς_i de los conjuntos difusos salga de la zona de operación de la señal de salida, y por otra que el ancho σ_i sea demasiado estrecho de modo que deje formas sin cubrir.

Sea entonces ϕ una función sigmoide tal que,

- $\phi(\bar{x}, a) \in (-a, a)$,
- $\phi(0) = 0$,
- $\phi \in C_\infty$,
- $\lim_{\bar{x} \rightarrow +\infty} \phi = a$, $\lim_{\bar{x} \rightarrow -\infty} \phi = -a$,
- $\left. \frac{\partial \phi}{\partial \bar{x}} \right|_{\bar{x}=0} = 1$.

De este modo, definiremos a los *parámetros* σ_i y ς_i como funciones dependientes de una función sigmoide y parámetros ajustables $\bar{\sigma}_i$ y $\bar{\varsigma}_i$ respectivamente, es decir

$$\sigma = \phi(\bar{\sigma}, a_\sigma), \text{ y } \varsigma = \phi(\bar{\varsigma}, a_\varsigma).$$

El algoritmo de entrenamiento es aplicado a $\bar{\sigma}$ y $\bar{\varsigma}$. Aunque estos parámetros crezcan hasta llevarlos fuera de la zona de operación de la red, se garantiza que

$$\sigma \in \Omega_\sigma, \quad \Omega_\sigma = \{\sigma \mid |\sigma| < a_\sigma\}, \quad (3.58)$$

$$\varsigma \in \Omega_\varsigma, \quad \Omega_\varsigma = \{\varsigma \mid |\varsigma| < a_\varsigma\}, \quad (3.59)$$

donde los valores a_σ y a_ς actúan como límites para los parámetros.

De esta forma, las ecuaciones de la red se reducen a

$$y_{k+1} = f_1(\mathbf{C}^*, \bar{\sigma}^*, \bar{\zeta}^*, \mathbf{z}_k) + f_2(\mathbf{H}^*, \bar{\sigma}^*, \bar{\zeta}^*, \xi_k) + \underline{\zeta}_k \quad (3.60)$$

$$= f(\mathbf{C}^*, \mathbf{H}^*, \bar{\sigma}^*, \bar{\zeta}^*, \mathbf{z}_k, \xi_k) + \underline{\zeta}_k \quad (3.61)$$

Expandiendo nuevamente en series de Taylor en torno a los valores *subóptimos* de los parámetros, obtenemos

$$f(\mathbf{C}^*, \mathbf{H}^*, \bar{\sigma}^*, \bar{\zeta}^*, \mathbf{z}_k, \xi_k) = f(\mathbf{C}, \mathbf{H}, \bar{\sigma}, \bar{\zeta}, \mathbf{z}_k, \xi_k) + f(\mathbf{C}^* - \mathbf{C}, \mathbf{H}^* - \mathbf{H}, \bar{\sigma}^* - \bar{\sigma}, \bar{\zeta}^* - \bar{\zeta}, \mathbf{z}_k, \xi_k) + (\bar{\sigma}^* - \bar{\sigma})^T \left. \frac{\partial f}{\partial \bar{\sigma}} \right|_{\bar{\sigma}, \bar{\zeta}, \mathbf{c}} + (\bar{\zeta}^* - \bar{\zeta})^T \left. \frac{\partial f}{\partial \bar{\zeta}} \right|_{\bar{\sigma}, \bar{\zeta}, \mathbf{c}} + R_k. \quad (3.62)$$

Con lo que las ecuaciones parciales de las ecuaciones anteriores se calculan como

$$\left. \frac{\partial f_1}{\partial \bar{\sigma}^*} \right|_{\bar{\sigma}^* = \bar{\sigma}} = \frac{\partial \varphi^T}{\partial \bar{\sigma}} \mathbf{Cz}(k) \quad (3.63)$$

$$\begin{aligned} \frac{\partial \varphi^T}{\partial \bar{\sigma}} &= 2 \begin{bmatrix} \frac{\partial \phi(\bar{\sigma}_1, a_\sigma)}{\partial \sigma_1} \frac{\bar{\sigma}_1 (y_k - \bar{\zeta}_1)^2}{(\sum_j \mu_l)} \mu_1 \\ \vdots \\ \frac{\partial \phi(\bar{\sigma}_n, a_\sigma)}{\partial \sigma_n} \frac{\bar{\sigma}_n (y_k - \bar{\zeta}_n)^2}{(\sum_j \mu_l)} \mu_n \end{bmatrix} \otimes [\mu_1 \ \dots \ \mu_n] - \\ &2diag \left(\begin{bmatrix} \frac{\partial \phi(\bar{\sigma}_1, a_\sigma)}{\partial \sigma_1} \frac{\mu_1}{(\sum_l \mu_l)} (\bar{\sigma}_1 (y_k - \bar{\zeta}_1)^2) \\ \vdots \\ \frac{\partial \phi(\bar{\sigma}_n, a_\sigma)}{\partial \sigma_n} \frac{\mu_n}{(\sum_l \mu_l)} (\bar{\sigma}_n (y_k - \bar{\zeta}_n)^2) \end{bmatrix} \right) \end{aligned} \quad (3.64)$$

$$\left. \frac{\partial f_1}{\partial \bar{\zeta}^*} \right|_{\bar{\zeta}^* = \bar{\zeta}} = \frac{\partial \varphi^T}{\partial \bar{\zeta}} \mathbf{Cz}(k) \quad (3.65)$$

$$\begin{aligned} \frac{\partial \varphi^T}{\partial \bar{\zeta}} &= -2 \begin{bmatrix} \frac{\partial \phi(\bar{\zeta}_1, a_\zeta)}{\partial \zeta_1} \frac{\bar{\sigma}_1^2 (y_k - \bar{\zeta}_1)}{(\sum_j \mu_l)} \mu_1 \\ \vdots \\ \frac{\partial \phi(\bar{\zeta}_n, a_\zeta)}{\partial \zeta_n} \frac{\bar{\sigma}_n^2 (y_k - \bar{\zeta}_n)}{(\sum_j \mu_l)} \mu_n \end{bmatrix} \otimes [\mu_1 \ \dots \ \mu_n] + \\ &2diag \left(\begin{bmatrix} \frac{\partial \phi(\bar{\zeta}_1, a_\zeta)}{\partial \zeta_1} \frac{\mu_1}{(\sum_l \mu_l)} (\bar{\sigma}_1^2 (y_k - \bar{\zeta}_1)) \\ \vdots \\ \frac{\partial \phi(\bar{\zeta}_n, a_\zeta)}{\partial \zeta_n} \frac{\mu_n}{(\sum_l \mu_l)} (\bar{\sigma}_n^2 (y_k - \bar{\zeta}_n)) \end{bmatrix} \right) \end{aligned} \quad (3.66)$$

Nótese que estas ecuaciones son muy similares a las obtenidas en (3.20), con el término $\frac{\partial \phi(\bar{\zeta}_i, a_\zeta)}{\partial \zeta_i}$ añadido en forma multiplicativa en (3.64) (en forma similar ocurre

para el término $\frac{\partial \phi(\bar{\sigma}_i, a_\sigma)}{\partial \sigma_i}$ en (3.66)). De este modo, se puede emplear el mismo algoritmo de inicialización analizado en el caso anterior.

Es posible determinar la estabilidad de la red con restricciones en los parámetros antecedentes con base en los mismos resultados reportados en la sección 3.1.1.3.

3.2. Control basado en predicción un-paso-adelante

La red entrada-salida en tiempo discreto puede ser empleada para controlar un sistema previamente identificado empleando una estructura de control de equivalencia cierta (Wang, 1994) por predicción un-paso-adelante. De este modo, podemos reescribir a la red (3.3) como:

$$\begin{aligned} \hat{y}_{k+1} &= (\varphi_k^T \otimes \mathbf{z}_k) \text{vect}(\mathbf{C}) + \varphi_k^T (\mathbf{h}_{c1} u_k + \bar{\mathbf{H}} \bar{\xi}_k) \\ &= \hat{f}_y(\mathbf{z}_k, \bar{\xi}_k, \mathbf{C}, \bar{\mathbf{H}}) + \hat{f}_u u_k = \hat{f}_y + \hat{f}_u u_k \end{aligned} \quad (3.67)$$

donde $\hat{f}_y(\mathbf{z}_k, \bar{\xi}_k, \mathbf{C}, \bar{\mathbf{H}}) = (\varphi_k^T \otimes \mathbf{z}_k) \text{vect}(\mathbf{C}) + \varphi_k^T \bar{\mathbf{H}} \bar{\xi}_k$, $\hat{f}_u = \varphi_k^T \mathbf{h}_{c1}$, \mathbf{h}_{c1} es la primer columna de la matriz \mathbf{H} y $\bar{\mathbf{H}} = [\mathbf{h}_{c2} \dots \mathbf{h}_{cn_R}]$ es la misma matriz \mathbf{H} con la primer columna removida. Por otra parte, $\bar{\xi}_k = [u_{k-1} \dots u_{k-m+1}]^T$. Si tras el entrenamiento la red cumple con $\hat{f}_u = \varphi^T \mathbf{h}_{c1} \neq 0$, es decir, que no existan puntos de singularidad para el control identificados por la red, es posible aplicar un control de equivalencia cierta un-paso-adelante, como se indica en el siguiente teorema:

Teorema 3.3. *Si la red (3.67) no tiene puntos singulares para el control (esto es, $\varphi^T \mathbf{h}_{c1} \neq 0 \forall k$), $u \in [\underline{u}, \bar{u}] = U$, el control*

$$u_k = \begin{cases} v_k & , \quad v_k \in U \\ \bar{u} & , \quad v_k > U \\ \underline{u} & , \quad v_k < U \end{cases} \quad (3.68)$$

$$v_k = \frac{1}{\varphi \mathbf{h}_{c1}} \left(-(\varphi_k^T \otimes \mathbf{z}_k) \text{vect}(\mathbf{C}) - \varphi_k^T \bar{\mathbf{H}} \bar{\xi}_k + y_{d(k+1)} \right) = \frac{-\hat{f}_y + y_{d(k+1)}}{\hat{f}_u}. \quad (3.69)$$

hace que la red alcance cualquier punto deseado $y_{d(k+1)}$ dentro del espacio de trabajo de la misma en tiempo finito. Si el control es aplicado al sistema real, el error de seguimiento $e_{k+1} = y_{d(k+1)} - y_{k+1}$ es acotado mientras el error de modelado ε sea acotado.

Demostración. Al identificar el sistema no lineal $y_{k+1} = f(\cdot)$, la aproximación dada por la red está dada por (3.3). Al aplicar el control propuesto a (3.67), los términos se cancelan y la salida del la misma en tiempo $k + 1$ es

$$y_{k+1} = y_{d(k+1)} \quad (3.70)$$

Al suponer que el sistema original tiene una estructura afín en el control u_k , identificada por la red con parámetros *óptimos* en la forma

$$y_{k+1} = f_y(\mathbf{z}_k, \xi_{k-1}, \theta^*) + f_u(\mathbf{z}, \xi_{k-1}, \theta^*)u_k + \vartheta_k, \quad (3.71)$$

con $\tilde{f}_y = \hat{f}_y - f_y$ y $\tilde{f}_u = \hat{f}_u - f_u$, se obtiene, al aplicar el control al sistema original,

$$\begin{aligned} y_{k+1} &= f_y + f_u \left(\frac{1}{\hat{f}_u} \left(-\hat{f}_y + y_{d(k+1)} \right) \right) + \zeta_k \\ &= f_y - \hat{f}_y + \hat{f}_y + \left(\frac{f_u - \hat{f}_u + \hat{f}_u}{\hat{f}_u} \left(-\hat{f}_y + y_{d(k+1)} \right) \right) + \zeta_k \\ &= \hat{f}_y - \tilde{f}_y + \left(\frac{\hat{f}_u - \tilde{f}_u}{\hat{f}_u} \left(-\hat{f}_y + y_{d(k+1)} \right) \right) + \zeta_k \\ &= \hat{f}_y + \frac{\hat{f}_u}{\hat{f}_u} \left(-\hat{f}_y + y_{d(k+1)} \right) - \tilde{f}_y + \frac{\tilde{f}_u}{\hat{f}_u} \left(\hat{f}_y - y_{d(k+1)} \right) + \zeta_k \\ &= y_{d(k+1)} - \tilde{f}_y + \tilde{f}_u \overbrace{\left(\frac{\hat{f}_y - y_{d(k+1)}}{\hat{f}_u} \right)}^{-u_k} + \zeta_k = y_{d(k+1)} - \left(\tilde{f}_y + \tilde{f}_u u_k \right) + \zeta_k \\ &= y_{d(k+1)} + \tilde{y}_{k+1} + 2\zeta_k - R_k \end{aligned}$$

mostrando que el error de seguimiento depende del error de identificación/modelado. \square

Dado que no es posible asumir que los parámetros identificados sean los *ideales* (ya que el algoritmo de entrenamiento elegido solo asume la existencia de un mínimo local), es posible añadir un término integral extra al control tal que maneje dicha incertidumbre en la identificación. De este modo, el control propuesto es

$$\bar{u}_k = u_k + \frac{1}{\varphi \mathbf{h}_{c1}} \left(-K_p e_k - K_i \sum_{j=1}^k e_j \right). \quad (3.72)$$

3.3. Ejemplos

Para analizar las capacidades de identificación y control de esta red, se prueba nuevamente con el sistema (3.55), descrito en la Sección 3.1.2.4. En este caso el algoritmo de identificación se prueba nuevamente empleando una señal pseudoaleatoria con cota $|u| \leq 1,02$, con 8,000 datos entrada/salida, empleándose $n_R = 4$, $n = 3$, $m = 2$, $a_\sigma = 2$ y $a_\zeta = 3$ (28 parámetros en total). El algoritmo de entrenamiento elegido fue aquél basado en el análisis de Lyapunov visto en la Sección 3.1.2.3, lográndose un error RMS de 0.03692. Los resultados del proceso de identificación se muestran en la Fig. 3.7, donde se aprecia que el desempeño es similar al mostrado por la red sin restricciones en los parámetros en la Fig. 3.3. Sin embargo, en la Fig. 3.9 se muestran los conjuntos difusos obtenidos al final del proceso de entrenamiento, donde ahora ninguno de ellos sale de la zona de operación de la red, a diferencia de lo mostrado en la Fig. 3.5.

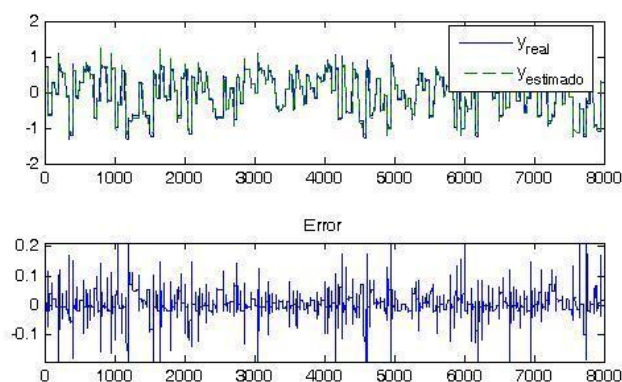


Figura 3.7: Identificación de un sistema no lineal: Red con restricciones en parámetros.

Una vez que la red ha sido entrenada, se aplica el control de equivalencia cierta al sistema (3.55). Dado que no existen puntos *singulares* para la red ya entrenada. Los resultados obtenidos se muestran la Fig. 3.10.a, donde se logra un error RMS entre la señal deseada y la obtenida de 0.03945. Sin embargo, al añadir una compensación PI mediante los valores $K_p = 0,2$ y $K_i = 0,12$, el error RMS se reduce a 0.01744. Los resultados para este control compensado se muestran en la Fig. 3.10.b.

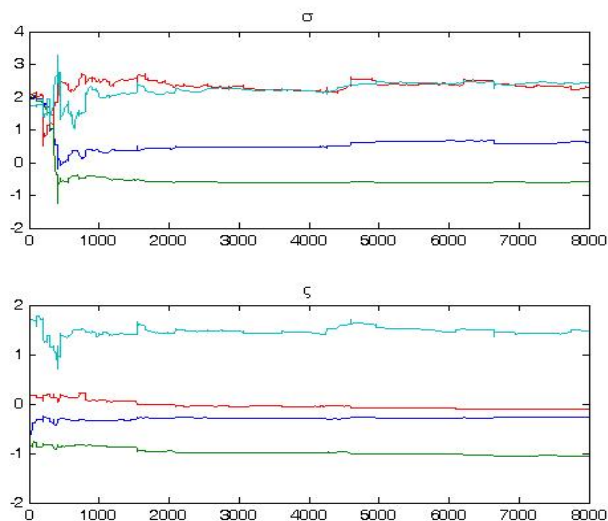


Figura 3.8: Identificación de un sistema no lineal: Evolución de los parámetros en una red con restricciones en parámetros

3.4. Comentarios generales

Los casos vistos anteriormente muestran cómo una red recurrente (en este caso, en tiempo discreto) es capaz de modelar e identificar el comportamiento entrada/salida de sistemas dinámicos no lineales. Se observa cómo al aplicar un control por equivalencia cierta el error de seguimiento depende de la calidad de aproximación de la red. Se logra compensar este error mediante un control con un componente integral. De acuerdo con

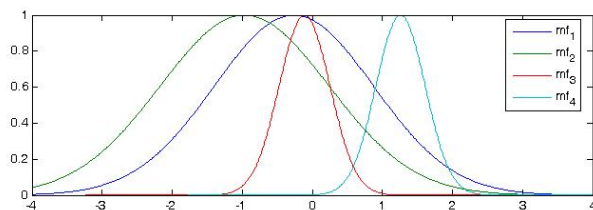


Figura 3.9: Identificación de un sistema no lineal: Funciones de membresía tras entrenamiento

resultados previos de redes neuronales y sistemas difusos, es posible que la calidad de la aproximación (y en consecuencia del control) mejoren bajo un mayor número de reglas, evitando caer en la *maldición de la dimensionalidad*.

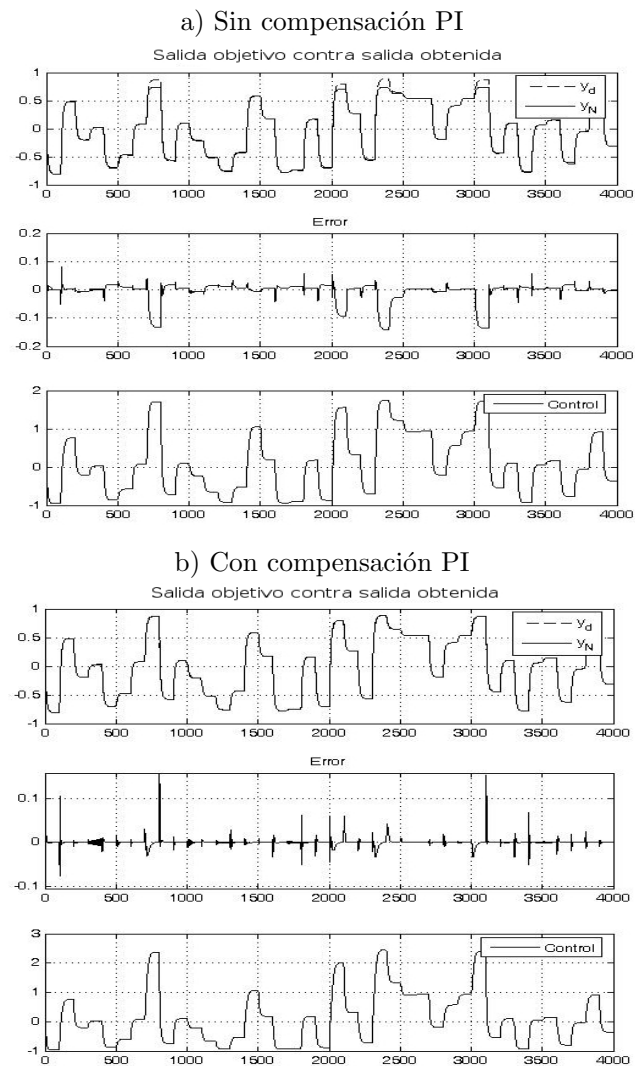


Figura 3.10: Control de un sistema no lineal: Red con restricciones en parámetros.

3. IDENTIFICACIÓN MEDIANTE REDES RECURRENTE ENTRADA-SALIDA

Capítulo 4

Identificación mediante redes recurrentes en espacio de estados

La estructura mostrada en el capítulo anterior fue desarrollada en tiempo discreto, e incorpora algoritmos de entrenamiento en cierta forma *típicos*, como mínimos cuadrados o el filtro de Kalman. En ella la medición de los estados no es estrictamente necesaria, ya que es posible obtener retrasos en tiempo de la salida y de la entrada y con base en ellos establecer la red neurodifusa e identificar sus parámetros.

Sin embargo, en tiempo continuo el paradigma es distinto, ya que para realizar un proceso similar sería necesario obtener las $n - 1$ derivadas con respecto al tiempo de la señal de salida. Si bien esto es matemáticamente posible, en la práctica resulta inconveniente, ya que al operar con señales obtenidas de un experimento, la derivación directa implica la amplificación de los componentes de alta frecuencia, generalmente asociados al ruido. Por ello, es necesario replantear el problema de modelado e identificación con redes neurodifusas recurrentes cuando solo se dispone de mediciones entrada salida.

Considérese de este modo a la clase de sistemas MISO¹ no lineales que pueden ser representados mediante

$$\Sigma_O : \begin{cases} \dot{\mathbf{x}}(t) &= f(\mathbf{x}, \mathbf{u}), \\ y(t) &= h(\mathbf{x}), \end{cases} \quad (4.1)$$

donde $u(t) \in \mathfrak{R}^m$, $y(t) \in \mathfrak{R}$ son las señales de entrada y salida respectivamente, con $\mathbf{x} \in \mathfrak{R}^n$ el estado del sistema en tiempo t , $f : \mathfrak{R}^n \times \mathfrak{R}^m \rightarrow \mathfrak{R}^n$ una función Lipschitz continua y $h : \mathfrak{R}^n \rightarrow \mathfrak{R}$ una función suave.

¹Múltiples Entradas, Una Salida, por sus siglas en inglés *Multiple Input, Multiple Output*.

Con el fin de proponer un algoritmo que logre identificar el comportamiento entrada-salida de este sistema, es necesario contar con un cierto conocimiento del mismo. En primer lugar, tómesese la clase de sistemas transformable a la forma

$$\begin{cases} \dot{\mathbf{z}} &= A\mathbf{z} + \bar{f}(y, u), \quad \mathbf{z} = T(\mathbf{x}), \\ y &= C\mathbf{z}. \end{cases} \quad (4.2)$$

donde $\bar{f}(y, u)$ es una función *desconocida*, además

$$A = \begin{bmatrix} \mathbf{0}_{n \times 1} & I_{(n-1) \times (n-1)} \\ & \mathbf{0}_{1 \times (n-1)} \end{bmatrix}, \quad (4.3)$$

$$C = [1 \quad \mathbf{0}_{1 \times (n-1)}]. \quad (4.4)$$

En Marino (1990); Marino & Tomei (1996), así como en el Apéndice A.1, se encuentran las condiciones necesarias y suficientes que debe cumplir este sistema para ser transformado, a partir de un difeomorfismo $\mathbf{z} = T(\mathbf{x})$, en el sistema (4.2), las cuales son condiciones de observabilidad, controlabilidad y existencia de la transformación. Las primeras dos condiciones implican que toda la dinámica introducida a partir de la entrada se refleje en los estados, y que a su vez estos puedan ser reconstruidos a partir de mediciones de la señal de salida. En general, se asume que los sistemas a identificar cumplen con esta condición.

El objetivo es encontrar una red neurodifusa recurrente en tiempo continuo tal que, al emplear únicamente mediciones de la salida $y(t)$ y el conocimiento de la entrada $u(t)$ pueda generar un mapeo dinámico $u \rightarrow \hat{y}$ tal que la respuesta del mismo sea lo más cercana posible a aquella de la planta. Esto es, tratar de hacer el error de identificación $\sup_{t \geq 0} |\hat{y}(t) - y(t)|$ tan pequeño como sea posible.

Para dicho propósito, se propone la siguiente estructura de una red neurodifusa recurrente:

$$\Sigma : \begin{cases} \dot{\eta} &= A\eta + \varphi^*(\hat{y}, u)\theta'_\eta, \quad \eta \in \mathbb{R}^n, \\ \hat{y} &= C\eta, \end{cases} \quad (4.5)$$

con $\dim \eta = n$. Nótese que el componente $\varphi^*(\hat{y}, u)\theta'_\eta$ aproxima a \bar{f} en (4.2). Sin embargo, dado que A y C son observables por construcción y de estructura conocida, es posible diseñar una dinámica deseada para la parte lineal de la red mediante una inyección de salida Ly :

$$\dot{\eta} = A\eta + \varphi^*(\hat{y}, u)\theta'_\eta + L\hat{y} - L\hat{y}. \quad (4.6)$$

Al reescribir la ecuación anterior, se obtiene

$$\dot{\eta} = (A - LC)\eta + \overbrace{\varphi^*(\hat{y}, u)\theta'_\eta + L\hat{y}}^{\varphi(\hat{y}, u)\theta_\eta}. \quad (4.7)$$

Finalmente, la red se reescribe como:

$$\Sigma : \begin{cases} \dot{\eta} &= \bar{A}\eta + \varphi(\hat{y}, u)\theta_\eta, \\ \hat{y} &= C\eta, \end{cases} \quad (4.8)$$

con $\bar{A} = A - LC$ y $\varphi(y, u)\theta_\eta = \varphi^*(y, u)\theta'_\eta + Ly$. De este modo, el objetivo es ahora encontrar el vector de parámetros θ_η tal que logre que $\sup_{t \geq 0, \theta} |y(t) - \hat{y}(t)|$ sea tan pequeño como sea posible. Con base en el trabajo de Zhang (2002), es posible resolverlo como un problema dual de identificación: bajo las condiciones de observabilidad para el par (A, C) y una condición de excitación persistente para $\varphi(y, u)$, diseñar un observador adaptable que estime en forma conjunta \hat{y} y θ_η .

4.1. Estructura de la red: Tiempo continuo

Para llevar a cabo el modelado del sistema con base en una red neurodifusa recurrente, se propone la siguiente estructura:

$$\Sigma_R : \begin{cases} \text{Regla 1} & : \text{ Si } R_1(y, u) \text{ entonces} \\ & \dot{\eta}^1 = A\eta - Ly + \mathbf{r}_1(y, u)\theta_{\mathbf{1}}, \\ & \vdots \\ \text{Regla } i & : \text{ Si } R_i(y, u) \text{ entonces} \\ & \dot{\eta}^i = A\eta - Ly + \mathbf{r}_i(y, u)\theta_{\mathbf{i}}, \\ & \vdots \\ \text{Regla } r & : \text{ Si } R_r(y, u) \text{ entonces} \\ & \dot{\eta}^r = A\eta - Ly + \mathbf{r}_r(y, u)\theta_{\mathbf{r}}, \end{cases}$$

donde $\mathbf{r}_i : \mathfrak{R} \times \mathfrak{R}^m \rightarrow \mathfrak{R}^{m \times p_i}$ es una función propuesta de u y/o y , $\theta_i \in \mathfrak{R}^{p_i}$ es el vector de parámetros de la i -ésima regla, $R_i : \mathfrak{R} \times \mathfrak{R}^m \rightarrow \mathfrak{R}$ es el *enunciado difuso* dependiente de u y y . Como R_i es también el nivel de activación o peso de cada regla, empleando una suma ponderada de cada una de las funciones $\varphi(y, u)\theta_\eta$ en (4.8) se calcula como

$$\varphi(y, u)\theta_\eta = \sum_{i=1}^r R_i \mathbf{r}_i(y, u)\theta_{\mathbf{i}}. \quad (4.9)$$

La red recurrente queda entonces definida mediante

$$\dot{\eta} = \bar{A}\eta + \sum_{i=1}^r R_i \mathbf{r}_i(y, u) \theta_i = \bar{A}\eta + \varphi(y, u) \theta_\eta. \quad (4.10)$$

donde ahora $\varphi(y, u) = [R_1 \mathbf{r}_1(y, u) \ \cdots \ R_r \mathbf{r}_r(y, u)]$, y $\theta_\eta = [\theta_1^T \ \cdots \ \theta_r^T]^T$.

Nota 4.1. *La estructura de la red está inspirada en la investigación realizada en observadores adaptables dada de Bastin & Gevers (1988); Marino (1990); Zhang (2002), en el que el regresor puede ser obtenido como una función de las señales de entrada y salida de la planta. La parte consecuyente de las reglas difusas puede diseñarse de modo tal que permita evitar una explosión de reglas, pero que satisfaga la condición de excitación persistente, necesaria para el proceso de identificación paramétrica que se describirá en la sección siguiente.*

4.2. Algoritmo de entrenamiento

Se considera la red definida por (4.8), que será empleada como modelo en paralelo del sistema (4.2), el problema de identificación es el estimar conjuntamente los estados η y los parámetros θ_η . Dado el par (A, C) observable y L conocido del proceso de diseño para obtener una inyección de salida tal que preserve la observabilidad del par (\bar{A}, C) , es posible encontrar un nuevo vector $K \in \mathfrak{R}^n$ tal que la matriz $A - LC - KC = \bar{A} - KC$ sea Hurwitz.

Nota 4.2. *La matriz $\bar{A} = A - LC$ se diseña de forma tal que la parte homogénea de la red sea exponencialmente estable (la matriz \bar{A} sea Hurwitz), con valores característicos deseados; mientras que la matriz $\bar{A} - KC$ es igualmente diseñada para ser Hurwitz para el algoritmo de entrenamiento/identificación, el cual está basado en la teoría de observadores adaptables.*

Por simplicidad, se denota $\varphi = \varphi(y, u)$. De este modo, se considera que la inyección de salida está dada por

$$\dot{\eta} = (\bar{A} - KC)\eta + \varphi \theta_\eta + Ky. \quad (4.11)$$

Siguiendo un tratamiento similar al desarrollado por Zhang (2002), los estados pueden ser descompuestos, previa transformación, en

$$\eta = \eta_y + \eta_\varphi, \quad (4.12)$$

donde η se compone por la combinación lineal de las señales η_y, η_φ ,

$$\dot{\eta}_y = (\bar{A} - KC)\eta_y + Ky, \quad (4.13)$$

$$\dot{\eta}_\varphi = (\bar{A} - KC)\eta_\varphi + \varphi\theta_\eta. \quad (4.14)$$

Al ser la dinámica de η_y lineal, se puede diseñar un observador

$$\dot{\hat{\eta}}_y = (\bar{A} - KC)\hat{\eta}_y + Ky. \quad (4.15)$$

Por otra parte, un observador para η_φ se propone como:

$$\dot{\hat{\eta}}_\varphi = (\bar{A} - KC)\hat{\eta}_\varphi + \varphi\hat{\theta}_\eta + \Upsilon(t)\dot{\hat{\theta}}_\eta, \quad (4.16)$$

donde Υ define una transformación dinámica $\hat{\eta}_\varphi = \Upsilon(t)\hat{\theta}_\eta$. Al desarrollar la derivada con respecto al tiempo de $\hat{\eta}_\varphi$,

$$\frac{d\hat{\eta}_\varphi}{dt} = \dot{\Upsilon}\hat{\theta}_\eta + \Upsilon\dot{\hat{\theta}}_\eta = (\bar{A} - KC)\Upsilon\hat{\theta}_\eta + \varphi\hat{\theta}_\eta + \Upsilon\dot{\hat{\theta}}_\eta, \quad (4.17)$$

se obtiene

$$\dot{\Upsilon}\hat{\theta}_\eta = (\bar{A} - KC)\Upsilon\hat{\theta}_\eta + \varphi\hat{\theta}_\eta, \quad (4.18)$$

$$\left(\dot{\Upsilon} - (\bar{A} - KC)\Upsilon - \varphi\right)\hat{\theta}_\eta = 0. \quad (4.19)$$

La segunda ecuación resulta en

$$\dot{\Upsilon} = (\bar{A} - KC)\Upsilon + \varphi, \quad (4.20)$$

la cual describe una matriz filtrada del regresor φ . Al ser φ acotada, de igual forma lo será la matriz Υ , dado que $\bar{A} - KC$ es una matriz Hurwitz. El algoritmo de entrenamiento propuesto toma la forma

$$\dot{\hat{\theta}}_\eta = \Gamma\Upsilon^T C^T (y - C\hat{\eta}), \quad (4.21)$$

donde Γ es una matriz de ganancia positiva definida. La estabilidad y convergencia de la red y del algoritmo propuesto se resume en el Teorema 4.1.

Teorema 4.1. *Si se considera al sistema (4.1) y se asume que es observable en la región de interés y transformable mediante un difeomorfismo en el sistema equivalente dado por (4.2), la siguiente red neurodifusa*

$$\dot{\hat{\eta}} = \bar{A}\hat{\eta} + \varphi(y, u)\hat{\theta}_\eta + (K + \Upsilon\Gamma\Upsilon^T C^T)(y - C\hat{\eta}), \quad (4.22)$$

con el algoritmo de entrenamiento

$$\dot{\hat{\theta}}_\eta = \Gamma \Upsilon^T C^T (y - C\hat{\eta}), \quad (4.23)$$

donde $\Gamma > 0$ y la matriz de ganancia $K \in \mathbb{R}^n$ es tal que $A - KC$ es Hurwitz, y

$$\dot{\Upsilon} = (\bar{A} - KC) \Upsilon + \varphi(y, u), \quad (4.24)$$

un filtro para $\varphi(y, u)$, garantiza que todas las señales estén acotadas, y que el error de estimación de estados $\tilde{\mathbf{z}} = \hat{\eta} - \mathbf{z}$ y el error paramétrico $\tilde{\theta} = \hat{\theta}_\eta - \theta_\eta$ poseen estabilidad práctica (sean uniforme y finalmente acotados), siempre y cuando el regresor cumpla con la condición de excitación persistente, esto es:

$$\int_t^{t+T} \varphi^T(\tau) \varphi(\tau) d\tau \geq \alpha I. \quad (4.25)$$

para alguna $0 < \alpha$, y $T > 0$, donde $\varphi(\tau) = \varphi(y(\tau), u(\tau))$.

Demostración. A partir del Teorema de Stone-Weierstrass, la función suave $\bar{f}(y, u) + Ly$ puede ser aproximada mediante $\varphi(y, u)\theta_\eta$ con una precisión arbitraria en un conjunto compacto y cerrado de la región de interés, y entonces $\bar{f}(y, u) + Ly - \varphi(y, u)\theta_\eta = \varepsilon(t)$, siendo $\varepsilon(t)$ el error de aproximación para los parámetros reales θ_η ; de esta manera el sistema (4.2) puede representarse mediante

$$\dot{\mathbf{z}} = \bar{A}\mathbf{z} + \varphi(y, u)\theta_\eta + \varepsilon(t), \quad (4.26)$$

$$y = C\mathbf{z}. \quad (4.27)$$

Denotando $\varphi = \varphi(y, u)$ y considerando una red (4.8) tal que η sea un estimado de \mathbf{z} , se define la señal de error

$$\xi(t) = \tilde{\mathbf{z}} - \Upsilon\tilde{\theta}. \quad (4.28)$$

La derivada con respecto al tiempo de la ecuación anterior es la dinámica de la señal de error

$$\dot{\xi}(t) = (\bar{A} - KC)\xi - \varepsilon, \quad (4.29)$$

$$\begin{aligned} \dot{\tilde{\theta}} &= -\Gamma \Upsilon^T C^T C (\xi + \Upsilon\tilde{\theta}) \\ &= -\Gamma \Upsilon^T C^T C \Upsilon \tilde{\theta} - \Gamma \Upsilon^T C^T C \xi. \end{aligned} \quad (4.30)$$

De lo anterior, se deduce que mientras ε esté acotado, ξ también estará acotado, dado que la matriz $A - KC$ es Hurwitz. Incluso, Υ debe ser acotada dado que es obtenida a partir de un filtrado asintóticamente estable de φ , que a su vez es acotado por diseño.

Ahora, es posible probar (Narendra & Parthasarathy, 1990; Zhang, 2002) que bajo una condición de excitación persistente (4.25), el origen del sistema

$$\dot{\chi} = -\Gamma\varphi^T\varphi\chi \quad (4.31)$$

es exponencialmente estable. Lo anterior implica que la parte homogénea de (4.30) es también exponencialmente estable. Por lo tanto, la ecuación (4.30) es asimismo exponencialmente estable con una perturbación acotada, lo que implica la cota final de $\tilde{\theta}$.

A partir de (4.29) y definiendo $A_0 = \bar{A} - KC$, la solución está dada por

$$\xi(t) = e^{A_0 t}\xi(0) - \int_0^t e^{A_0(t-\tau)}\epsilon(\tau)d\tau. \quad (4.32)$$

Empleando la cota $\|e^{A_0(t-t_0)}\| \leq ke^{-\lambda_1(t-t_0)}$, donde $\lambda_1 = |\Re\{\lambda_{\min}(A_0)\}|$, la ecuación anterior se acota mediante

$$\|\xi(t)\| \leq ke^{-\lambda_1 t} \|\xi(0)\| + \frac{k}{\lambda_1} \sup_{t \geq 0} \|\epsilon\|. \quad (4.33)$$

De este modo, $\lim_{t \rightarrow \infty} \|\xi(t)\| \leq \frac{k}{\lambda_1} \sup_{t \geq 0} \|\epsilon\|$.

Se sabe que, a partir de (4.30), existe una matriz de transición $\psi(t, t_0)$ tal que

$$\tilde{\theta}(t) = \psi(t, t_0)\tilde{\theta}(t_0) - \int_{t_0}^t \psi(t, \tau)\Gamma\Upsilon^T C^T C\xi(\tau)d\tau. \quad (4.34)$$

El origen de la parte homogénea de (4.30), bajo una condición de excitación persistente (4.25), es exponencialmente estable, como se demuestra en (Anderson & Moore, 1979; Zhang, 2002). Con esto en mente, se sabe que existe un par de constantes positivas k_2 y λ_2 tales que, para la matriz de transición

$$\|\psi(t, t_0)\| \leq k_2 e^{-\lambda_2(t-t_0)}. \quad (4.35)$$

En (4.34), tenemos entonces que

$$\begin{aligned} \|\tilde{\theta}(t)\| &\leq \left\| \psi(t, t_0)\tilde{\theta}(t_0) + \int_{t_0}^t \psi(t, \tau)\Gamma\Upsilon^T C^T C\xi(\tau)d\tau \right\|, \\ &\leq k_2 e^{-\lambda_2(t-t_0)} \|\tilde{\theta}(t_0)\| + \left\| \int_{t_0}^t k_2 e^{-\lambda_2(t-\tau)}\Gamma\Upsilon^T C^T C\xi(\tau)d\tau \right\|, \\ &\leq k_2 e^{-\lambda_2(t-t_0)} \|\tilde{\theta}(t_0)\| + \sup_{t \geq 0} \|\Gamma\Upsilon^T C^T C\xi(\tau)\| \left\| \int_{t_0}^t k_2 e^{-\lambda_2(t-\tau)}d\tau \right\|, \\ &\leq k_2 e^{-\lambda_2(t-t_0)} \|\tilde{\theta}(t_0)\| + \frac{k_2}{\lambda_2} \left(1 - e^{-\lambda_2(t-t_0)}\right) \sup_{t \geq 0} \|\Gamma\Upsilon^T C^T C\xi(\tau)\|. \end{aligned} \quad (4.36)$$

Finalmente, en el límite $t \rightarrow \infty$, en (4.36) la condición inicial se desvanece, y por lo tanto

$$\bar{\theta} \triangleq \lim_{t \rightarrow \infty} \|\tilde{\theta}(t)\| \leq \frac{k_2}{\lambda_2} \sup_{t \geq 0} \|\Gamma \Upsilon^T C^T C \xi(\tau)\|. \quad (4.37)$$

□

Como se indica en la prueba, el que el regresor cumpla con la condición de excitación persistente (4.25) es crucial para el análisis y el proceso de identificación. A continuación se describirán algunas elecciones de reglas difusas que permiten obtener regresores que cumplen con esta condición.

4.2.1. Casos particulares de regresores

En esta sección se analizan algunos casos particulares de regresores, cuya estructura permite determinar las condiciones bajo las cuales las señales generadas por este cumplen con la condición de excitación persistente (4.25) ante la variación de las señales u y y . En primer lugar, se muestra un caso general con una forma lineal tipo Takagi-Sugeno. A continuación, se describen dos estructuras simplificadas. La primera de ellas solo lleva a cabo la partición del espacio en y . Sin embargo, dependiendo del sistema en particular a identificar, esta partición puede ser insuficiente, por lo que se mostrarán diversas elecciones de regresores cuya partición del espacio se lleve en términos de y y u . Se propondrán también diversas definiciones para el proceso de desdifusión.

Regresor lineal general En este caso las reglas difusas en (4.9) pueden se expresan como

$$\begin{cases} R_i & : \text{ Si } y \text{ es } F_i(y) \text{ y } \mathbf{u} \text{ es } H_i(\mathbf{u}) \text{ entonces} \\ \dot{\eta}^1 & = \bar{A}\eta + \boldsymbol{\theta}_{\mathbf{u},i}\mathbf{u} + \boldsymbol{\theta}_{y,i}\rho(y), \end{cases} \quad (4.38)$$

donde

$$F_i(y) = \exp\left(-\sigma_{y,i}^2 (y - \mu_{y,i})^2\right), \quad H_i(\mathbf{u}) = \exp\left(-\sum_{j=1}^m \sigma_{u_j,i}^2 (u_j - \mu_{u_j,i})^2\right) \quad (4.39)$$

son funciones de membresía gaussianas, donde $\mathbf{u} = [u_1 \ \dots \ u_m]^T$ y

$$\begin{aligned} \boldsymbol{\theta}_{u_j,i} &= [\theta_{u_j,i,1} \ \theta_{u_j,i,2} \ \dots \ \theta_{u_1,i,n}]^T, \\ \boldsymbol{\theta}_{y,i} &= [\theta_{y,i,1} \ \theta_{y,i,2} \ \dots \ \theta_{y,i,n}]^T. \end{aligned}$$

La activación de cada regla, y por lo tanto la función conjunción entre conjuntos difusos se define como el producto

$$R_i(y, \mathbf{u}) = F_i(y)H_i(\mathbf{u}). \quad (4.40)$$

Nota 4.3. *El principal propósito de la estructura difusa a la entrada de la red es proveer con una partición del espacio $y - u$ y permitir que cada sección i sea modelada por un sistema no lineal en la forma R_i .*

Regresor como función de y

Sin pérdida de generalidad, por simplicidad de presentación se mostrará el caso SISO. En primer lugar, para evitar una *explosión* paramétrica, las reglas difusas en (4.9) se expresan como

$$R_i : \quad \text{Si } y \text{ es } F_i(y) \text{ entonces } \dot{\eta}^i = \bar{A}\eta + \begin{bmatrix} 0_{(i-1) \times 1} \\ \theta_{i1} + \theta_{i2}u \\ 0_{(r-i) \times 1} \end{bmatrix}. \quad (4.41)$$

donde $F_i(y) = \exp(-\sigma_{y,i}^2(y - \mu_{y,i})^2)$ son funciones de membresía gaussianas. Obsérvese que para este caso el número de reglas es el mismo que el número de estados. En este caso, el proceso de desdifusión puede ser tomado como un promedio ponderado, o como una suma ponderada. Si este es efectuado únicamente en la parte consecuente, la red recurrente toma la forma

$$\begin{aligned} \dot{\eta} &= \bar{A}\eta + \overbrace{\begin{bmatrix} F_1 & \cdots & 0 & \cdots & 0 \\ 0 & \ddots & F_i & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & F_r \end{bmatrix}}^G \begin{bmatrix} \theta_{11} \\ \vdots \\ \theta_{i1} \\ \vdots \\ \theta_{r1} \end{bmatrix}, \\ &+ \overbrace{\begin{bmatrix} F_1u & \cdots & 0 & \cdots & 0 \\ 0 & \ddots & F_iu & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & F_ru \end{bmatrix}}^{Gu} \begin{bmatrix} \theta_{12} \\ \theta_{22} \\ \vdots \\ \theta_{i2} \\ \vdots \\ \theta_{r2} \end{bmatrix}, \\ &= \bar{A}\eta + [G \quad Gu] \theta_\eta \end{aligned} \quad (4.42)$$

En consecuencia, el regresor se define como

$$\varphi(y, u) = \begin{bmatrix} G & \vdots & Gu \end{bmatrix} \in \mathfrak{R}^{n \times 2r}, \quad (4.43)$$

donde $n = r$ y

$$\theta_{\eta}^T = [\theta_{11} \quad \cdots \quad \theta_{r1} \quad \theta_{12} \quad \cdots \quad \theta_{r2}] \in \mathfrak{R}^{2r} \quad (4.44)$$

$$G = \text{diag}(F_1, \dots, F_r) \in \mathfrak{R}^r, G_{ii} = F_i, i = 1 \dots r. \quad (4.45)$$

El siguiente lema demuestra las condiciones bajo las cuales (4.43) satisface la condición de excitación persistente (4.25).

Lema 4.1. *Dado (4.43), y por simplicidad se denota $\varphi(t) = \varphi(y(t), u(t))$, existe $T, \alpha \in \mathfrak{R}^+$ tal que se satisface*

$$\int_t^{t+T} \varphi^T(t) \varphi(t) dt \geq \alpha I_{r \times r} \quad (4.46)$$

mientras $u(t)$ no sea constante ni linealmente dependiente de F_i , y que además cada regla se activada en el intervalo de tiempo $[t, t+T]$ ($F_i \neq 0$, en $i = 1, \dots, r$).

Demostración. En primer lugar, tenemos que $G = G^T$, y por lo tanto $G^T G = G^2$. Entonces, el producto $\varphi^T \varphi$ se desarrolla mediante

$$\varphi^T \varphi = \begin{bmatrix} G^2 & G^2 u \\ G^2 u & G^2 u^2 \end{bmatrix} \quad (4.47)$$

Con un pequeño abuso de notación, denotaremos $\int_t^{t+T} f(x) dx$ por $\int f(x) dx$. Con esto en mente, es claro que si para el intervalo $[t, t+T]$ el sistema activa cada $F_i, i = 1, \dots, r$, entonces $\int G^2 dt > \alpha_1 I$, dado que $\int G_i^2 dt > \alpha_1$, and $\alpha_1 = \min_i \{\alpha_1\}$.

Se define

$$M \triangleq \int \varphi^T \varphi dt = \begin{bmatrix} \int G^2 dt & \int G^2 u dt \\ \int G^2 u dt & \int G^2 u^2 dt \end{bmatrix}. \quad (4.48)$$

Mediante el cálculo del determinante de esta matriz es posible evaluar si

- Pierde rango para algún valor de G y u (i.e., $\det(M) = 0$),
- Es positiva definida, y por lo tanto $\exists T, \alpha > 0$ tal que $M \geq \alpha I$.

Se sabe que para una matriz cuadrada $M = \begin{bmatrix} A & B \\ B & D \end{bmatrix}$, con $A \in \mathfrak{R}^{p \times p}$, $B \in \mathfrak{R}^{p \times q}$ y $D \in \mathfrak{R}^{q \times q}$, se puede evaluar su determinante por sus complementos de Schur mediante

$$\det(M) = \det(A) \det(D - BA^{-1}B). \quad (4.49)$$

Al definir $A = \int G^2 dt$, $B = \int G^2 u dt$, $D = \int G^2 u^2 dt$, y emplear (4.49), obtenemos

$$\det(M) = \det \left(\int G^2 dt \right) \times \det \left(\int G^2 u^2 dt - \int G^2 u dt \left(\int G^2 dt \right)^{-1} \int G^2 u dt \right) \quad (4.50)$$

Dado que G es una matriz diagonal, $\det(\int G^2 dt) = \prod_{i=1}^r \int G_i^2 dt > 0$, y

$$\begin{aligned} \det \left(\int G^2 u^2 dt - \int G^2 u dt \left(\int G^2 dt \right)^{-1} \int G^2 u dt \right) &= \\ \prod_{i=1}^r \int G_i^2 u^2 dt - \int G_i^2 u dt \left(\int G_i^2 dt \right)^{-1} \int G_i^2 u dt &= \\ \prod_{i=1}^r \frac{\int G_i^2 u^2 dt \int G_i^2 dt - \left(\int G_i^2 u dt \right)^2}{\left(\int G_i^2 dt \right)} & \end{aligned}$$

De la desigualdad de Cauchy-Schwartz, es sabido que para funciones escalares integrables $f(x)$ y $g(x)$ linealmente independientes se cumple

$$\left(\int f(x)g(x)dx \right)^2 < \left(\int f^2(x)dx \right) \left(\int g^2(x)dx \right). \quad (4.51)$$

Finalmente, al igualar $f(t) = G_i(t)$ y $g(t) = G_i(t)u(t)$, obtenemos $\int_0^T G_i^2 u^2 dt \int_0^T G_i^2 dt > \int_0^T G_i^2 u dt \int_0^T G_i^2 u dt$. De este modo, si $u(t) \neq 0$ es linealmente independiente de y y $F_i \neq 0$ para $[t, t+T]$, $i = 1, \dots, r$, entonces $\det(M) > 0$. En consecuencia, existe $\alpha \in \mathfrak{R}^+$ tal que $M \geq \alpha I$. \square

4.2.1.1. Regresor como función de y y u

En este caso, definimos las reglas difusas (4.9) como

$$\begin{aligned} R_i &: \text{ Si } y \text{ es } F_i \text{ y } u \text{ es } H_i \text{ entonces} \\ \hat{\eta}^i &= \bar{A}\eta + \begin{bmatrix} 0_{(i-1) \times 1} \\ \theta_{i1} + \theta_{i2}u + \theta_{i3}\rho(y) \\ 0_{r-i \times 1} \end{bmatrix}. \end{aligned} \quad (4.52)$$

En consecuencia, el regresor tiene la estructura

$$\varphi(y, u) = [G \quad Gu \quad G\rho(y)] \in \mathfrak{R}^{n \times (2 + \dim(u))r}, \quad (4.53)$$

donde $G_{ii} = F_i H_i$, $G_{ij} = 0$, $\forall i \neq j$, $F_i = \exp(-\sigma_{y,i}^2 (y - \mu_{y,i})^2)$, $H_i = \exp(-\sigma_{u,i}^2 (u - \mu_{u,i})^2)$ y $\rho(y)$ es una función sigmoide de y . En este caso, el vector de parámetros es

$\theta_\eta \in \mathbb{R}^{3r}$. En forma similar a como se demostró en la sección 4.2.1, es posible demostrar, de manera similar al Lema 4.1, que (4.53) satisface a (4.25), siempre y cuando tanto las señales de entrada y salida sean linealmente independientes. En la siguiente sección se mostrarán algunas diferencias en el desempeño de los regresores hasta ahora descritos.

4.3. Ejemplos

Para validar e ilustrar el desempeño de la estructura propuesta, a continuación se muestran dos ejemplos de simulación. En el primero de ellos, el sistema objetivo a identificar es propiamente una red recurrente con la estructura (4.8). De este modo se muestra el desempeño del algoritmo y de la estructura ante un sistema donde no existe error de modelado teórico.

Por su parte, el segundo ejemplo ilustra el desempeño de la red en la identificación de un sistema no lineal, donde existe un error de modelado, que no es linealmente parametrizable y por lo tanto no puede ser directamente identificado por el observador adaptable propuesto por Marino (1990). En este ejemplo se muestra también como una partición del espacio u - y permite una mejor identificación. En el capítulo siguiente se mostrarán aplicaciones de la red mostrada a sistemas físicos.

4.3.1. Identificación de una red neuronal recurrente

El sistema a identificar es una red recurrente SISO en la forma (4.8) con tres estados ($n = 3$), y el regresor (4.53). Las funciones de membresía se muestran en la Fig. 4.1, y sus parámetros son $\sigma_{y,i} = \sigma_{u,i} = 0,2$, $i = 1, 2, 3$, y $\mu_{y,1} = 1$, $\mu_{u,1} = 1$, $\mu_{y,2} = -1$, $\mu_{u,2} = -1$, y $\mu_{y,3} = 0$, $\mu_{u,3} = 0$. El vector de parámetros θ_η es elegido en forma arbitraria como

$$\theta_\eta = [5,82 \quad 3,13 \quad 0,375 \quad 1,76 \quad -3,48 \quad 8,48 \quad 0,295 \quad 8,99 \quad 1,32]^T,$$

y

$$\bar{A} = \begin{bmatrix} -5 & 1 & 0 \\ -12 & 0 & 1 \\ -8 & 0 & 0 \end{bmatrix}, \quad C = [1 \quad 0 \quad 0]. \quad (4.54)$$

El vector K se elige de forma tal que los valores característicos de la matriz $A - KC$ sean $\lambda\{\bar{A} - KC\} = \{-16, -20, -24, -38\}$, con $\Gamma = 10^7 I_{3r \times 3r}$. En este sentido, el valor

de K se diseña de modo que el identificador converja a la salida real de la planta tan rápido como sea posible, mientras el error se amplifica por Γ para adaptar $\hat{\theta}_\eta$. La señal de entrada u es elegida como aleatoria acotada con media cero y varianza unitaria, con cambio de nivel cada 5 segundos.

En la Fig. 4.2 se muestra la evolución de los parámetros de la red durante el entrenamiento, donde se observa cómo estos convergen a sus valores reales. Por su parte, en la Fig. 4.3 se observa cómo la evolución de los estados de la red ya entrenada en paralelo con aquellos de la original es prácticamente igual bajo una misma señal de entrada, ya que los parámetros obtenidos al final del entrenamiento son muy cercanos a los originales:

$$\hat{\theta}_\eta = [5,83 \quad 3,10 \quad 0,371 \quad 1,77 \quad -3,49 \quad 8,48 \quad 0,292 \quad 8,96 \quad 1,32]^T.$$

En la Fig. 4.4 se muestra la salida de ambas redes (la original y la entrenada). Nuevamente se puede apreciar cómo la segunda logra identificar y modelar correctamente el desempeño dinámico de la primera.

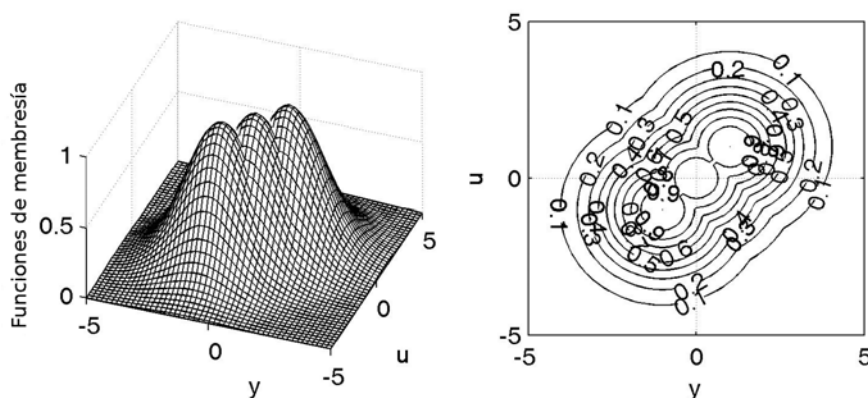


Figura 4.1: Identificación ideal: Funciones de membresía

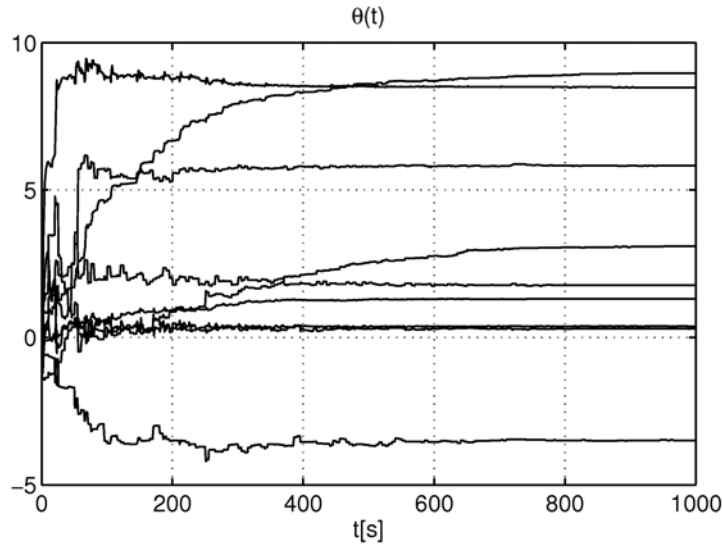


Figura 4.2: Identificación ideal: Evolución de los parámetros durante el entrenamiento

4.3.2. Identificación de un sistema no lineal

El sistema está descrito por las ecuaciones de estado no lineales

$$\begin{aligned} \dot{\mathbf{z}} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{z} + \begin{bmatrix} ae^{by} \\ 0 \\ cu \end{bmatrix} - \begin{bmatrix} 6 \\ 11 \\ 6 \end{bmatrix} [1 \ 0 \ 0] y, \\ y &= [1 \ 0 \ 0] \mathbf{z}. \end{aligned}$$

Debe notarse que este sistema no es linealmente parametrizable y cumple con las condiciones de observabilidad mostradas en Marino & Tomei (1993), con el par (A, C) observable. En la simulación se toman los valores $a = b = c = 1$. La red es elegida con $\dim\{\eta\} = 3 = n$, y

$$A = \begin{bmatrix} -5/2 & 1 & 0 \\ -3 & 0 & 1 \\ -1 & 0 & 0 \end{bmatrix}, \quad C = [1 \ 0 \ 0].$$

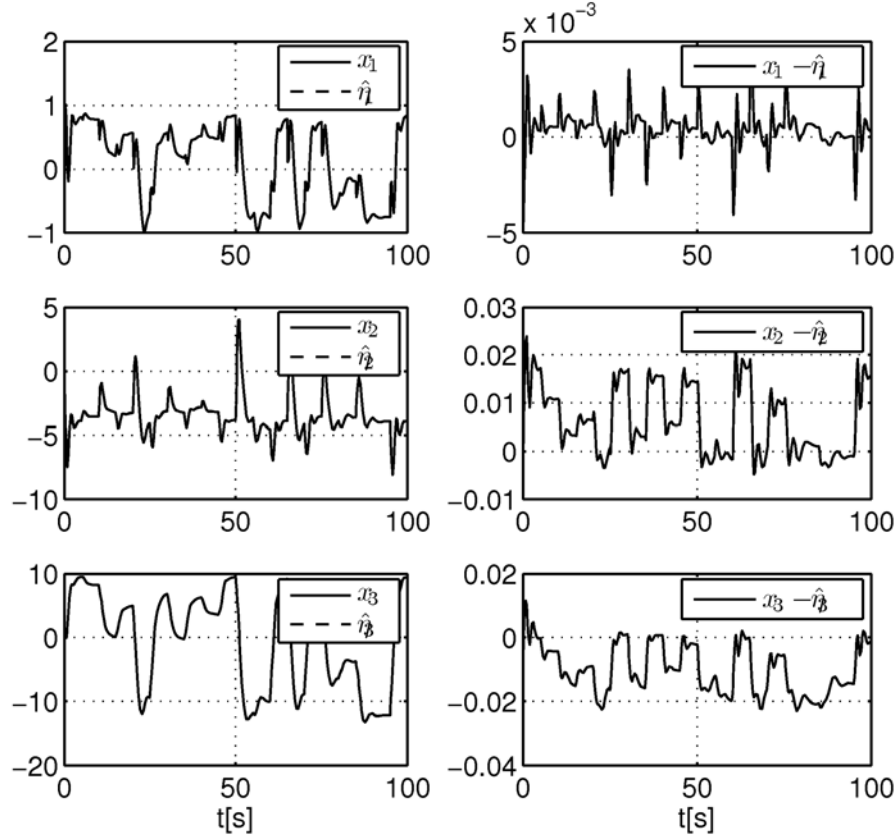


Figura 4.3: Identificación ideal: Comparación de los estados de ambos sistemas ante una misma entrada.

4.3.2.1. Identificación con partición del espacio en y y u

Para esta simulación, el regresor empleado es aquel descrito por (4.53) y los parámetros antecedentes son obtenidos a partir del algoritmo Fuzzy C-Means, descrito en el Capítulo 2. La señal de entrada $u(t)$ es una señal aleatoria que cambia cada $10s$, con media cero y varianza unitaria, filtrada por un sistema de primer orden paso-bajas con frecuencia de corte en $\omega = 5 \frac{rad}{s}$ y ganancia unitaria. El vector K es elegido de modo que los valores característicos de la matriz $A - KC$ sean $\{-60, -90 \pm 30i\}$, y $\Gamma = 10^{11} I_{3r \times 3r}$. Los parámetros obtenidos tras aplicar este método de agrupamiento son $\sigma_{y,1} = \sigma_{u,1} = 0,5$, $\sigma_{y,i} = \sigma_{u,i} = 0,2$, $i = 2, 3$, and $\mu_{y,1} = -0,7328$, $\mu_{u,1} = 0,0529$, $\mu_{y,2} = -0,3871$, $\mu_{u,2} = 0,2644$, $\mu_{y,3} = -1,1873$, $\mu_{u,3} = -0,1547$. La parte antecedente

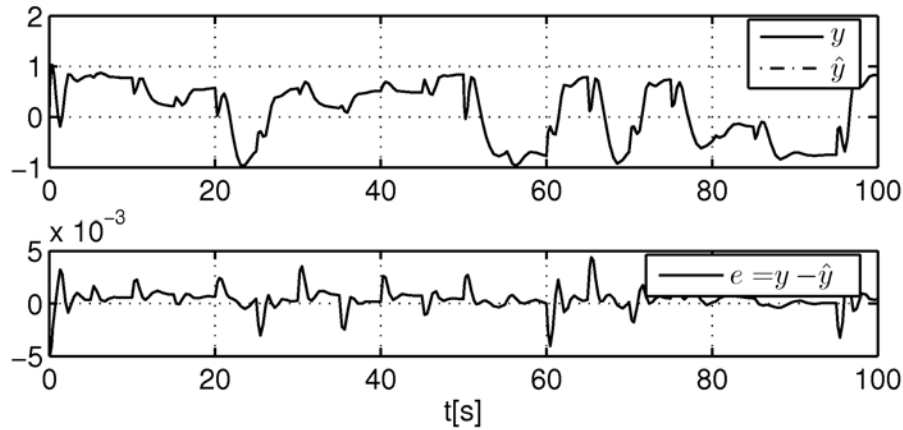


Figura 4.4: Identificación ideal: Comparación de la salida de ambos sistemas ante una misma entrada. Señales de salida y error.

definida de este modo es mostrada en la Fig. 4.5. Tras el entrenamiento, se obtuvo el vector de parámetros

$$\hat{\theta}_\eta = [-1,73 \quad -8,24 \quad -1,48 \quad 1,45 \quad 4,83 \quad 3,12 \quad -0,615 \quad -4,13 \quad -0,906]^T.$$

La evolución de los parámetros durante el entrenamiento se muestra en la Fig. 4.6a, donde se observa cómo los estados convergen, mientras que en la Fig. 4.7a se muestra el desempeño de la red en paralelo con el sistema original. Igualmente, en la Fig. se muestran los estados de ambos sistemas ante una misma entrada, y el la Fig. 4.8a se muestra la salida de ambos sistemas, con un error RMS de 0,1228. Se puede notar que, pese a que la evolución de los estados de ambos sistemas no es la misma, la respuesta entrada-salida de la red entrenada emula aquella del sistema original, como se esperaba.

4.3.2.2. Identificación con partición del espacio en y

Ahora, el regresor (4.43) es empleado para comparar el desempeño de ambos sistemas, así como los mismos parámetros obtenidos en el caso anterior para $\sigma_{y,i}$ y $\mu_{y,i}$. Tras el entrenamiento, el siguiente conjunto de parámetros es obtenido:

$$\hat{\theta}_\eta = [0,400 \quad -8,40 \quad -0,851 \quad 1,06 \quad 4,0 \quad 2,23].$$

La evolución de estos parámetros se muestra en la Fig. 4.6b, mientras que los estados y salida de la red se muestran en la Fig. 4.7b y Fig.4.8b respectivamente, con un error

RMS de 0,1660. Se puede observar que una partición más fina del espacio (como en el caso anterior), permite una mejor aproximación del comportamiento dinámico del sistema, como se podía esperar.

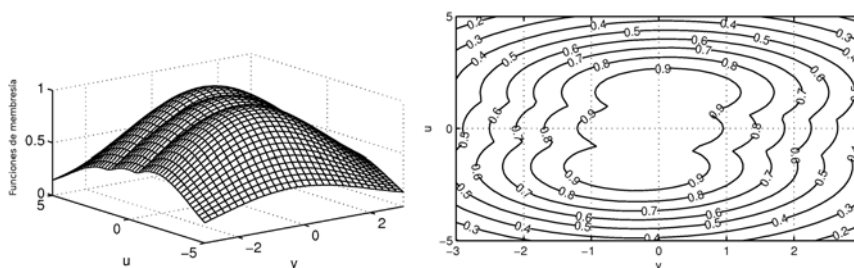


Figura 4.5: Identificación de un sistema no lineal: Funciones de membresía. Regresor dependiente de a) $\varphi = \varphi(y, u)$, b) $\varphi = \varphi(u)$

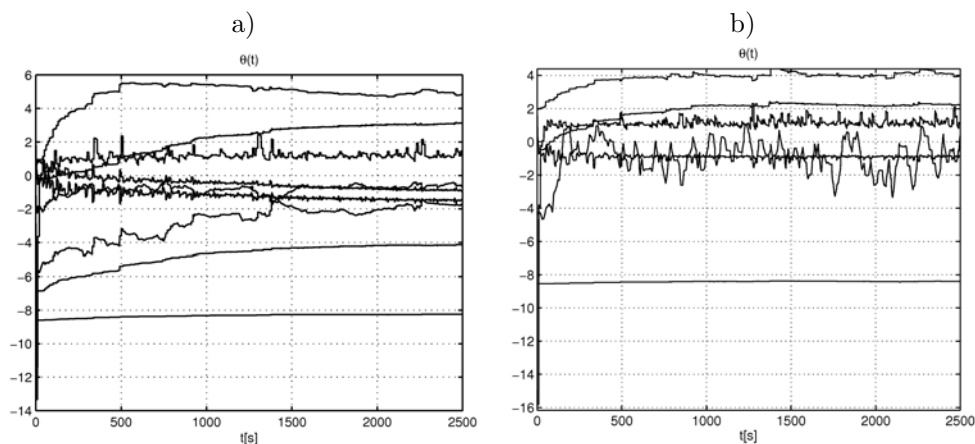


Figura 4.6: Identificación de un sistema no lineal: Comparación de la evolución de los parámetros. Regresor dependiente de a) $\varphi = \varphi(y, u)$, b) $\varphi = \varphi(u)$

4.3.3. Identificación de un sistema en forma de realimentación de salida

Consideremos ahora el caso en el cual se cuenta con un sistema en la forma (4.2). En este caso se analiza el efecto del cambio de los parámetros antecedentes en la calidad de la identificación al existir un error de aproximación inherente a la red.

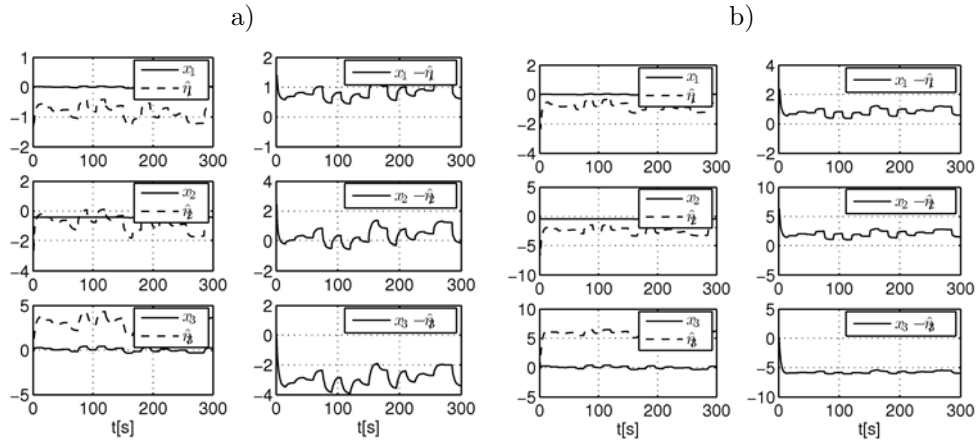


Figura 4.7: Identificación de un sistema no lineal: Comparación de la evolución de los estados. Regresor dependiente de a) $\varphi = \varphi(y, u)$, b) $\varphi = \varphi(u)$

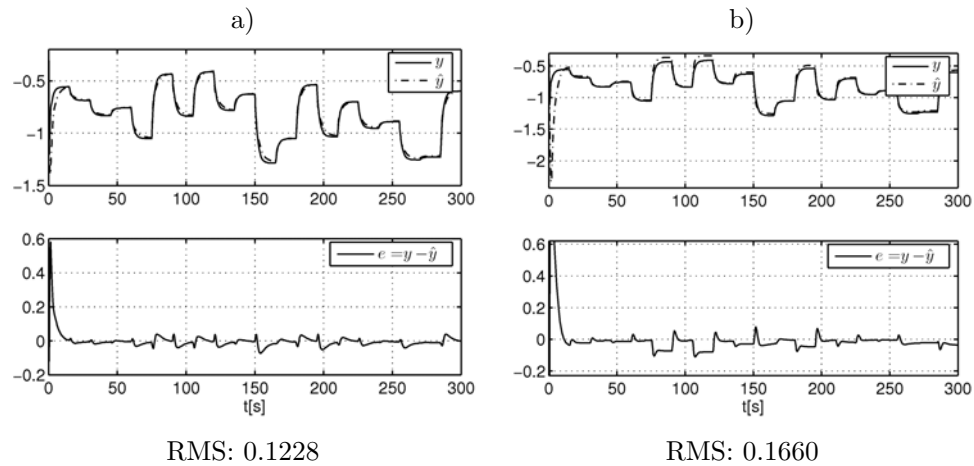


Figura 4.8: Identificación de un sistema no lineal: Comparación de las señales de salida. Regresor dependiente de a) $\varphi = \varphi(y, u)$, b) $\varphi = \varphi(u)$

El sistema a identificar está dado por

$$\dot{\mathbf{z}} = \begin{bmatrix} 0 & 0 & -8 \\ 1 & 0 & -12 \\ 0 & 1 & -5 \end{bmatrix} \mathbf{z} + \begin{bmatrix} 0,1 \\ 0 \\ -0,1 \sin(\frac{1}{2}y) \end{bmatrix} + \begin{bmatrix} 0,1 \sin(\frac{1}{2}y) + 0,2 \\ 0,1 \\ 0 \end{bmatrix} u,$$

$$y = [1 \ 0 \ 0] \mathbf{z}.$$

El par (A, C) es observable, y para la estructura de la red se elige $\dim\{\eta\} = 3 = n$. Para esta simulación, elegimos los mismos parámetros que los empleados en el ejemplo anterior: esto es, $\sigma_{y,i} = 0,2$, $i = 1, 2, 3$, and $\mu_{y,1} = -1$, $\mu_{y,2} = 0$ y $\mu_{y,3} = 1$. El vector K se elige de forma que los valores característicos de $A - KC$ sean $\{-40, -50, -60\}$, y $\Gamma = 10^7 I_{3r \times 3r}$. En la Fig. 4.9.a se muestra la evolución de los parámetros durante el entrenamiento, mientras que en la Fig. 4.9.b se muestra el desempeño de la red al simularse en paralelo con respecto al sistema original. Se puede observar que la salida de la red (en la Fig. 4.10) es muy cercana con respecto a la del sistema original, mientras que los estados capturan la dinámica (con un error de estado estable, en particular en el caso de x_3 y z_3 , que afecta en forma mínima al error de identificación $y - \hat{y}$).

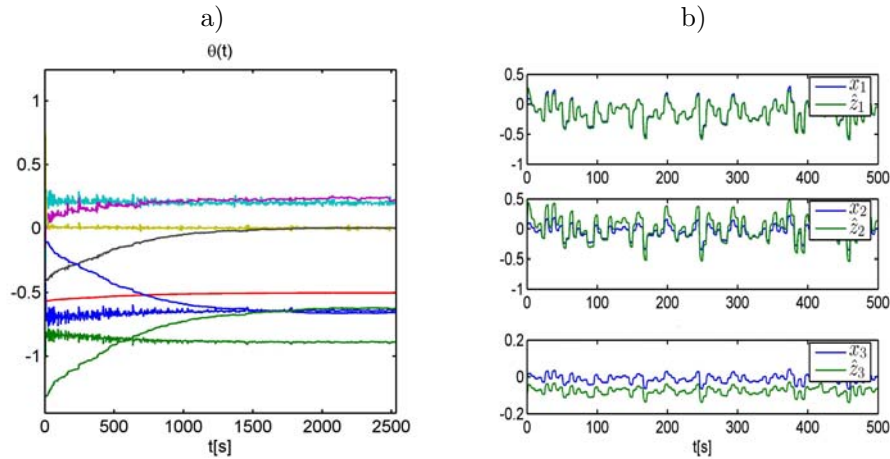


Figura 4.9: Sistema no lineal en realimentación de salida: a) Evolución de los parámetros y b) estados

Con el fin de analizar la forma en que la elección de los parámetros antecedentes afecta al desempeño de la red, se pueden llevar a cabo varias simulaciones con diversas elecciones de estos. En este ejemplo, se observa que con $\sigma_{y,i}$ situado entre 0.2 y 0.1 los resultados no tienen una variación importante. Sin embargo, cuando $\sigma_{y,i} > 0,5$, el desempeño de la red se deteriora. En la Fig. 4.11.a se muestra el resultado de simulación cuando se elige $\sigma_{y,i} = 0,5$, y se puede notar que el desempeño de la red decae con respecto a la selección anterior. No obstante, debe notarse que los estados y parámetros permanecen acotados, como se muestra en la Fig. 4.11.b). Esto indica que es necesario

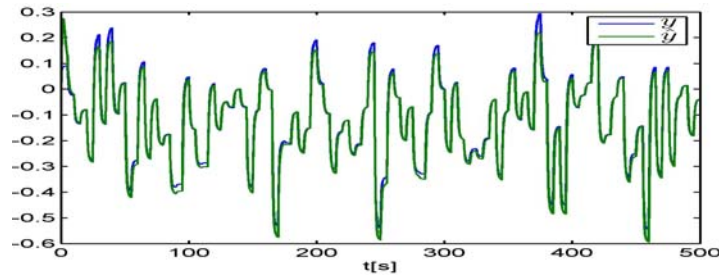


Figura 4.10: Sistema no lineal en realimentación de salida: Comparación de la salida

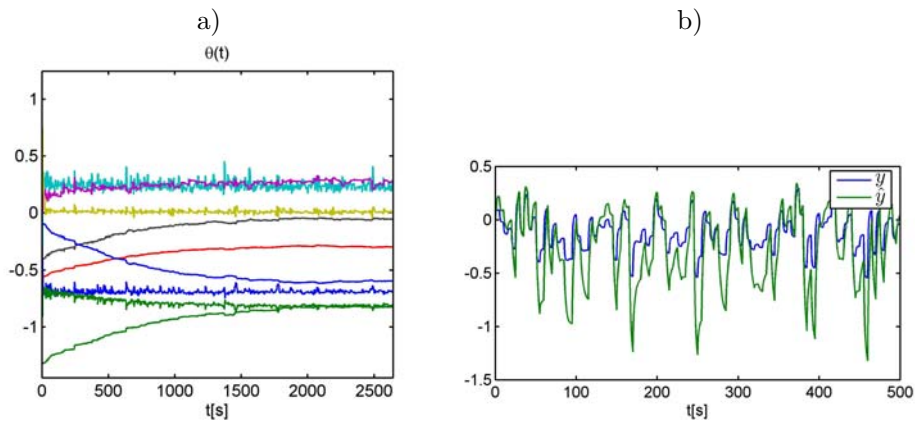


Figura 4.11: Sistema no lineal en realimentación de salida. $\sigma_i = 0,5$. a) Evolución de los parámetros, b) Comparación de la salida

tener en cuenta a la parte antecedente (y consecuentemente, la partición del espacio) antes de aplicar el algoritmo de entrenamiento.

4.4. Control basado en la red identificada: Caso particular

Consideremos ahora un caso particular en el cual es posible diseñar un esquema de control para la red propuesta, al emplearse a la red identificada como un modelo del sistema, como se muestra en la Fig. 4.12.

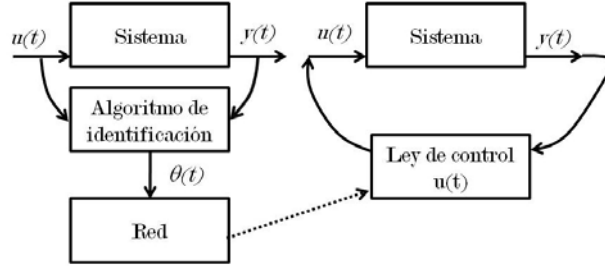


Figura 4.12: Esquema de control mediante la red identificada

Para tal efecto, tómesese el siguiente sistema una-entrada-una-salida afín en el control

$$\begin{aligned}\dot{\mathbf{x}}(t) &= f(\mathbf{x}) + g(\mathbf{x})u, \\ y(t) &= h(\mathbf{x}),\end{aligned}\tag{4.55}$$

donde $u(t)$ y $y(t)$ son las señales de entrada y salida, respectivamente, $\mathbf{x} \in \mathfrak{R}^n$ es el vector de estados del sistema, y $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$, $g : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$, $h : \mathfrak{R}^n \rightarrow \mathfrak{R}$ son funciones suaves que cumplen con $f(0) = 0$, $g(x) \neq 0$, $\forall x \in \mathfrak{R}^n$. Asíumase que solo es posible medir la salida, no así los estados \mathbf{x} , del sistema.

Dado este sistema, los principales objetivos son:

1. Modelar al sistema mediante una red neurodifusa recurrente, similar a (4.8), e identificar los parámetros de ésta de modo que el mapeo $u \rightarrow \hat{y}$ cumpla con hacer a $\sup_{t \geq 0} |\hat{y}(t) - y(t)|$ tan pequeño como sea posible.
2. Diseñar, con base en el modelo obtenido por una red neurodifusa recurrente, un esquema de control realimentado para el sistema original, tal que permita que la planta siga a una cierta referencia y_r sin necesidad de medir directamente los estados \mathbf{x} .

Para este propósito, se asume que el grado relativo ρ del sistema no lineal (4.55) es conocido, y que el sistema se encuentra además representado, o es transformable, en la forma

$$\begin{aligned}\dot{\zeta} &= A_c \zeta + \mathbf{b} \sigma(y)u + \psi(y), \quad \zeta \in \mathfrak{R}^n, \\ y &= C_c \zeta,\end{aligned}\tag{4.56}$$

donde $\mathbf{b} = [0 \ \dots \ 0 \ b_\rho \ \dots \ b_n]^T$ es un vector Hurwitz (esto es, el polinomio asociado con el vector es Hurwitz), $\sigma(y) : \mathfrak{R} \rightarrow \mathfrak{R}$ es una función suave, y $\psi = \begin{bmatrix} 0_{(\rho-1) \times 1} \\ \bar{\psi}_{(n-\rho+1) \times 1} \end{bmatrix}$, con

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \quad C_c = [1 \ 0 \ 0 \ \dots \ 0]. \quad (4.57)$$

Las condiciones necesarias y suficientes que el sistema debe cumplir para que, mediante un difeomorfismo local $\zeta = T(\mathbf{x})$ el sistema (4.1) se transforme en (4.56), se encuentran reportadas por Marino (1990); Marino & Tomei (1993) y se señalan en el Apéndice. El tipo de sistemas que cumplen con dicha condición son aquellos que son *linealizables mediante una realimentación dinámica de la salida*. Para fines de presentación, nos centraremos en el caso donde el grado relativo es unitario (esto es, $\rho = 1$).

De este modo, el modelo propuesto para identificar al sistema (4.55) es la red recurrente en tiempo continuo

$$\begin{aligned} \dot{\eta} &= A_c \eta + \mathbf{b} \varphi_u^T(y) \theta_u u(t) + \varphi_y(y) \theta_y \\ y &= C \eta, \end{aligned} \quad (4.58)$$

donde $\varphi_u(y) = [\varphi_{u1} \ \dots \ \varphi_{uq}]^T \in \mathfrak{R}^q$, $\theta_u \in \mathfrak{R}^q$, $\varphi_y(y) \in \mathfrak{R}^{n \times p}$, y $\theta_y \in \mathfrak{R}^p$. Es claro que en esta estructura los términos $\varphi_u^T(y) \theta_u$ y $\varphi_y(y) \theta_y$ son aproximaciones de $\sigma(y)$ y $\psi(y)$ respectivamente en (4.56), mientras que suponemos que el vector \mathbf{b} debe ser conocido de antemano a partir del análisis del sistema (4.55). Nuevamente, mediante el teorema de Stone-Weierstrass para redes neuronales (Wang, 1994), (4.56) se puede escribir mediante

$$\begin{aligned} \dot{\zeta} &= A_c \zeta + \mathbf{b} \varphi_u^T(y) \theta_u u(t) + \varphi_y(y) \theta_y + \varepsilon(t) \\ y &= C \zeta \end{aligned} \quad (4.59)$$

donde $\varepsilon(t)$ es el error de modelado de la red neuronal.

Reescribiendo los términos anteriores en (4.58), obtenemos

$$\begin{aligned}
 \dot{\eta} &= A_c \eta + \begin{bmatrix} b_1 \varphi_{u1}(y)u & b_1 \varphi_{u2}(y)u & \dots & b_1 \varphi_{uq}u \\ b_2 \varphi_{u1}(y)u & b_2 \varphi_{u2}(y)u & \dots & b_2 \varphi_{uq}u \\ \vdots & \vdots & \ddots & \vdots \\ b_n \varphi_{u1}(y)u & b_n \varphi_{u2}(y)u & \dots & b_n \varphi_{uq}u \end{bmatrix} \theta_u + \varphi_y \theta_y \\
 &:= A_c \eta + \varphi(y, u, \mathbf{b}) \theta \\
 y &= C_c \eta
 \end{aligned} \tag{4.60}$$

donde $\varphi(y, u, \mathbf{b}) = [\mathbf{b} \varphi_u^T u(t) \quad \varphi_y] \in \mathbb{R}^{n \times (q+p)}$ y $\theta = \begin{bmatrix} \theta_u \\ \theta_y \end{bmatrix} \in \mathbb{R}^{q+p}$.

Como en los casos analizados en las secciones 4.1 y 4.2, es posible generar una aproximación de θ mediante una aproximación de η mediante un algoritmo desacoplado, siempre y cuando se cumplan ciertas condiciones de excitación persistente. Dado que se pueden elegir varias definiciones para φ_u y φ_y , proponemos una aproximación mediante un esquema neurodifuso.

4.4.1. Estructura neurodifusa para control.

De este modo, proponemos una estructura basada en sistemas neurodifusos descrita por las siguientes r reglas, muy similar a aquella propuesta en (4.43):

$$NN : \left\{ \begin{array}{l} \text{Si } y \text{ es } F_1 \text{ entonces } \dot{\eta}^1 = A_c \eta + \mathbf{b} \theta_{u1} + \begin{bmatrix} \theta_{y11} \\ \theta_{y12} \\ \vdots \\ \theta_{y1n} \end{bmatrix} \delta(y), \\ \vdots \\ \text{Si } y \text{ es } F_r \text{ entonces } \dot{\eta}^r = A_c \eta + \mathbf{b} \theta_{ur} + \begin{bmatrix} \theta_{yr1} \\ \theta_{yr2} \\ \vdots \\ \theta_{yrn} \end{bmatrix} \delta(y), \end{array} \right.$$

donde nuevamente, F_i se proponen como funciones de membresía gaussianas $F_i(y) = \exp(-\sigma_{y,i}^2(y - \mu_{y,i})^2)$, con la desfusión como una suma ponderada

$$\dot{\eta} = \sum_{i=1}^r F_i \dot{\eta}^i / \sum_{i=1}^r F_i. \tag{4.61}$$

De este modo, los regresores φ_u y φ_y se pueden expresar como

$$\varphi_u = [F_1 u \quad \dots \quad F_r u] / \sum_{i=1}^r F_i, \quad (4.62)$$

$$\varphi_y = [F_1 I_{n \times n} \quad \dots \quad F_r I_{n \times n}] \delta(y) / \sum_{i=1}^r F_i \quad (4.63)$$

con $\delta(y)$ una función sigmoide suave, monotónicamente creciente y

$$\lim_{y \rightarrow \pm\infty} \delta(y) = \pm 1, \quad \delta(0) = 0.$$

Por lo tanto, los vectores de parámetros θ_u y θ_y se pueden definir mediante

$$\theta_u = [\theta_{u1} \quad \dots \quad \theta_{ur}]^T, \quad (4.64)$$

$$\theta_y = [\theta_{y11} \quad \dots \quad \theta_{y1n} \quad \theta_{y21} \quad \dots \quad \theta_{y2n} \quad \dots \quad \theta_{yr1} \quad \dots \quad \theta_{yrm}]^T, \quad (4.65)$$

y $\theta \in \mathbb{R}^{nr+r}$.

4.4.2. Diseño de un control de seguimiento por realimentación de salida basado en el modelo neurodifuso

Una vez que el proceso de identificación ha sido llevado a cabo, es posible implementar una ley de control de seguimiento, ya que la propia estructura de la red (4.58) garantiza (para $\varphi_u \neq 0$) que es posible estabilizarla por una realimentación de salida.

Con el fin de diseñar la ley de control, en primer lugar se emplea el diseño de un control dinámico por realimentación de salida en el caso ideal en que las funciones $\sigma(y)$ $\psi(y)$ en (4.56) sean conocidas (Marino & Tomei, 1996). Con base en este resultado, se mostrará el diseño empleando el modelo (4.59) por red neurodifusa mediante un control por equivalencia cierta (Wang, 1994).

4.4.2.1. Diseño de control con σ conocida.

En el caso $\rho = 1$, se define una *dinámica de referencia*

$$\dot{\eta}_r = \Gamma \eta_r + \beta y_r + \nu(y_r) \quad (4.66)$$

donde $\dim \eta_r = n - \rho = n - 1$, y_r es la trayectoria deseada y

$$\Gamma = \begin{bmatrix} -\frac{b_2}{b_1} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{b_{n-1}}{b_1} & 0 & \dots & 1 \\ -\frac{b_n}{b_1} & 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{(n-\rho) \times (n-\rho)}, \quad \bar{\psi}(0) = 0.$$

El control que logra el seguimiento exponencial (Marino & Tomei, 1996) se puede demostrar que está dado por

$$u^*(t) = \frac{1}{\sigma(y)} (-b_2 y - \psi_1(y) + \dot{y}_r - \eta_{r1} - k e - e (\beta^T P^2 \beta + \gamma^T(y_r, e) P^2 \gamma(y_r, e))) \quad (4.67)$$

donde P es la solución de la ecuación de Lyapunov $\Gamma^T P + P \Gamma = -3I$, k con una constante positiva y

$$e = y - y_r, \quad (4.68)$$

$$\beta = \begin{bmatrix} b_3 - b_2^2 \\ b_4 - b_3 b_2 \\ \vdots \\ b_n - b_{n-1} b_2 \\ -b_2 b_2 \end{bmatrix}, \quad (4.69)$$

$$\nu(y) = \begin{bmatrix} \psi_2 - d_2 \psi_1 \\ \psi_3 - d_3 \psi_1 \\ \vdots \\ \psi_{n-1} - d_{n-1} \psi_1 \\ \psi_n - d_n \psi_1 \end{bmatrix}. \quad (4.70)$$

$\gamma(y_r, e)$ se calcula mediante

$$e\gamma(y_r, e) = \tilde{\nu}(y_r, e) = \nu(y) - \nu(y_r) = \nu(e + y_r) - \nu(y_r). \quad (4.71)$$

Dado que $\nu_i(y)$ es, para la red propuesta, $\nu_i(y) = \psi_{i+1} - d_{i+1} \psi_i$, $i \in [1, n - \rho]$, entonces, de (4.8) tenemos que $\psi = \varphi_y \theta_y$, así que

$$\psi_j = \delta(y) \sum_{j=1}^r \varphi_{yj} \theta_{yj} \quad (4.72)$$

$$\begin{aligned} \nu_i(y) &= \psi_{i+1} - d_{i+1} \psi_1, \\ &= \delta(y) \left(\sum_{j=1}^r \varphi_{yj} \theta_{y(i+1)j} - d_{i+1} \sum_{j=1}^r \varphi_{yj} \theta_{y1j} \right), \\ &= \delta(y) \underbrace{\left(\sum_{j=1}^r \varphi_{yj} (\theta_{y(i+1)j} - d_{i+1} \theta_{y1j}) \right)}_{\chi_i(y)}, \\ &= \delta(y) \chi_i(y). \end{aligned} \quad (4.73)$$

la cual es una función suave. En este caso se puede escribir $\tilde{\nu}_i(y_r, e) = e\gamma_i(e, y_r)$, donde $\gamma_i(e, y_r)$ es también una función suave en la forma

$$\gamma(e, y_r) = \begin{cases} \frac{\nu_y(y_r+e) - \nu_i(y_r)}{e}, & e \neq 0 \\ \left. \frac{\partial \nu_i(x)}{\partial x} \right|_{x=0} & e = 0 \end{cases} \quad (4.74)$$

4.4.3. Diseño de control con σ desconocida y \mathbf{b} conocida.

Dado que la red (4.8) solo puede aproximar el valor de $\sigma(y)$ en (4.56) y (4.59), el control debe tomar en cuenta al error de modelado ε , compuesto por los errores de identificación

$$\tilde{\sigma} = \sigma - \varphi_u^T(y, u)\theta_u, \quad (4.75)$$

$$\tilde{\psi} = \psi - \varphi_y(y, u)\theta_y \quad (4.76)$$

De esta forma, el control para este sistema está basado en (4.67), y es

$$u(t) = \frac{1}{\varphi_u^T(y, u)\theta_u} \left(-b_2 y - \hat{\psi}_1(y) + \dot{y}_r - \eta_{r1} - k e - e (\beta^T P^2 \beta + \gamma^T(y_r, e) P^2 \gamma(y_r, e)) \right) \quad (4.77)$$

donde $\hat{\psi} = \varphi_y(y, u)\theta_y$.

Nota 4.4. *Este esquema de control se basa completamente en el modelo identificado, sus parámetros, funciones y la señal de referencia, sin ser necesario diseñar un observador sobre los estados η . Dado que está basado en una aproximación del control ideal $u^*(t)$, solo puede esperarse que existan errores de convergencia hacia una bola alrededor del origen, debido a los errores de identificación $\tilde{\sigma} = \sigma - \varphi_u^T \theta_u$ y $\tilde{\psi} = \psi - \varphi_y \theta_y$. Nótese que el procedimiento es fuertemente dependiente del conocimiento de \mathbf{b} , el cual es parte del sistema real (no el identificado), lo cual es una limitante del esquema propuesto.*

4.4.4. Ejemplo

Para probar el esquema de control presentado, se lleva a cabo la identificación y control de un sistema no lineal, el cual está dado por las ecuaciones

$$\begin{aligned} \dot{x}_1 &= -x_1^3 + (1 + x_1^2)u(t), \\ \dot{x}_2 &= -x_1 - x_2^3, \\ y &= x_1. \end{aligned}$$

La red considera dos estados y cuatro reglas ($n = 2, r = 4$), con $\sigma_{y,i} = 2, i = 1, \dots, 4$ y $\mu_y = \{-1, -1/2, 1/2, 1\}$, $\Gamma = 2000I$, $K = [30, 200]^T$, $\delta(y) = \tanh(y)$. Se puede determinar que $\mathbf{b} = [1 \ 1]^T$ (el cual es un vector Hurwitz).

La evolución de los parámetros durante el entrenamiento se muestra en la Fig. 4.13. La salida para ambos sistemas funcionando en paralelo se muestra en la Fig. 4.14, y se puede comprobar cómo su desempeño entrada/saliuda es muy similar. En ambos casos, el error de estimación paramétrica muestra una cota final (como anteriormente se analizó), con un bajo error de modelado.

Tras el entrenamiento, los parámetros obtenidos son (redondeados, por propósitos de presentación):

$$\begin{aligned} \theta_u &= [1,13 \ 0,85 \ 0,87 \ 1,09]^T \\ \theta_y &= [-1,79 \ -0,88 \ -0,42 \ 0,0035... \\ &\quad -0,42 \ -0,0088 \ -1,77 \ -0,861]^T. \end{aligned}$$

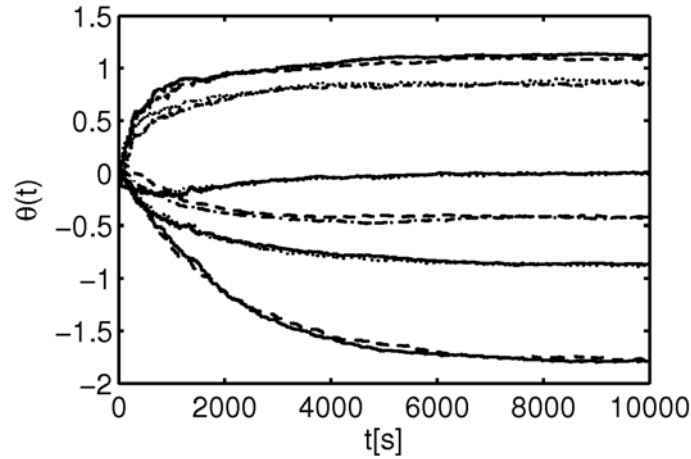


Figura 4.13: Sistema no lineal: Evolución de parámetros en el entrenamiento

Una vez que la planta ha sido modelada e identificada por la red, el control propuesto (4.77) se diseña basado en el modelo por red neurodifusa recurrente y se implementa en la planta no lineal. Es interesante el notar que $\varphi_u^T \theta_u \neq 0$, así que el control no queda indeterminado.

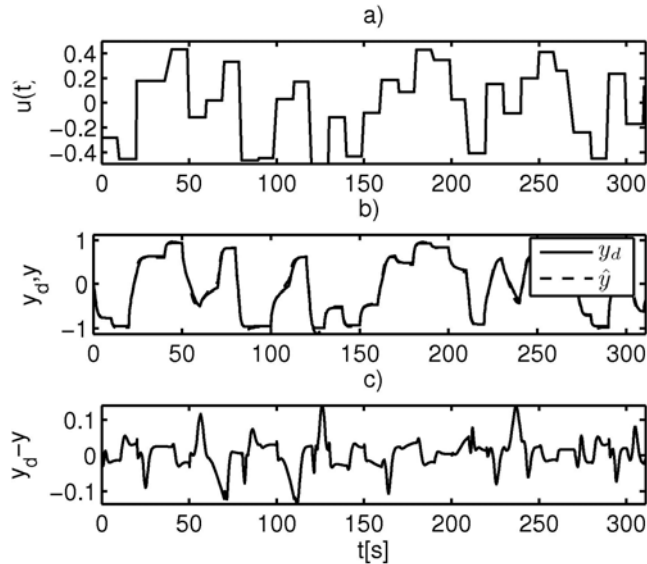


Figura 4.14: Sistema no lineal: Resultados de la identificación. a) Señal de excitación b) Salida de ambos sistemas c) Error.

Los resultados de simulación se muestran en la Fig. 4.15. Nótese que el error de seguimiento se encuentra acotado.

Se puede entonces observar que con la red neurodifusa recurrente que modela al sistema, es posible establecer un control de seguimiento para el mismo sistema basado solo en la estructura propia de la red y en cierto conocimiento del sistema original. Nótese que el control no requiere del conocimiento directo ni de los estados, sino solo de la salida. Sin embargo, el conocimiento del vector \mathbf{b} puede ser de difícil acceso, por lo que es necesario explorar otras técnicas de control basadas en la red ya identificada, suponiéndola como un modelo *confiable* del sistema.

4.5. Estructura de la red: Tiempo discreto

En esta sección se desarrolla en forma paralela la versión discreta de la red y el algoritmo de entrenamiento analizados en la sección 4.1. Una versión discreta de estos algoritmos permite una implementación más sencilla en sistemas de cómputo digitales (como microcontroladores o microprocesadores) en los que sea poco factible resolver ecuaciones diferenciales en tiempo real. En lugar de recurrir a una discretización directa

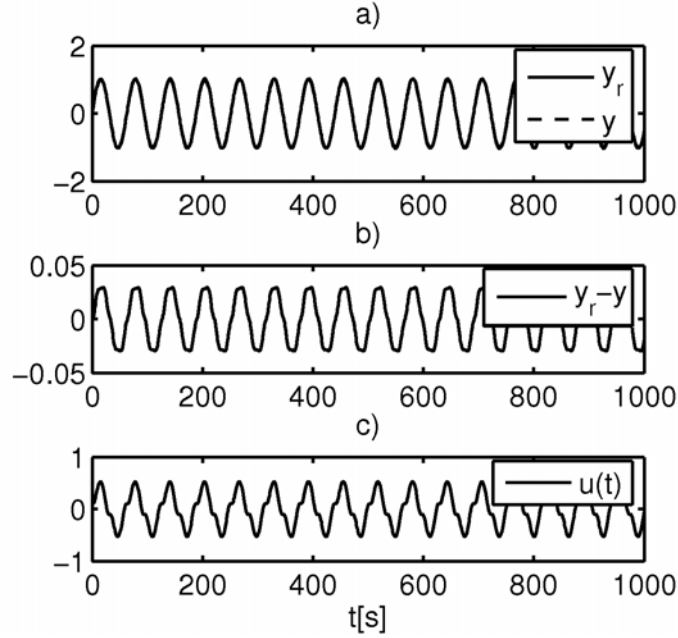


Figura 4.15: Sistema no lineal: Control a) Señal de salida y referencia b) Error c) Señal de control.

de las ecuaciones obtenidas mediante una aproximación de la derivada, se considera el mismo problema partiendo de una representación discreta del sistema (4.1).

De esta forma, considérese el sistema SISO en tiempo discreto

$$\begin{aligned}\mathbf{x}[k+1] &= f(\mathbf{x}[k], u[k]), \\ y[k] &= h(\mathbf{x}[k]),\end{aligned}\tag{4.78}$$

donde $u[k]$ y $y[k]$ son las secuencias de entrada y salida respectivamente, y $\mathbf{x} \in \mathfrak{R}^n$ es el vector de estados en el tiempo k , $f : \mathfrak{R}^n \times \mathfrak{R} \rightarrow \mathfrak{R}^n$, $h : \mathfrak{R}^n \rightarrow \mathfrak{R}$, son funciones suaves tales que la ecuación en diferencias tenga una respuesta única para cada secuencia de entrada. Así como en el caso de tiempo continuo, se asume que el sistema (4.78) es observable en la región de interés y que pertenece a la clase de sistemas transformable a

$$\begin{aligned}\mathbf{z}[k+1] &= A\mathbf{z}[k] + \bar{f}(y[k], u[k]), \\ y &= C\mathbf{z},\end{aligned}\tag{4.79}$$

con (A, C) un par observable en forma canónica de observador, con $\bar{f}(y, u)$ una función suave.

El objetivo es nuevamente encontrar una red neurodifusa recurrente en tiempo discreto descrita por

$$\begin{aligned}\eta[k+1] &= A\eta[k] + \varphi(y[k], u[k])\theta_\eta, \\ y[k] &= C\eta,\end{aligned}\tag{4.80}$$

donde mediante la información sobre las secuencias $u[k]$ y salida $y[k]$ se genere un mapeo dinámico $u \rightarrow \hat{y}$, tal que se minimice $\sup_{t \geq 0} |\hat{y}[k] - y[k]|$ mediante la identificación de θ_η .

La red neurodifusa se define por las reglas difusas

$$R_i : \text{ Si } y[k] \text{ es } F_i \text{ y } u[k] \text{ es } H_i \text{ entonces}$$

$$\eta^1[k+1] = A\eta[k] + \begin{bmatrix} \theta_{u11} \\ \theta_{u12} \\ \vdots \\ \theta_{u1n} \end{bmatrix} u[k] + \begin{bmatrix} \theta_{y11} \\ \theta_{y12} \\ \vdots \\ \theta_{y1n} \end{bmatrix} \rho(y[h]).\tag{4.81}$$

con $\rho : \mathfrak{R} \rightarrow \mathfrak{R}$, $\rho = \rho(y)$ una función sigmoideal suave, con los conjuntos difusos definidos como (4.39). En forma similar que en caso de tiempo continuo, el problema se puede resolver mediante un observador en tiempo discreto inspirado en observadores adaptables en tiempo discreto (Combastel *et al.*, 2006), donde con base en las ecuaciones en diferencias

$$\Upsilon[k+1] = (A - KC)\Upsilon[k] + \varphi[k],\tag{4.82}$$

$$\hat{\theta}[k+1] = \hat{\theta}[k] + \mu[k]\Upsilon[k]^T C^T (y[k] - C\hat{\eta}[k]),\tag{4.83}$$

$$\hat{\eta}[k+1] = (A - KC)\hat{\eta}[k] + \varphi[k]\hat{\theta}[k] + Ky[k] + \Upsilon[k+1](\Delta\hat{\theta}[k]),\tag{4.84}$$

donde si el regresor $\varphi[k]$ cumple con la condición de excitación persistente con el filtrado $\Upsilon[k]$ satisface la desigualdad

$$\frac{1}{L} \sum_{i=k}^{k-L-1} \mu[i]\Upsilon_i^T[i]C^T C\Upsilon[i] \geq \alpha I,\tag{4.85}$$

el error de identificación y estimación permanece acotado. Cabe mencionar que tanto las ecuaciones de entrenamiento como la condición de excitación persistente en tiempo

discreto son equivalentes a aquellas necesarias para la operación en tiempo continuo, así como el procedimiento empleado para la demostración.

4.6. Comentarios

En este capítulo se presenta una estructura para modelado e identificación de sistemas mediante una red neurodifusa recurrente en tiempo continuo. La estructura y el algoritmo de entrenamiento requieren únicamente de las mediciones de las señales entrada/salida del sistema a modelar, así como de un cierto conocimiento previo de la estructura del mismo (número de estados, condiciones de observabilidad). Se mostraron diversas estructuras particulares capaces de llevar a cabo el modelado del mismo. El algoritmo de entrenamiento mostrado se basa en la teoría de observadores adaptables mediante un esquema desacoplado. Éste permite identificar los parámetros de la red neuronal recurrente y minimizar el error de identificación de la red. Con los ejemplos de simulación presentados se muestra cómo la red es capaz de identificar a varios sistemas no lineales representativos, así como el efecto que tiene la elección de los parámetros antecedentes. En el siguiente capítulo se muestran aplicaciones de la red a problemas de ingeniería reales.

4. IDENTIFICACIÓN MEDIANTE REDES RECURRENTE EN ESPACIO DE ESTADOS

Capítulo 5

Aplicaciones

En este capítulo se muestran diversos sistemas físicos reales en los cuales es posible obtener un modelo basado en las redes neurodifusas recurrentes presentadas en los capítulos anteriores, con el fin de mostrar la capacidad de aproximación de la red ante sistemas donde solo se tienen datos entrada-salida y cierta información estructural de los mismos.

Los ejemplos mostrados son seleccionados tanto por la importancia que presentan en problemas de ingeniería, la dificultad que presenta su modelado y para ilustrar la capacidad de identificación. En primer lugar, se muestra la identificación de dos sistemas ampliamente estudiados en la literatura: un motor de corriente directa con no linealidades asociadas (donde los datos son obtenidos de forma experimental) y un robot con de un solo eslabón flexible. A continuación se identifica un sistema de tráfico vehicular, el cual es identificado a partir de datos obtenidos en forma experimental de una autopista real. Se muestra la aplicación de esta misma red en el problema de identificación de un robot de dos grados de libertad en un sistema de servo-visual experimental y por último un sistema biológico, en el cual se considera la dinámica glucosa-insulina en pacientes diabéticos. En todos los casos se buscó que las señales generadas por el regresor satisficieran la condición de excitación persistente.

5.1. Motor de corriente directa

Para este ejemplo se considera un motor de DC con no linealidades asociadas, tales como fricción de Coulomb e inercia no lineal, el cual se muestra en la Fig. 5.1. El motor

se controla por voltaje, y se toman 400 segundos de señales entrada/salida para este experimento, con un tiempo de muestreo de 20ms.

Las señales empleadas para excitar al sistema se muestran en la Fig. 5.2. Para la red, se elige el regresor (4.43), con $n = 3$. Los parámetros antecedentes se obtienen mediante el algoritmo Fuzzy C-Means sobre los datos de salida y , y los valores generados son $\mu_1 = -0,3180$, $\mu_2 = 0,3047$ y $\mu_3 = -0,0129$.

Las matrices A y C se eligen en forma canónica de observador. Por su parte, los valores característicos de A se eligen de modo tal que $\lambda\{A\} = \{-0,1, -0,2, -0,3\}$, y el vector K se elige mediante un el algoritmo LQR, con las matrices $Q_{LQR} = I_{3 \times 3}$ y $R = 1$. El algoritmo de entrenamiento se simula para 7 épocas, con $\Gamma = 1_{3r \times 3r}$. En la Fig. 5.3 se muestra la evolución de los parámetros durante el entrenamiento, y en la Fig. 5.4 se muestra la salida de ambos sistemas corriendo en paralelo ante una misma entrada.

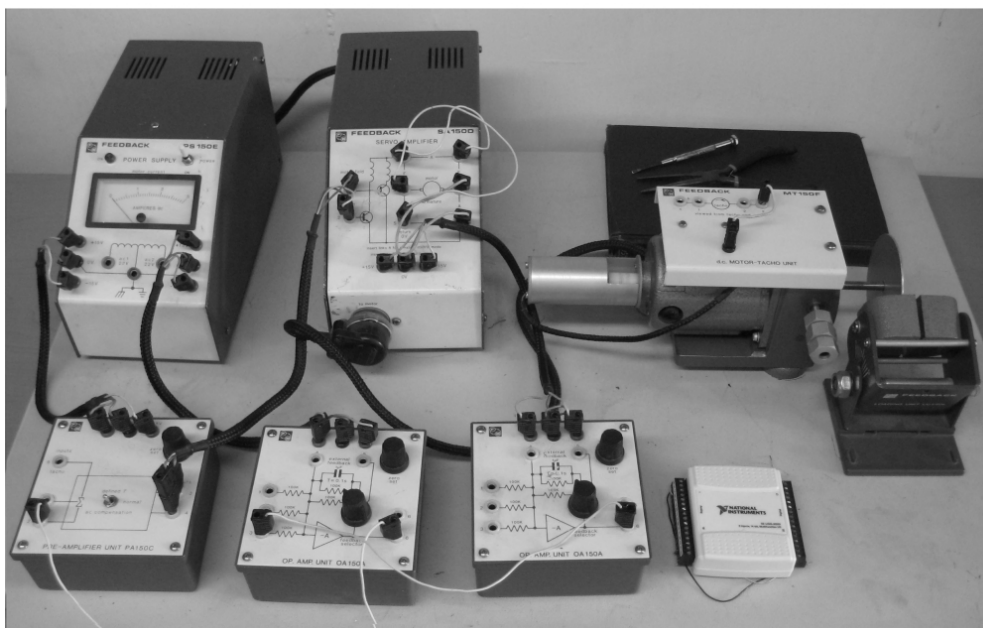


Figura 5.1: Motor CD: Sistema físico

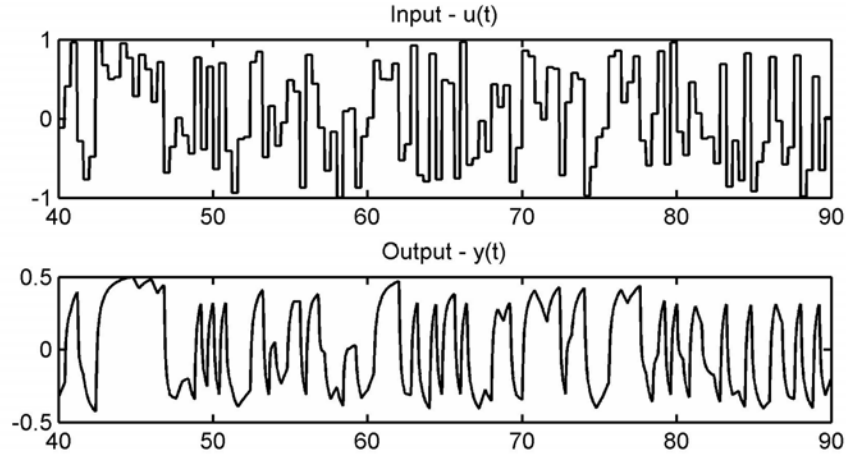


Figura 5.2: Motor CD: Señales de excitación

5.2. Robot flexible de un eslabón

En este ejemplo se lleva a cabo la identificación de un robot de un único eslabón como el mostrado en la Fig. 5.5, el cual presenta una unión flexible lineal. Este sistema es empleado por Isidori (1995) y es analizado para las propiedades de observabilidad por Marino & Tomei (1993). Este sistema puede ser modelado mediante las ecuaciones

$$\begin{aligned} J_1 \ddot{q}_1 + F_1 \dot{q}_1 + K(q_1 - q_2) + Mgl \sin(q_1) &= 0, \\ J_m \ddot{q}_2 + F_m \dot{q}_2 - K(q_1 - q_2) &= u, \end{aligned} \quad (5.1)$$

donde J_1 es la inercia del eslabón, J_m la inercia del motor, K la constante elástica del robot, M la masa del eslabón, g la aceleración asociada a la gravedad, l la distancia al centro de masa, F_1 y F_m los coeficientes de fricción viscosa. Este sistema cumple con las condiciones necesarias para su identificación, como se demuestra en Marino & Tomei (1993), por lo que puede ser identificado por la red propuesta 4.8. Para fines de simulación, se considera que todos los parámetros son unitarios (a excepción de la constante de gravedad). Para excitar al sistema se consideró una señal u aleatoria de media cero y varianza unitaria, que cambia cada 5 segundos. En este caso se emplea el regresor (4.53), y para iniciar los conjuntos de membresía se hace uso del algoritmo

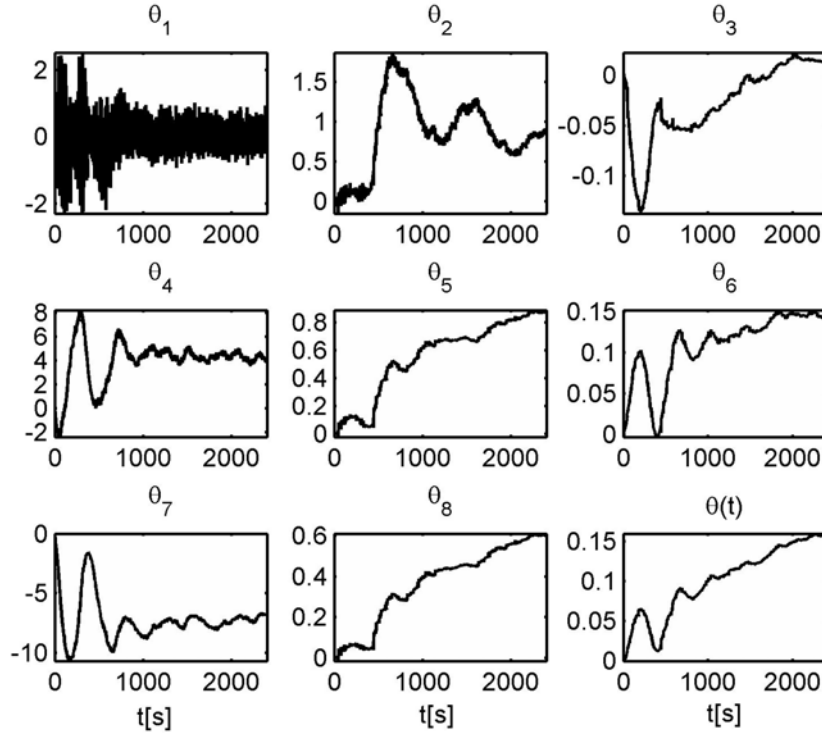


Figura 5.3: Motor CD: Evolución de los parámetros

Fuzzy C-Means con 4 reglas, obteniéndose los siguientes parámetros

$$\mu = \begin{bmatrix} \mu_{y,1} & \mu_{u,1} \\ \mu_{y,2} & \mu_{u,2} \\ \mu_{y,3} & \mu_{u,3} \\ \mu_{y,4} & \mu_{u,4} \end{bmatrix} = \begin{bmatrix} -0,308 & -0,964 \\ 1,85 & 0,762 \\ 4,28 & -0,118 \\ 0,459 & 0,0670 \end{bmatrix},$$

$${}_y\sigma_i = {}_u\sigma_i = 0,1, \quad i = 1, \dots, 4. \quad (5.2)$$

Las matrices A y C se eligieron en forma canónica de observador, con K tal que $(A - KC)$ cumpla con $\lambda(A - KC) = \{-12, -12 \pm 3i, -18\}$ y $\Gamma = 10^7 I_{n \times n}$.

La evolución de estos parámetros durante el entrenamiento (una única época) se muestra en la Fig. 5.6. Tras el entrenamiento, ambos sistemas se simulan en paralelo (el robot y la planta), donde ahora la entrada es una señal aleatoria con media cero y varianza de 5 unidades que cambia cada 20 segundos. La respuesta de la red y del

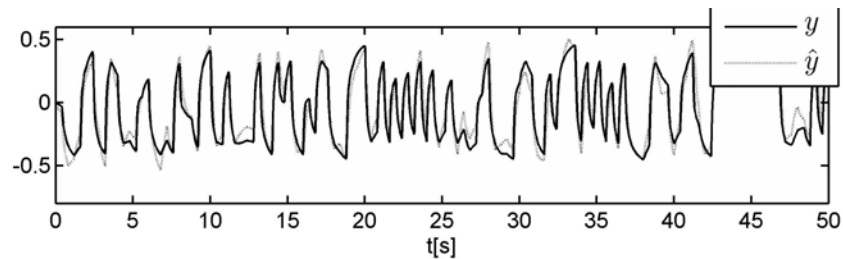


Figura 5.4: Motor CD: Desempeño de la red en paralelo con el sistema

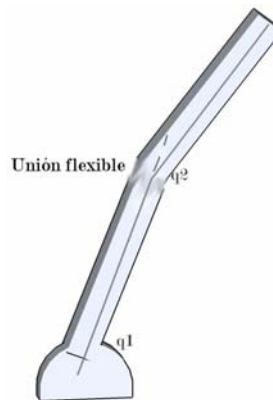


Figura 5.5: Robot Flexible: Esquema

sistema se muestra en la Fig. 5.7. Se puede ver que la red logra identificar al sistema no lineal a partir de datos entrada/salida.

5.3. Celda de tráfico

5.3.1. Descripción del problema

El problema de modelar el comportamiento dinámico de una autopista, o cualquier estructura vial, es el primer paso hacia el diseño de esquemas que permitan emplearlos óptimamente: reducir el tiempo de viaje, evitar condiciones de congestión, etc. En general, la respuesta de este tipo de sistemas es no lineal, dado que a bajas concentraciones de automóviles el flujo de los mismos se comporta en forma lineal (flujo libre); mientras que a altas densidades de vehículos se tiende a caer en condiciones

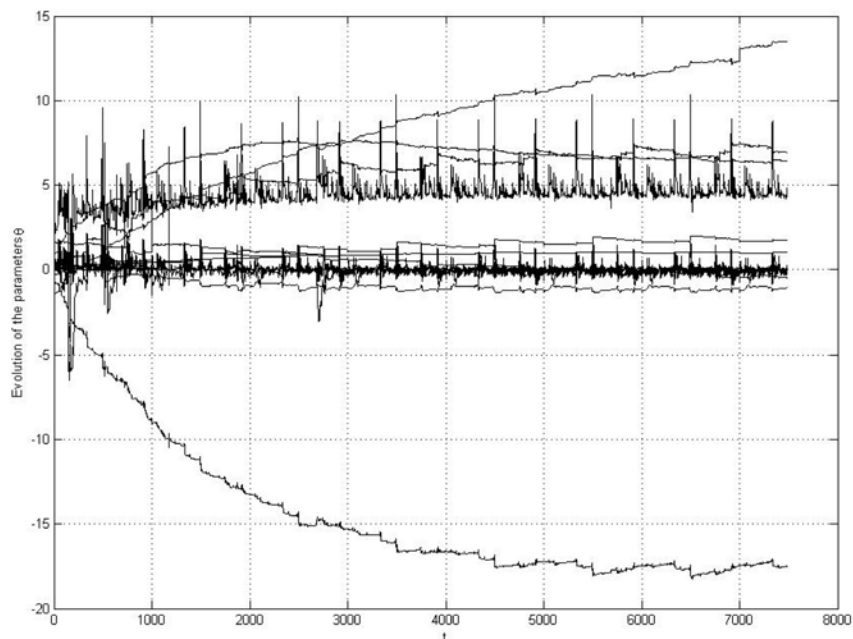


Figura 5.6: Robot flexible: Evolución de los parámetros

de congestión, donde el flujo decae a medida que aumenta la densidad (flujo congestionado). Cada sección muestra la respuesta típica flujo-densidad que se aprecia en la Fig. 5.8.

Estos modelos son en ocasiones difíciles de caracterizar, ya sea por la presencia de incertidumbre paramétrica y dinámicas no modeladas, entre otros.

En este sentido, varias estructuras han sido propuestas en la literatura. En primer lugar, los *modelos macroscópicos* (Nagel, 1996) no toman en cuenta las variables relacionadas con cada vehículo en el camino, sino que toman en cuenta variables como la velocidad y flujo promedio de vehículos en una sección de una longitud determinada (generalmente algunos cientos de metros). Algunos ejemplos de estos modelos son dados por Helbing (1996); Payne (1971). Generalmente es posible expresar estos modelos por conjuntos finitos de ecuaciones diferenciales. Uno de los más empleados es *modelo de transmisión por celdas*, propuesto por Daganzo (1994), el cual es empleado en esta aplicación.

Por su parte, los *Modelos microscópicos* consideran las relaciones y comportamiento

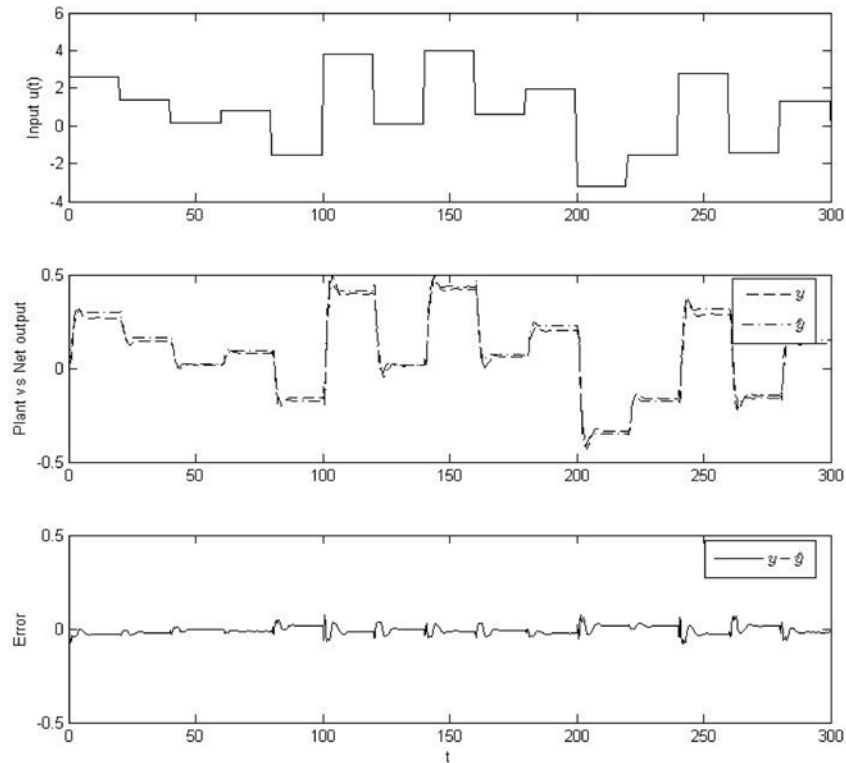


Figura 5.7: Robot flexible: Señales de salida

individual de cada vehículo en particular. Algunos ejemplos pueden encontrarse en el trabajo de Arem & Hogema (1995); Minderhoud (1999); Pipes (1953). Finalmente, los *Modelos mesoscópicos* modelan el comportamiento del grupos de vehículos a partir de ecuaciones inspiradas en la física de gases, empleando ecuaciones probabilísticas.

Como se mencionó, el *modelo de transmisión por celdas*¹ propuesto por Daganzo (1994), propone dividir a la autopista en segmentos o *celdas*, como se muestra en la Fig. 5.9, tras lo cual analiza el comportamiento de la densidad de vehículos en la misma (por unidad de longitud) con base en los flujos vehiculares de entrada y salida, adyacentes a ésta. Este modelo es sencillo en su estructura y cuenta con interpretación y sentido físicos, pero falla en describir cierta dinámica que aparece en el sistema real, como ciertos efectos de sincronización entre grupos de vehículos. En este mismo sentido, algunos modelos más complejos incluyen la dinámica de la velocidad media de los

¹Mejor conocido por su nombre en inglés : *Cell transmission model*

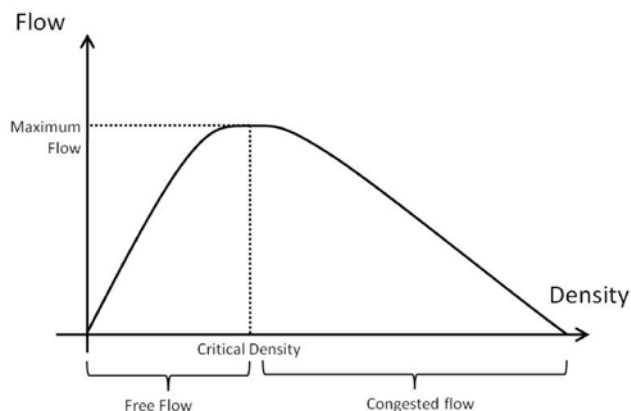


Figura 5.8: Celda de Tráfico: Relación densidad-flujo vehicular en una autopista

vehículos dentro de la celda (Papageorgiou, 1990). Sin embargo, la complejidad del modelo es proporcional a la dificultad de análisis.

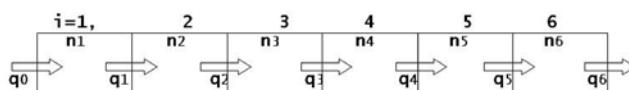


Figura 5.9: Celda de Tráfico: División de una autopista por celdas. q_i Representa el flujo que abandona la celda i en [vehículos por hora] y n_i la densidad de vehículos por milla.

Las redes neuronales y los sistemas difusos han sido empleados para atacar el problema de modelado de este tipo de sistemas, ya sea como predictores a corto plazo (Ho & Ioannou, 1996; Yasdi, 1999; Yu & Chen, 1993), o para conmutar dos modelos lineales en paralelo (Rosas-Jaimes & Alvarez-Icaza, 2007). En esta aplicación se mostrará el desempeño de la red recurrente en tiempo continuo propuesta para atacar el problema de modelado e identificación de una celda de tráfico.

5.3.2. Desarrollo

La dinámica de una celda de tráfico puede ser modelada mediante ecuaciones diferenciales donde la señal de entrada es la densidad en la celda anterior n_{i-1} , con la señal de salida el flujo q_i . Para evaluar el desempeño de la red para modelar e identificar la dinámica densidad-flujo en una celda de tráfico, se emplean datos reales obtenidos de



Gráfica tomada de Wikipedia Commons, en http://commons.wikimedia.org/wiki/File:California_Interstate_and_State_Route_210.svg

Figura 5.10: Celda de Tráfico: Localización de la autopista i210-West.

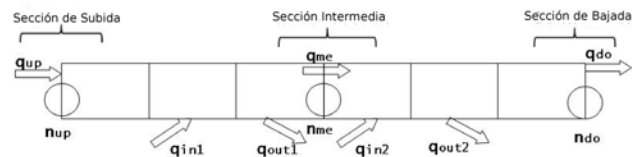


Figura 5.11: Celda de Tráfico: Sección de la autopista i210-West

la autopista i210-West en California el 25 de abril de 2001, con un tiempo de muestreo de 5 segundos. La sección de autopista se muestra en la Fig. 5.10, de la cual solo se considera una sección, cuyo esquema se muestra en la Fig. 5.11. El segmento cuenta con varios puntos donde los vehículos pueden entrar y salir de la misma. Para este ejemplo, el modelado e identificación se lleva a cabo en tres celdas diferentes: en la sección de entrada (*UP*), intermedia (*ME*) y de salida (*DO*). En estos tres casos, la señal de entrada se considera como la densidad de vehículos n en la sección (n_{up} , n_{me} and n_{do} respectivamente), y la señal de salida $y(t)$ es el flujo q (q_{up} , q_{me} and q_{do} respectivamente).

Para este ejemplo, se considera el regresor (4.38) con funciones de membresía gaussianas y desdifusión con promedio ponderado. Para generar la parte antecedente de la red, los centros de las funciones de membresía son obtenidas a partir del algoritmo *Fuzzy C-Means*. Para este fin, los datos fueron normalizados con respecto a los valores máximos de densidad (densidad de congestionamiento) y flujo (máximo flujo posible

en la autopista). Dichos valores se encuentran reportados en la Tabla 5.2, obtenidos a partir del trabajo de Rosas-Jaimes & Alvarez-Icaza (2007). Las funciones de membresía obtenidas se muestran en la Tabla 5.3.

Con el fin de cubrir las diferentes secciones de la curva densidad-flujo (5.8), se consideran tres reglas en cada caso, con dos estados por cada modelo.

Las funciones de membresía finales se muestran en la Tabla 5.3, y la función sigmoide $\rho(\cdot)$ es definida como $\rho(y) = \tanh(y)$. Se puede observar que todas las combinaciones densidad-flujo son cubiertas por las tres secciones. Se define la dinámica lineal como se muestran en la Tabla 5.1.

Sect.	Matrices	θ_u	θ_y
UP	$A_{UP} = \begin{bmatrix} -3 & 1 \\ -2 & 0 \end{bmatrix},$ $\lambda\{A\} = \{-2, -1\}$ $C_{UP} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ $\lambda\{A_{UP} - K_{UP}C_{UP}\} = \{-3, -2\}$ $\Gamma_{UP} = 1I_{n_T \times n_T}.$	$\begin{bmatrix} -0,783 \\ -0,0275 \\ 0,155 \\ 0,653 \\ 0,0420 \\ 0,504 \end{bmatrix}$	$\begin{bmatrix} 0,443 \\ 2,24 \\ 0,355 \\ 2,27 \\ 0,0467 \\ 2,06 \end{bmatrix}$
ME	$A_{ME} = \begin{bmatrix} -1,5 & 1 \\ -0,5 & 0 \end{bmatrix},$ $\lambda\{A\} = \{-1, -0,5\}$ $C_{ME} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ $\lambda\{A_{ME} - K_{ME}C_{ME}\} = \{-3, -2\}$ $\Gamma_{ME} = 10I_{n_T \times n_T}.$	$\begin{bmatrix} -1,22 \\ 0,00608 \\ 0,412 \\ -0,118 \\ 0,156 \\ 0,709 \end{bmatrix}$	$\begin{bmatrix} 1,85 \\ 2,23 \\ 0,854 \\ 2,50 \\ 0,542 \\ 2,05 \end{bmatrix}$
DO	$A_{DO} = \begin{bmatrix} -3 & 1 \\ -2 & 0 \end{bmatrix},$ $\lambda\{A\} = \{-2, -1\}$ $C_{DO} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ $\lambda\{A_{DO} - K_{DO}C_{DO}\} = \{-3, -2\}$ $\Gamma_{ME} = 1I_{n_T \times n_T}.$	$\begin{bmatrix} -1,28 \\ -0,0874 \\ 0,0614 \\ 0,222 \\ 0,101 \\ -0,132 \end{bmatrix}$	$\begin{bmatrix} 1,25 \\ 2,29 \\ 0,0594 \\ 2,10 \\ 0,687 \\ 2,46 \end{bmatrix}$

Cuadro 5.1: Celda de Tráfico: Matrices diseñadas para cada sección y parámetros consecuentes

Sect.	máx n	máx q	máx n	máx q
	Data		Theoretical	
UP	241	5441	335	5441
ME	307	4530	374	4530
DO	314	5486	374	5486

Cuadro 5.2: Celda de Tráfico: Valores máximos de los datos

Los resultados de simulación se muestran en la Figura 5.12 para las secciones de subida, intermedia y bajada, así como la evolución de los parámetros. Se puede observar que ante la presencia del error de aproximación de la red (y dinámicas no capturadas por la red), existen aún pequeños errores de modelado, pero un general la dinámica es capturada por la red. Los parámetros consecuentes obtenidos (para los valores normalizados) se muestran en la Tabla 5.1.

La principal diferencia mostrada con otro tipo de estructuras es que en este caso una sola red modela la dinámica de la red. En otras aproximaciones, como en Rosas-Jaimes & Alvarez-Icaza (2007), la dinámica se maneja mediante un par de modelos lineales que corren en paralelo y un sistema difuso combina ambos desempeños. En la aproximación mostrada, únicamente mediante la densidad como señal de entrada es posible reproducir al flujo en una cierta celda, con un conocimiento mínimo de la dinámica real del sistema, así como de la información paramétrica del sistema. Comparativamente, la red mostrada reproduce el diagrama fundamental (Fig. 5.8) únicamente a partir de datos.

Sect.	μ_u	μ_y	σ_u	σ_y	Membresía
UP	$\begin{bmatrix} 0,6798 \\ 0,259 \\ 0,12 \end{bmatrix}$	$\begin{bmatrix} 0,53 \\ 0,783 \\ 0,488 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 10 \\ 6 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 10 \\ 3 \end{bmatrix}$	
ME	$\begin{bmatrix} 0,678 \\ 0,419 \\ 0,115 \end{bmatrix}$	$\begin{bmatrix} 0,597 \\ 0,791 \\ 0,510 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 4 \\ 4 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 4 \\ 4 \end{bmatrix}$	
DO	$\begin{bmatrix} 0,594 \\ 0,1 \\ 0,255 \end{bmatrix}$	$\begin{bmatrix} 0,559 \\ 0,45 \\ 0,729 \end{bmatrix}$	$\begin{bmatrix} 6 \\ 5 \\ 10 \end{bmatrix}$	$\begin{bmatrix} 6 \\ 5 \\ 10 \end{bmatrix}$	

Cuadro 5.3: Celda de tráfico: Funciones de membresía por sección

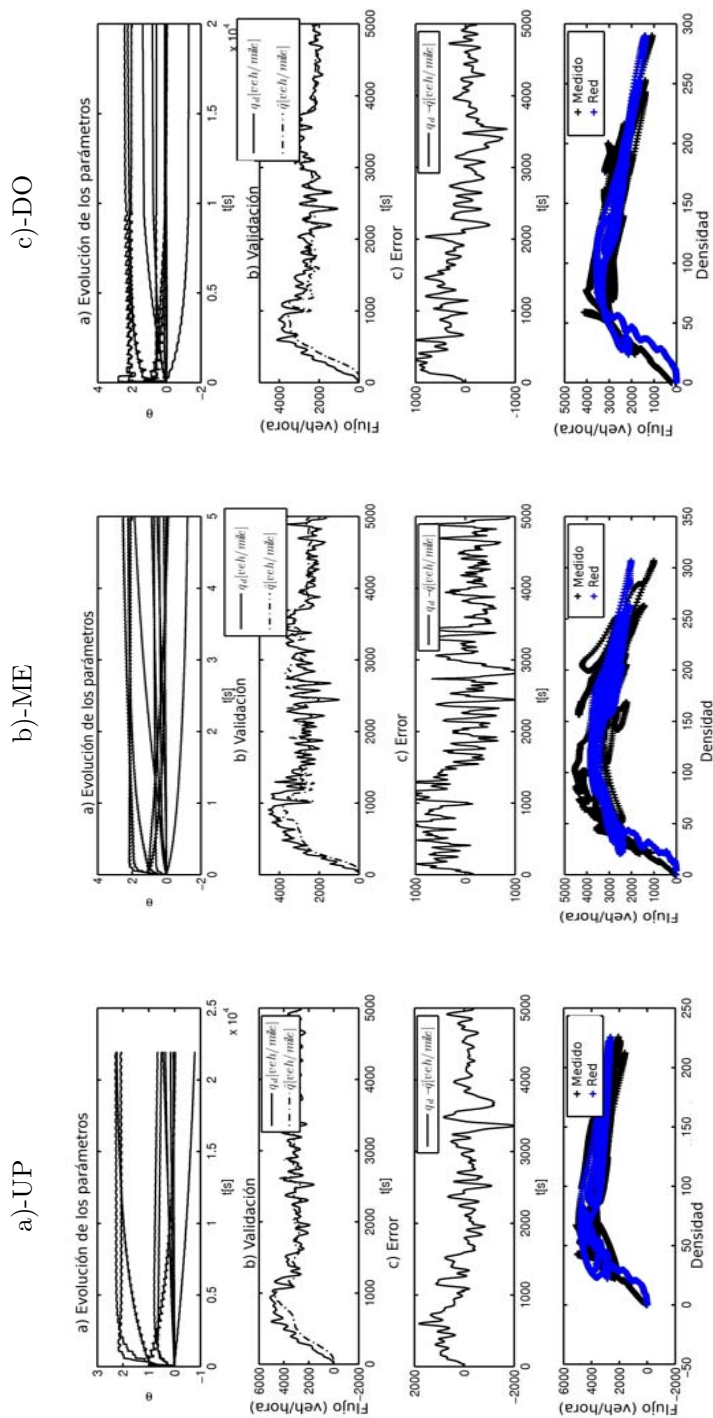


Figura 5.12: Celda de Tráfico: Evolución de los parámetros durante el entrenamiento

5.4. Robot planar de 2GDL para servo-visual

5.4.1. Descripción del problema

En esta aplicación se muestra se emplea a la red para identificar el desempeño dinámico de un robot de dos grados de libertad montado en un sistema de visión.

Como es indicado por Chaumette & Hunthinson (2006), un problema *servo-visual* se refiere al problema de controlar el movimiento de un robot empleando un sistema de visión por computadora, donde la cámara empleada puede estar tanto en movimiento como estática con respecto a los ejes de referencia.

En aplicaciones clásicas (Fu *et al.*, 1987), el control del robot se lleva a cabo en un ambiente estructurado mediante la medición directa de las posiciones y velocidades de cada eslabón mediante codificadores digitales. En este tipo de casos, un sistema de visión puede ser empleado como una forma de medición redundante (Kelly, 1996), o como complemento para visualizar la trayectoria a seguir.

En años recientes, el problema de servo-visión ha sido estudiado como una estrategia para evitar la medición directa de dichas variables, donde la única información sobre las mismas sea obtenida a través de una cámara. Esta aproximación ha sido empleada en mecanismos microrrobóticos o en esquemas donde es deseable mantener mediciones de respaldo (Wilson *et al.*, 1996), como ambientes no estructurados donde se desea mantener un correcto diseño de la trayectoria a seguir (Chen & Kak, 1989).

Considérese ahora el problema de un manipulador rotacional de dos grados de libertad con un sistema de visión como el mostrado en la Fig. 5.13, donde no existen más mediciones de posición ni velocidad más que aquellas obtenidas en el plano de imagen de la cámara. El robot es excitado por dos pares de fuerza, τ_1 y τ_2 , logrados a partir de excitar a los actuadores (motores de corriente directa) con señales de voltaje u_1 y u_2 respectivamente, lo que provoca que las posiciones angulares *reales* q_1 y q_2 de cada eslabón se modifiquen. En el plano de imagen, como se muestra en la Fig. 5.13, es posible medir estos ángulos como ϕ_1 and ϕ_2 .

Sin embargo, la cámara actúa como una transformación y altera la imagen recibida, cuando se compara con la posición de las articulaciones del robot, ya sea por su posición como por la propia óptica de la cámara. De este modo, muchos esquemas de control requieren que la cámara sea previamente calibrada para reducir este efecto. Mostraremos en este caso cómo la red recurrente propuesta es capaz de identificar la

dinámica voltaje-ángulo (en el plano de la cámara), como un sistema dos-salidas-una-entrada.

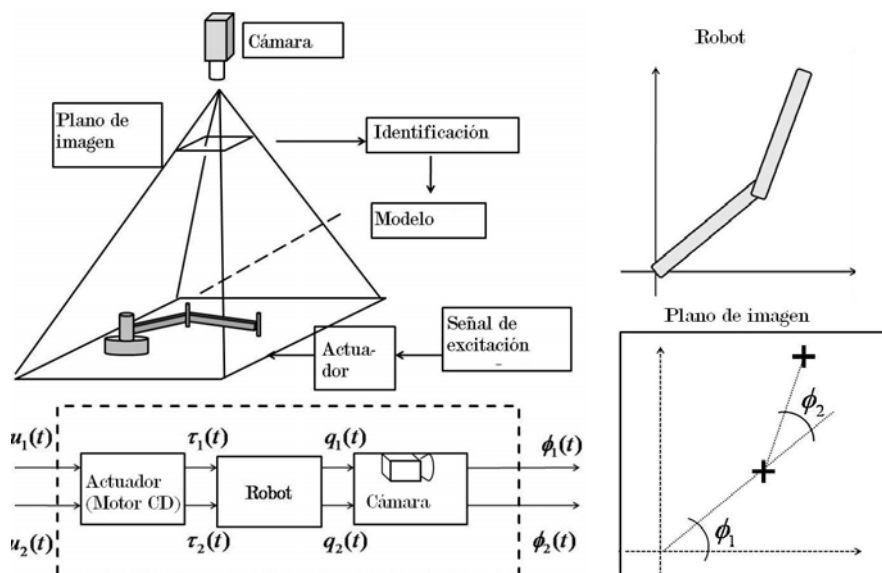


Figura 5.13: Sistema de servo-visual: Sistema de servo-visual y robot en el plano de imagen

5.4.2. Desarrollo

Dado el sistema de la Fig. 5.13, el problema se puede considerar como el identificar y modelar un sistema de múltiples entradas y una salida¹. En este caso, consideraremos en forma independiente a las salidas de ángulo en el plano de imagen (esto es, $y(t) = \phi_1(t)$ o $y(t) = \phi_2(t)$). Sabemos que este sistema puede ser expresado mediante el sistema

$$\begin{aligned}\dot{\mathbf{x}}(t) &= f(\mathbf{x}, \mathbf{u}), \\ y(t) &= h(\mathbf{x}),\end{aligned}\tag{5.3}$$

donde $\mathbf{u}(t) = [u_1(t)u_2(t)]^T \in \mathfrak{R}^2$ y $y(t) \in \mathfrak{R}$, y $\mathbf{x} \in \mathfrak{R}^4$ es el estado del sistema, $f : \mathfrak{R}^4 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^4$, $h : \mathfrak{R}^4 \rightarrow \mathfrak{R}$, son funciones desconocidas (función tanto de la

¹MISO por sus siglas en inglés: *Multiple-Input-Single-Output*.

dinámica del sistema como de la transformación asociada a la cámara). Se sabe además que (5.3) es observable en la región de interés (Spong, 1989) y que cumple con las condiciones necesarias para ser transformado en (4.2).

El proceso de modelado e identificación con base en la red neurodifusa recurrente (4.8) se lleva a cabo tomando al par de voltajes u_1 y u_2 como señales de entrada y a ϕ_1 o ϕ_2 como la salida del mismo. En primer lugar, se obtienen datos entrada-salida del sistema (5.3) simulado, asumiendo que solo existe fricción viscosa (a diferencia del modelo real, donde existe además fricción seca y viscosa no uniforme), con el fin de probar si el modelo teórico inicial puede ser correctamente identificado por la red. Una vez que se evalúe su desempeño ante condiciones *ideales*, se probará con datos experimentales, empleando una cámara de 16 cuadros por segundo. En ambos casos se consideró en regresor dado por (4.38).

5.4.2.1. Resultados de simulación

Para este caso, los datos se obtienen mediante la simulación del sistema dado por Spong (1989), y el modelo de cámara reportado por Kelly (1996), con los parámetros obtenidos directamente del sistema disponible en el laboratorio (Rodriguez & Tang, 2009). Para tal efecto, el robot es excitado por las señales de voltaje indicadas en la Fig. 5.14, y las trayectorias obtenidas se muestran en la Fig. 5.15. Para esta red, se consideran dos reglas ($r = 2$); y en virtud de que el sistema cuenta con dos grados de libertad, se considera $n = 4$. Las señales de entrada y salida son normalizadas con respecto a los valores máximos obtenidos a partir de los datos, y se entrena una red diferente para cada salida considerada (ϕ_1 o ϕ_2).

Por otra parte, las funciones de membresía son obtenidas a partir del algoritmo Fuzzy C-Means, mostradas en la Fig. 5.16 en cada caso. Nótese que las funciones de membresía cubren a todo el espacio de trabajo y por lo tanto no se dejan zonas sin cubrir. En ambos las matrices A y C se diseñan en forma canónica de observador y $\lambda(A) = \{0, -2, -3, -4\}$. Por su parte, para el algoritmo de entrenamiento se toma $\lambda(A - KC) = \{-2, -3, -4, -6\}$, y $\Gamma = I_{n_T \times n_T} \cdot \alpha$, donde α comienza en 10^4 , y decrece en forma lineal durante el entrenamiento hasta 100.

El entrenamiento se aplica por varias épocas, hasta lograr un error mínimo RMS o alcanzar un número máximo de las mismas. Estos resultados se muestran en la Figuras

5.17. Se puede notar que los parámetros son convergentes y que el error RMS decrece con cada época.

Tras el entrenamiento, la red es probada en paralelo con el modelo *real* de los ángulos obtenidos desde la cámara. Estos resultados se muestran en la Figura 5.18. Nótese que el error de modelado es acotado en ambos casos, a pesar de que en cada caso la señal de salida de un sistema no es considerada en la del otro (esto es, cuando $y = \phi_1$, ϕ_2 no es realimentado a la red, y viceversa cuando $y = \phi_1$).

5.4.2.2. Resultados experimentales

Tras probar el modelo en condiciones *ideales*, la red es ahora entrenada con datos obtenidos directamente de un sistema físico de servo-visual (Rodríguez & Tang, 2009). La red es nuevamente entrenada con el mismo algoritmo con 4 reglas, en vista de que es necesario aumentar la capacidad de aproximación de la red ante la presencia de una fricción viscosa no uniforme y fricción seca. Se emplean las mismas matrices A , C , y K consideradas en el paso anterior. Los resultados se comparan con un sistema lineal de 4o orden, con función de transferencia $H(s) = \frac{\beta_1 s^4 + \beta_2 s^3 + \beta_3 s^2 + \beta_4 s + \beta_5}{s^4 + \alpha_1 s^3 + \alpha_2 s^2 + \alpha_3 s + \alpha_4}$. Este último sistema es identificado mediante la herramienta *Ident* de *Matlab*.

La comparación del desempeño de ambos modelos identificados se muestra en la Fig. 5.19. Se puede ver que la red recurrente tiene un desempeño mejor que el sistema lineal (como se puede esperar, ya que la segunda es un sistema no lineal), pero que existen zonas donde el modelo identificado no logra seguir correctamente al sistema real, en gran medida debido a la irregularidad de la fricción, tanto seca como viscosa, del esquema experimental empleado.

5.5. Sistema glucosa-insulina en pacientes diabéticos

La glucosa es el principal substrato energético en el cuerpo humano, al ser descompuesta la molécula y liberarse la energía almacenada en la misma. Sin embargo, solo el 20% de las células pueden absorberla directamente, mientras que el restante requiere que la hormona insulina para realizar el anabolismo de los hidratos de carbono. Esta hormona, producida por las células beta (β) en los islotes de Langerhans del páncreas, interviene también en el almacenamiento de glucosa en el hígado, músculos y tejido

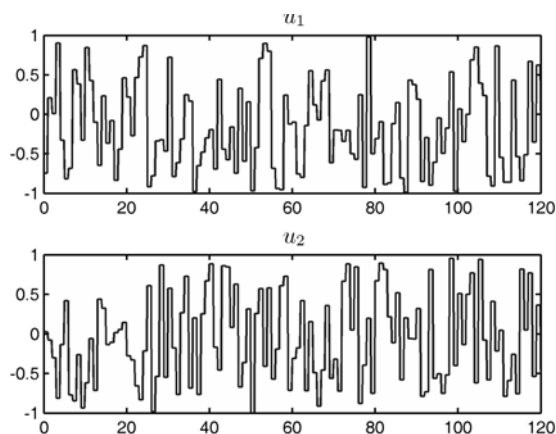


Figura 5.14: Sistema de servo visión: Señales de entrada

adiposo, para ser empleada durante periodos en los que no se disponga de la cantidad suficiente de glucosa en el torrente sanguíneo Guyton & Hall (2000).

La diabetes tipo 1 es una enfermedad autoinmune en la cual las células β son atacadas por el propio organismo, lo que provoca que la producción de insulina decaiga drásticamente, provocando que los niveles de glucosa se incrementen por encima de su nivel basal, lo que conlleva riesgos para el funcionamiento del organismo. La medicina actual recomienda administrar regularmente inyecciones subcutáneas con concentraciones específicas de insulina.

Un buen modelo de la dinámica de la insulina y la glucosa en un paciente con diabetes tipo 1 permite analizar el estado de salud del mismo y diseñar para dosificar dicha hormona. En algunos casos, estas dosis son suministradas mediante las mencionadas inyecciones (Asociation, 2009) o, con técnicas de control automático, mediante bombas de insulina empleando sensores de glucosa implantables. Con este último esquema, existen varios trabajos de investigación (Gallardo-Hernandez *et al.*, 2008; Hovorka *et al.*, 2004; Kaveh & Shtessel, 2007; Parker *et al.*, 2004).

En la literatura existen varias técnicas para modelar a este sistema (Bergman *et al.*, 1979; Bolie, 1961; Man *et al.*, 2007; Sorensen, 1985; Wilinska *et al.*, 2005), siendo la entrada al mismo la concentración de insulina provista mediante inyección y las variables de salida la concentración de insulina en sangre, y/o la glucosa (las cuales son también variables de estado del sistema, entre otras). La complejidad del mismo varía

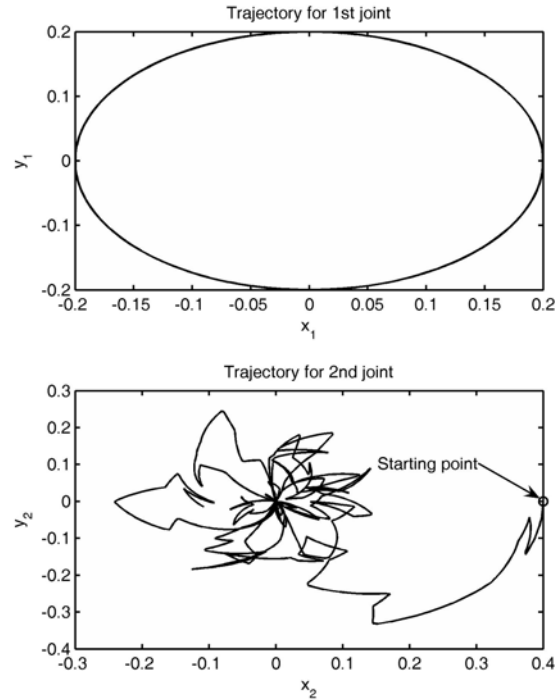


Figura 5.15: Sistema de servo visión: Trayectorias del robot

desde modelos con 3 estados (Bergman *et al.*, 1979) hasta muy detallados con 21 estados (Sorensen, 1985). Sin embargo, existen varios problemas al obtener este modelo, entre los cuales se encuentra la incertidumbre paramétrica, dinámica no modelada y variables de difícil medición (ya sea por costo o por cuestiones técnicas). En particular, el modelo metabólico propuesto por Sorensen (1985) involucra un gran número de parámetros, lo que no lo hace apto para su obtención para cada paciente, dada la complejidad de mediciones requeridas y gastos asociados.

La existencia de un modelo que permita describir en forma sencilla a cada paciente permite probar una estrategia de control para evitar las complicaciones típicas de la terapia con insulina, entre las que se encuentran la hiperinsulinemia (concentración excesiva de insulina en sangre, que a su vez provoca hipoglucemia y a largo plazo resistencia a la insulina) y la hipoinsulinemia (baja cantidad de insulina en sangre, que provoca una lipólisis excesiva, oxidación de tejidos grasos, acidosis metabólica y deshidratación (Larsen *et al.*, 2003)).

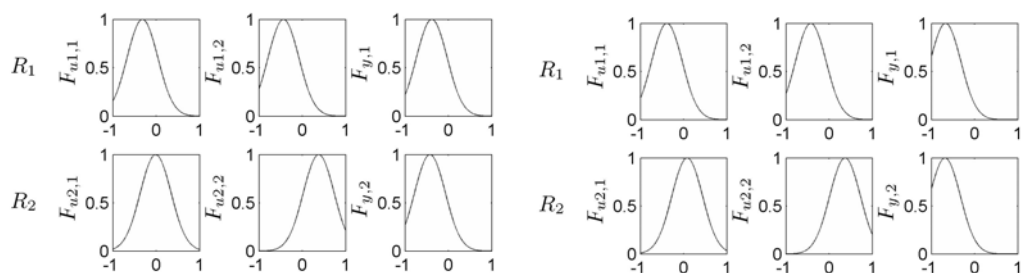


Figura 5.16: Sistema de servo visión: Funciones de membresía para $y(t) = \phi_1$ y $y(t) = \phi_2$

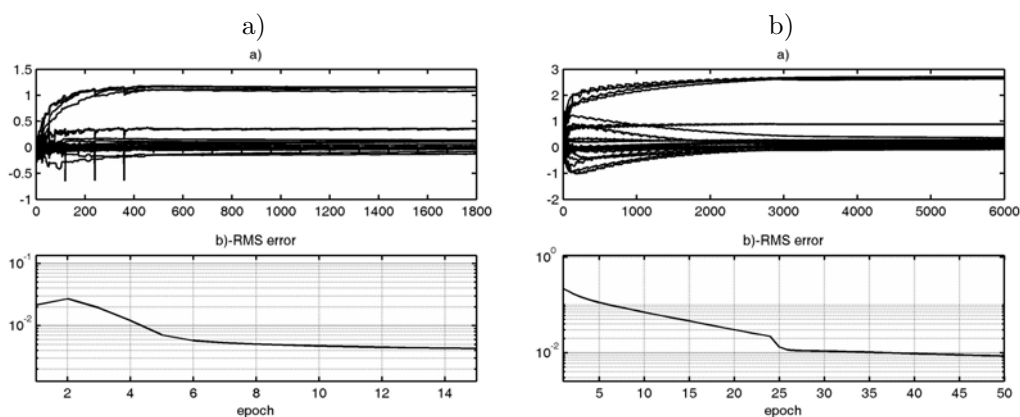


Figura 5.17: Sistema de servo visión: a) Evolución de parámetros y b) Error RMS error para a) $y(t) = \phi_1$, b) $y(t) = \phi_2$.

En esta aplicación se considera la obtención de un modelo basado en la red neuro-difusa recurrente en tiempo continuo que permita analizar la dinámica de la insulina en diversos pacientes bajo dos escenarios diferentes. En el primero de ellos el modelo toma la concentración de insulina inyectada al paciente como variable de entrada y la concentración de insulina en sangre como salida, ignorando la concentración de glucosa presente en sangre. En el segundo escenario, se supone la existencia de un medidor de glucosa, y se considera a esta variable como una entrada extra al sistema. En ambos casos, las señales de entrenamiento y validación son tomadas del trabajo de Gallardo-Hernandez *et al.* (2008) con base en el modelo mínimo con parámetros de Bergman publicado por Kaveh & Shtessel (2007).

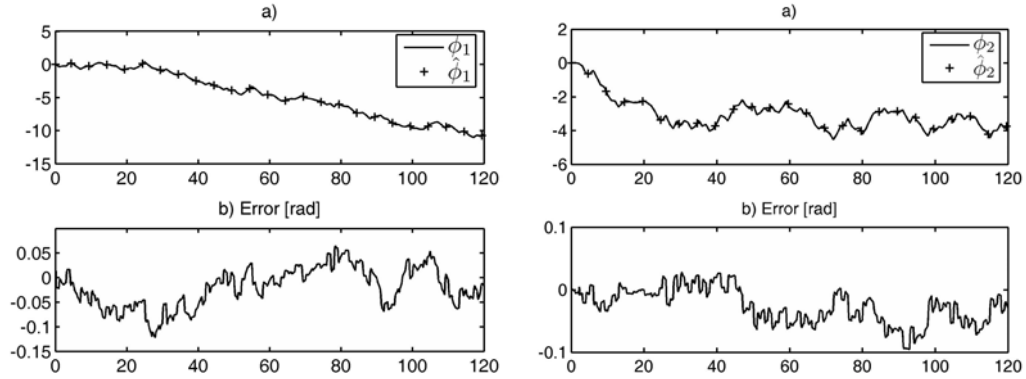


Figura 5.18: Sistema de servo visión: Resultados de modelado para $y(t) = \phi_1$. a) ϕ_1 vs. $\hat{\phi}_1$, b) Error $\phi_1 - \hat{\phi}_1$, c) ϕ_2 vs. $\hat{\phi}_2$, d) Error $\phi_2 - \hat{\phi}_2$.

5.5.1. Desarrollo

Para llevar a cabo la identificación de la dinámica de la insulina presente en sangre ante una inyección externa de esta hormona, se tomaron diversos casos de pacientes (a partir de un modelo de simulación), donde la entrada $u(t)$ es la tasa de insulina provista en $\frac{\mu U}{ml \cdot min}$ inyectada al paciente (calculada con base en el algoritmo cuasi-continuo descrito por Gallardo-Hernandez *et al.* (2008)). En un primer caso, se ignora a la concentración de glucosa en sangre (G en $\frac{mg}{dl}$) y se analiza la capacidad de aproximación de la red; mientras en el segundo se toma en cuenta dicha variable como señal de entrada al modelo. En ambos la señal de salida es la concentración de insulina en sangre I en $(\frac{\mu U}{ml})$. Esto obedece a que, si bien en pacientes sanos existe una fuerte correlación entre la glucosa y la insulina presente en sangre, en pacientes diabéticos esta relación sufre de un desacoplamiento, además de existir varios sensores de glucosa rápidos y relativamente baratos que permiten medirla en forma confiable.

La identificación se lleva a cabo en dos partes. En primer lugar, para demostrar la capacidad de aproximación de la red, esta es entrenada asumiendo que la glucosa e insulina pueden ser medidas en 5 ocasiones por segundo. Sin embargo, las técnicas actuales de medición de insulina no permiten realizar esto más que cada 5 minutos con un máximo de 80 mediciones en un cierto periodo de tiempo (ya que se requiere de extraer sangre del paciente en ciertas cantidades, y esto no es posible hacerlo por largos periodos de tiempo). En ambos casos, los conjuntos de membresía son fijos y se

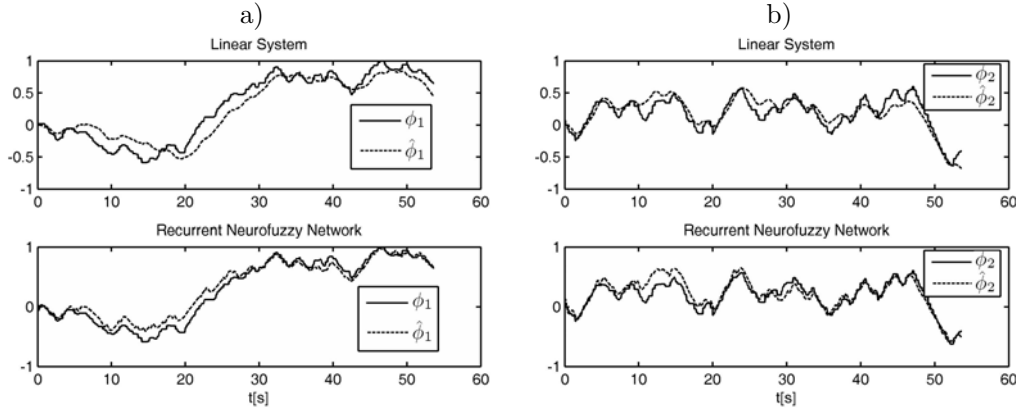


Figura 5.19: Sistema de servo visión: Resultados de modelado con datos experimentales a) ϕ_1 , b) ϕ_2 .

obtienen mediante un algoritmo c-medias difuso y se comparan en paralelo la dinámica *real* contra la provista por la red. En todas las figuras mostradas, la unidad de tiempo es el minuto, y se emplean tres reglas ($r = 3$) y tres estados ($n = 3$), con las siguientes matrices de diseño: $\lambda\{\bar{A}\} = [-0,8 \quad -0,6 \quad -0,4]$, $C = [1 \quad 0 \quad 0]$, $\lambda\{\bar{A} - KC\} = [-1,5 \quad -1 \quad -0,75]$, $Epoch = 30$, $\Gamma_{initial} = 1$, $\Gamma_{final} = 0,1$. En total, se entrenan 27 parámetros. Asimismo, en todos los casos las señales son tomadas considerando que existen 4 perturbaciones de glucosa (por ingestión de alimentos), para los cuales el control reportado por Gallardo-Hernandez *et al.* (2008) es empleado para suministrar las dosis de insulina, y los conjuntos difusos son obtenidos a partir del algoritmo Fuzzy C-means.

5.5.2. Resultados de simulación

En primer término, se muestran los resultados de identificación considerando la condición ideal donde la insulina presente en sangre puede ser medida hasta en 5 ocasiones por minuto durante un periodo de tiempo de 400 minutos. En este caso, solo se toma como señal de entrada a la insulina provista por el control (Gallardo-Hernandez *et al.*, 2008), y como salida a la concentración de insulina en sangre. En la Fig. 5.20 se muestra la evolución de los parámetros y el resultado de la identificación al operar la red en paralelo con el paciente, bajo la misma entrada de insulina. Se puede apreciar

que el entrenamiento es tal que los parámetros son convergentes, y la dinámica del sistema es capturada por la red (aún sin la medición de la glucosa).

Estos mismos experimentos fueron repetidos considerando ahora que la glucosa puede ser medida y tomada como una entrada más al sistema, con el mismo número de muestras y periodo de tiempo. En este caso, los resultados se muestran en la Fig. 5.21 y se puede apreciar cómo el error de identificación es en este caso menor en vista de la mayor información que se tiene disponible para el entrenamiento del sistema.

Finalmente, en la Fig. 5.22 se muestra el caso en el cual la cantidad de muestras y la frecuencia entre las mismas tiene el límite de 80, con un tiempo de muestreo de 5 minutos entre las mismas. Este caso es más cercano al tipo de experimentos que se pueden realizar en tiempo real dentro de un laboratorio médico. Es claro que existirá una pérdida de información con respecto a la dinámica ante esta baja frecuencia de muestreo, como se puede visualizar en la misma figura. Sin embargo, la red es aún capaz de identificar el desempeño de los tres sistemas, con una convergencia paramétrica. Es también de observar cómo el algoritmo de agrupamiento logra funciones de membresía muy similares para los tres casos identificados, las cuales se muestran en la Fig. 5.23.

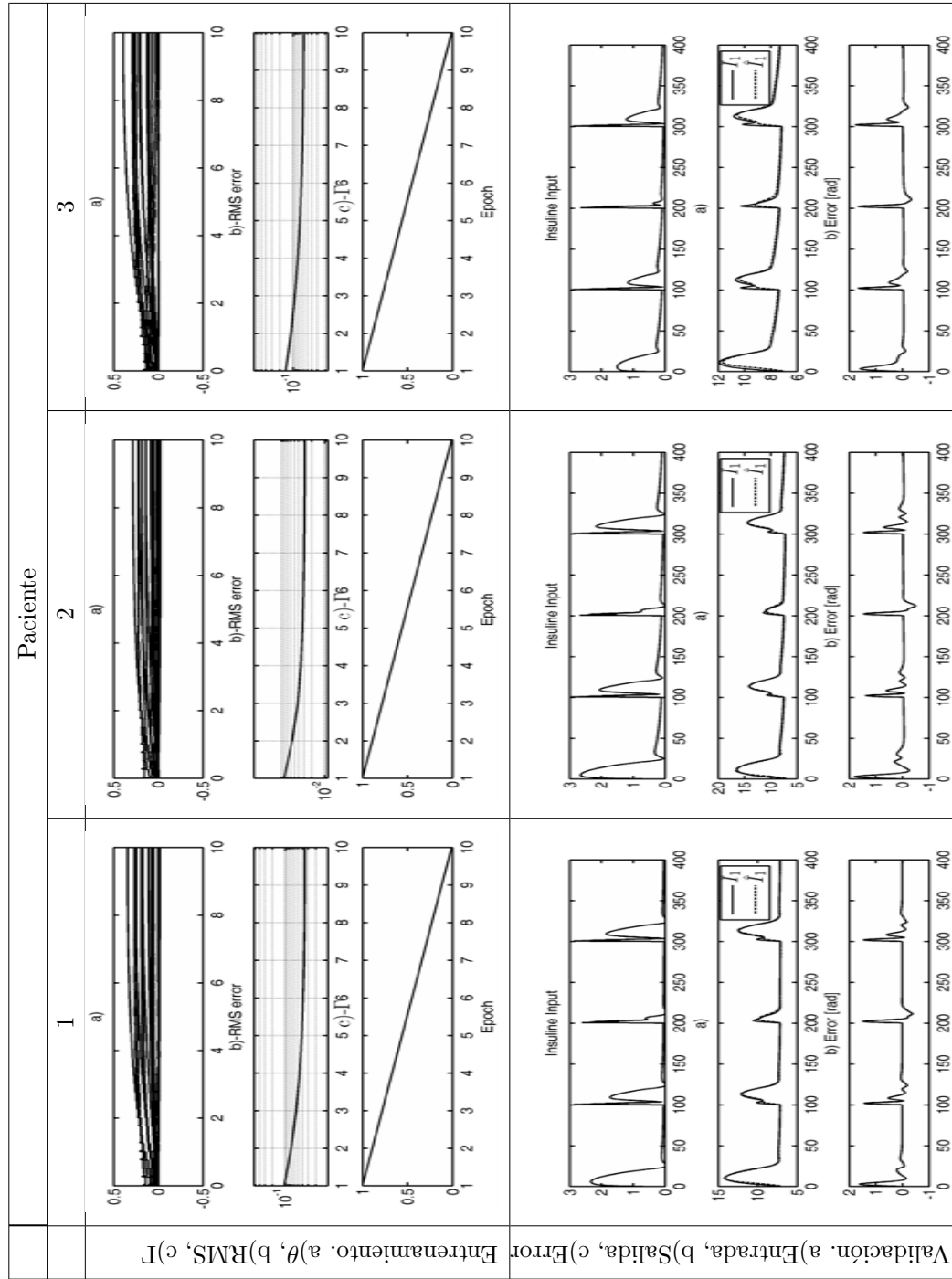


Figura 5.20: Dinámica de la insulina: Identificación con una entrada (insulina).

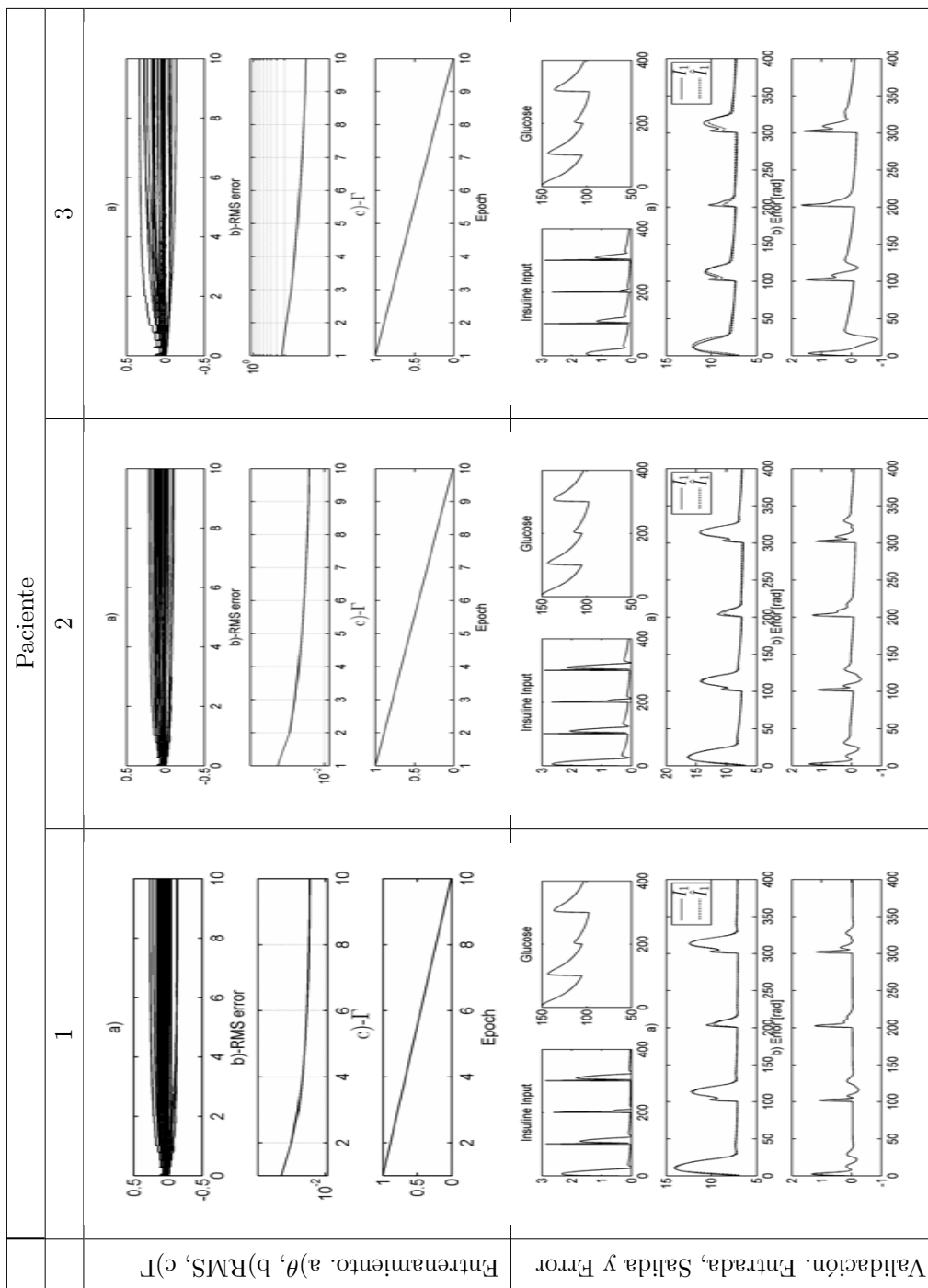


Figura 5.21: Dinámica de la insulina: Identificación con dos entradas (glucosa e insulina) y $T_s = 0,2[min]$

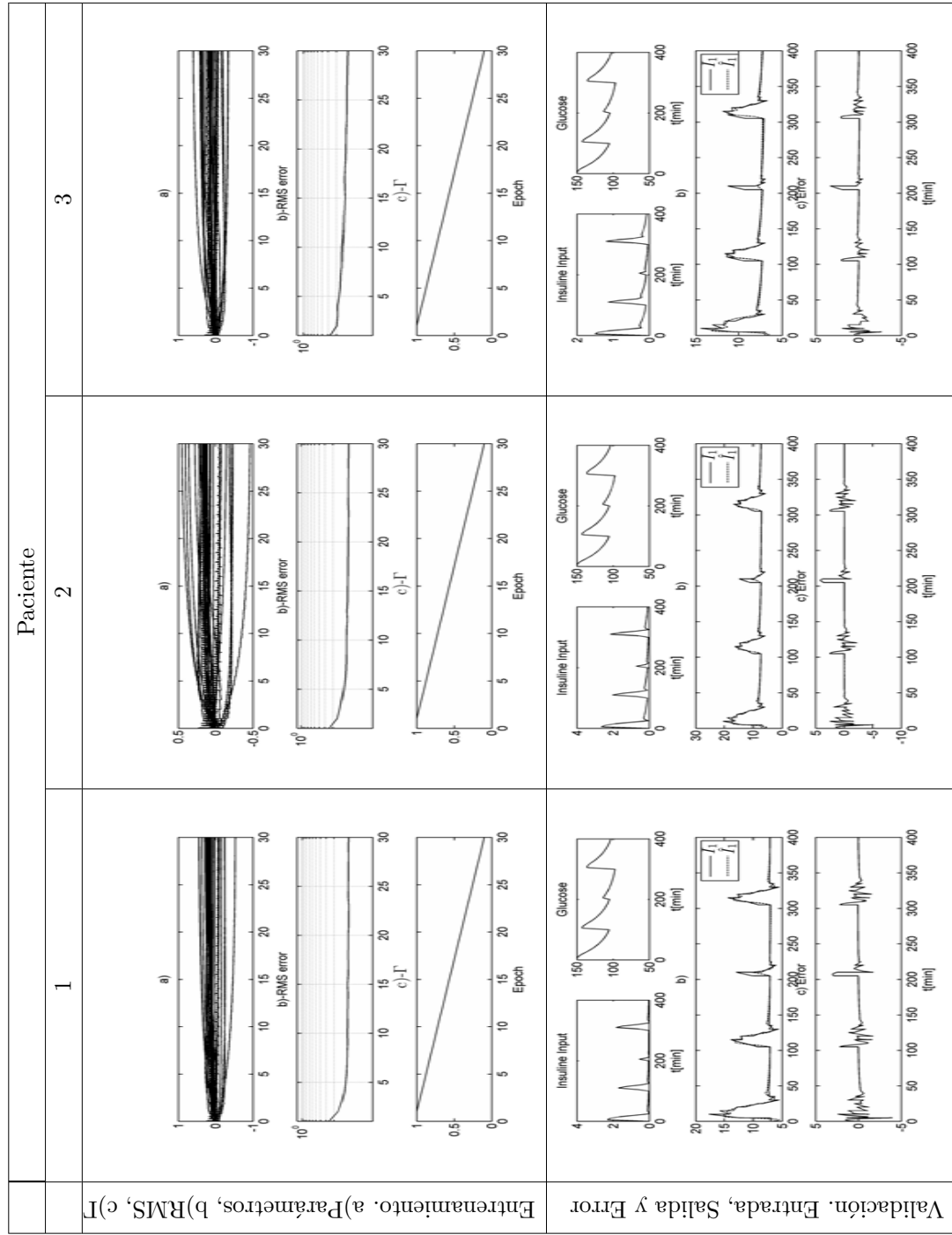


Figura 5.22: Dinámica de la insulina. Identificación con dos entradas (glucosa e insulina) y $T_s = 5 [min]$

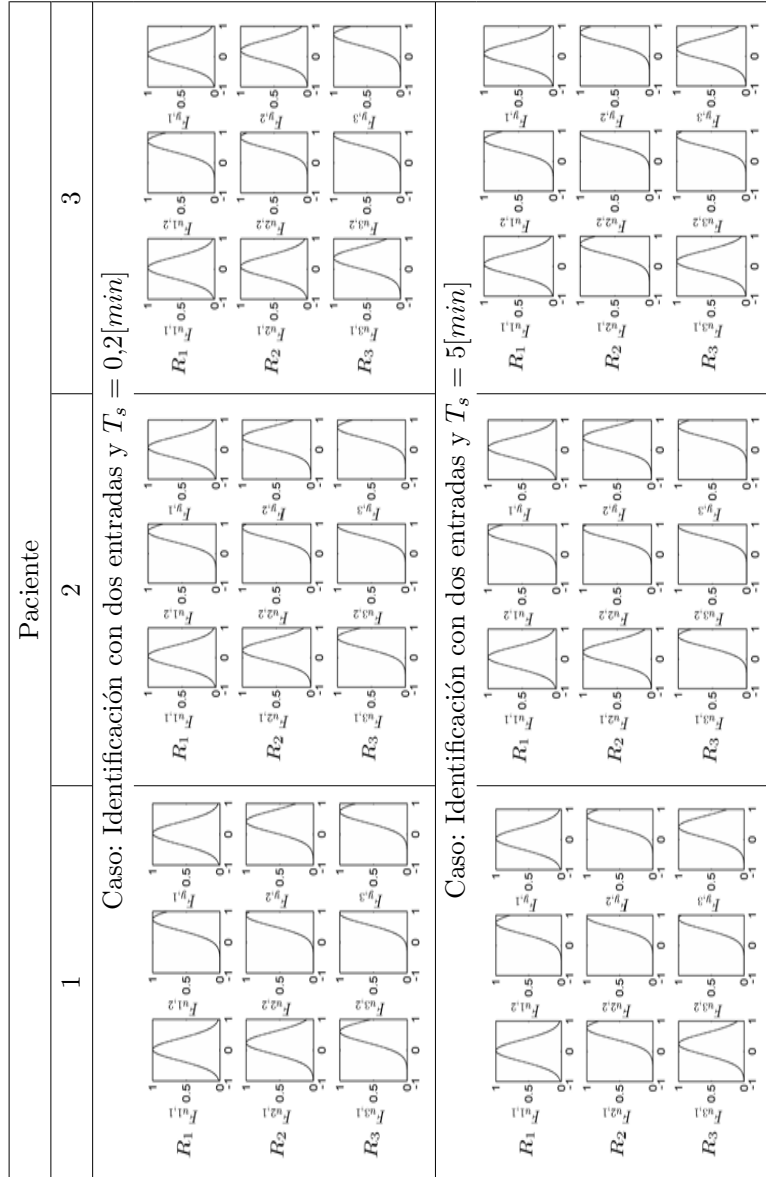


Figura 5.23: Dinámica de la insulina: Conjuntos difusos obtenidos con c-medias difuso para dos entradas (insulina y glucosa)

Capítulo 6

Conclusiones

6.1. Conclusiones

En este trabajo se proponen tres estructuras para identificación de sistemas no lineales, tanto en tiempo discreto como en tiempo continuo, con representación entrada salida y en espacio de estados, basadas en redes neurodifusas recurrentes. En ambos casos, el aprendizaje de la dinámica se lleva a cabo solo mediante mediciones de señales entrada/salida del sistema, y con algún conocimiento del mismo.

En primer lugar, la red en tiempo discreto se diseña con una estructura entrada/salida y reglas difusas tipo Takagi-Sugeno, donde la parte antecedente solo depende de la salida en tiempo actual (no de los retrasos), con la parte consecuente en la forma de sistemas no lineales. Se propone un aprendizaje tanto para los parámetros lineales con respecto a la estructura (parte consecuente), como para los no lineales (parte antecedente) basado en una linealización de los mismos con respecto a valores *ideales* que minimizan el error de identificación. Para evitar que los parámetros antecedentes evolucionen de forma tal que dejen zonas sin cubrir, se propone emplear funciones de restricción no lineales del tipo sigmoide, lo que evita el empleo de algoritmos de proyección. Esta misma red es empleada como modelo del sistema para diseñar un algoritmo de control basado en un control de equivalencia cierta, el cual es sabido en la literatura requiere de componentes de compensación para lograr un desempeño estable. Sin embargo, se muestra, mediante su aplicación a sistemas no lineales, cómo una red neurodifusa recurrente puede ser empleada como el modelo de un sistema no lineal al capturar la dinámica del sistema.

Por su parte, la red neurodifusa en tiempo continuo considera una estructura en variables de estado, donde el objetivo de identificación se restringe a una clase de sistemas no lineales mayor que aquella linealizable a partir de una realimentación (dinámica o estática) de la salida. Dicha restricción, si bien omite varias clases de sistemas, resulta de utilidad para varios sistemas no lineales como los descritos en el capítulo 5.

El algoritmo de entrenamiento de la red en tiempo continuo se basa en teoría desarrollada para observadores adaptables, a su vez conveniente dado el problema de identificación, ya que solo es necesario disponer de las mediciones de entrada y salida del sistema a identificar para lograr un algoritmo desacoplado que calcule los estados y parámetros de una red *ideal* que minimice el error de identificación. De este modo, se logra entrenar a la estructura, aún ante el desconocimiento práctico de los estados reales del sistema y se logra capturar la dinámica del mismo. El algoritmo permite aplicarse por épocas, contando con la demostración de su convergencia y estabilidad con base en la teoría de Lyapunov.

Con el fin de probar la efectividad de la red se muestran diversos sistemas reales donde, a partir de datos experimentales, se logra identificar su dinámica, a saber: un motor de corriente directa con no linealidades asociadas, un robot de un único eslabón flexible, un sistema de tráfico vehicular, un robot de dos grados de libertad en un problema de servo-visual y un sistema insulina-glucosa en sangre en pacientes diabéticos. Se comprueba de este modo la utilidad de la red para la identificación de sistemas de distinta naturaleza, donde dicha red puede ser empleada como un modelo del mismo habiéndose empleado solamente datos entrada/salida de éste.

Finalmente, se propone un método de identificación y control basado en una estructura modificada de la red neurodifusa recurrente en tiempo continuo, donde es necesario un conocimiento más profundo del sistema a identificar. Al suponer que en la operación del sistema y del esquema de control solo se tiene disponible la medición de la salida, se propone un control basado en la teoría de linealización mediante realimentación de salida. Sin embargo, este diseño, al requerir de un mayor conocimiento del sistema hace mucho más restringida a la clase de sistemas a identificar y controlar.

6.2. Líneas de investigación abiertas y trabajo futuro

Existen varios puntos que deben de ser analizados con respecto a cada una de las estructuras propuestas. En primer lugar, si bien en el caso discreto se propone un método de sintonización de parámetros, en ninguno de los dos casos este aprendizaje es llevado a cabo durante el propio algoritmo de entrenamiento. En el caso particular de redes estáticas, se han propuesto diversas estructuras que, en conjunción con el entrenamiento de parámetros, llevan a cabo modificaciones en la estructura, buscando equilibrar el desempeño y el número de parámetros empleado (por ejemplo, ANFIS, propuesto por Jang (1993)); lo cual ha sido investigado también en redes dinámicas, empleando algoritmos genéticos (Juang & Lin, 1999). Por otra parte, el algoritmo presentado no considera explícitamente alguna función de optimización, y opera en forma similar a los algoritmos basados en el gradiente.

Por lo anterior, se considera como trabajo futuro el proponer algoritmos de aprendizaje tanto de parámetros como de estructura, y explorar otras técnicas de entrenamiento, analizando igualmente las restricciones no lineales como forma de evitar la implementación de algoritmos de proyección.

En el caso particular de la red en tiempo discreto, se propone un algoritmo de adaptación de los parámetros antecedentes, de naturaleza no lineal, mientras que en el caso continuo únicamente se lleva a cabo el aprendizaje de los parámetros consecuentes (lineales). En el segundo caso, la adaptación de los parámetros antecedentes podría mejorar en mayor medida las capacidades de aproximación de la red. De igual forma, cabe investigar sobre diversos tipos de estrategias de aprendizaje que permitan una convergencia más rápida, preservando las propiedades de convergencia y estabilidad.

Si bien las estructuras mostradas resultaron ser efectivas para identificar una clase de sistemas no lineales, es aún poco el trabajo presentado con respecto a su utilización como un modelo matemático que permita emplearlas como base para diseño de controladores (los diseños mostrados son, ya sea restrictivos en el caso continuo, o en cierta forma *sencillos* en el caso del control de equivalencia cierta). Cabe aún realizar investigación sobre técnicas de control que exploten la propia estructura, diferentes a las del control de equivalencia cierta, que garanticen una mejor convergencia y estabilidad, incluso para problemas de seguimiento.

Finalmente, si bien en los ejemplos físicos mostrados permitieron evaluar la capacidad de la red para identificar la dinámica de sistemas reales, falta explotar al sistema identificado para emplearlo en estrategias de diseño, tanto de control como de detección de fallas. En el caso particular del sistema de insulina-glucosa en pacientes diabéticos, es necesario probar su desempeño empleando datos obtenidos directamente de estudios realizados en pacientes y analizar su desempeño para el cálculo en línea de la concentración de insulina en sangre una vez entrenada.

6.3. Publicaciones generadas

A continuación se enumeran los productos de esta investigación, divididos en artículos en revistas indexadas, colaboraciones en libros, y presentaciones en congresos nacionales e internacionales.

Artículos en revistas indexadas:

- Berrospe, Édgar; González-Olvera, Marcos A. y Tang, Yu. *Identification and Speed Control of a DC Motor Using an Input-Output Recurrent Neurofuzzy Network*. Special Issue: Advances in Computational Intelligence - Selected papers from the Second International Workshop on Advanced Computational Intelligence. *Enviado. En espera de decisión**.
- Gonzalez-Olvera, Marcos A; Gallardo, Ana, Tang, Yu; Revilla-Monsalve, María C e Islas-Andrade, Sergio. *Black-Box Modeling of Insulin Dynamics Using a Discrete-time Recurrent Neurofuzzy Network with an Experimental Framework*. Special Issue: Advances in Computational Intelligence - Selected papers from the Second International Workshop on Advanced Computational Intelligence. *Enviado. En espera de decisión**.
- Gonzalez-Olvera, Marcos Angel; Tang, Yu. *Identification of a Class of Nonlinear Systems by a Recurrent Neurofuzzy Network*. IEEE Trans. on Neural Networks. 2009. 2a revisión enviada, en espera de decisión.*
- Gonzalez-Olvera, Marcos Angel; Tang, Yu y Álvarez-Icaza, Luis. *Modeling of a Traffic Cell Based on a Recurrent-Neural Network*. International Journal of Modelling, Identification and Control (IJMIC). 2009. A publicarse.

- Gonzalez-Olvera, Marcos Angel; Tang, Yu. *A New Recurrent Neurofuzzy Network for Identification of Dynamic Systems*. Fuzzy Sets and Systems. Vol. 158. No. 10. pp 1023-1035. Elsevier 2007.

Capítulo en libro:

- González-Olvera, Marcos y Tang, Yu. *Chapter 10: Identification and Control of Nonlinear Dynamic Systems via a Constrained Input-Output Neurofuzzy Network*, en “Advances in Control Systems Theory and Applications”. Jing Sun, Gang Tao, Ed. ISBN 978-7-312-02238-8. 382pag. University of Science and Technology of China.

Artículos en congresos nacionales e internacionales:

- Gonzalez-Olvera, Marcos A; Tang, Yu. *Discrete-Time Recurrent Neurofuzzy Network for Identification of Nonlinear Systems*. Congreso Nacional de la Asociación de México de Control Automático (AMCA) 2009. Zacatecas, Zacatecas. *Enviado. En espera de decisión**.
- Gonzalez-Olvera, Marcos A; Gallardo, Ana, Tang, Yu; Revilla-Monsalve, María C e Islas-Andrade, Sergio. *Black-Box Modeling of Insulin Dynamics Using Recurrent Neurofuzzy Networks*. International Workshop on Advanced Computational Intelligence, 2009.
- Berrospe, Édgar; González-Olvera, Marcos A. y Tang, Yu. *Identification and Speed Control of a DC Motor Using an Input-Output Recurrent Neurofuzzy Network*. International Workshop on Advanced Computational Intelligence, 2009.
- Gonzalez-Olvera, Marcos A; Rodríguez-Morales, Ángel L. y Tang, Yu. *Black-Box Modeling of a 2-DOF Manipulator in the Image Plane Using Recurrent Neurofuzzy Networks*. Control and Decision Conference 2009.
- Gonzalez-Olvera, Marcos Angel; Tang, Yu. *Modeling, Identification and Control Based on Recurrent Neural Networks of a Class of Nonlinear Systems*. 15th IFAC Symposium on System Identification, SYSID 2009. San-Malo, Francia.
- Gonzalez-Olvera, Marcos Angel; Tang, Yu. *Identification of a Class of Nonlinear Systems by a Continuous-Time Recurrent Neurofuzzy Network*. American Control Conference 2009, San Luis, Misuri, E.U.A.

6. CONCLUSIONES

- Gonzalez-Olvera, Marcos Angel; Tang, Yu. *Nonlinear System Identification and Control Using an Input-Output Recurrent Neurofuzzy Network*. 17th IFAC World Congress, 2008. Seúl, Corea.
- Gonzalez-Olvera, Marcos Angel; Tang, Yu. *Velocity Estimation Using Adaptive Observers*. Congreso Nacional de la Asociación de México de Control Automático (AMCA) 2008. Cd. de México.
- Gonzalez-Olvera, Marcos Angel; Tang, Yu. *Identificación de sistemas dinámicos por una red neurodifusa recurrente E/S mediante mínimos cuadrados*. Congreso Nacional de la Asociación de México de Control Automático (AMCA) 2007. Monterrey, Nuevo León.

Apéndice A

Teoremas empleados

La teoría siguiente es tomada del trabajo de Marino & Tomei (1996).

A.1. Transformación a realimentación de salida no lineal

Considérese el sistema de múltiples entradas y una salida

$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x}, u), \quad \mathbf{x} \in \mathfrak{R}^n, u \in \mathfrak{R}^m, \\ y &= h(\mathbf{x}), \quad y \in \mathfrak{R}\end{aligned}\tag{A.1}$$

con f y h campos vectoriales suaves en \mathfrak{R}^n , $g(\mathbf{x}, 0) = 0 \in \mathfrak{R}^n$ $h : \mathfrak{R}^n \rightarrow \mathfrak{R}$ con $h(0) = 0$.

Teorema A.1. *Existe un difeomorfismo local en una vecindad del origen $\mathbf{z} = T(\mathbf{x})$, $T(0) = 0$, $\mathbf{z} \in \mathfrak{R}^n$, que transforma al sistema (A.4) en*

$$\begin{aligned}\dot{\mathbf{z}} &= \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} \mathbf{z} + \begin{bmatrix} \gamma_1(y, u) \\ \gamma_2(y, u) \\ \vdots \\ \gamma_n(y, u) \end{bmatrix} \triangleq A_0 \mathbf{z} + \gamma(y, u) \\ y &= \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \end{bmatrix} \mathbf{z} \triangleq C_0 \mathbf{z}\end{aligned}\tag{A.2}$$

si y solo si, las siguientes condiciones se cumplen en una vecindad U_0 del origen:

1. $\rho\{dh, d(L_f h), d(L_f^2 h), \dots, d(L_f^{n-1} h)\} = n$,
2. $[ad_f^i \mathbf{r}, ad_f^j \mathbf{r}] = 0$, $0 \leq i, j \leq n-1$,

$$3. [g, ad_f^j \mathbf{r}] = 0, j \in [0, n-2], \forall u \in \mathfrak{R}^m,$$

con \mathbf{r} un campo vectorial solución del sistema

$$\begin{bmatrix} \langle dh, \mathbf{r} \rangle \\ \vdots \\ \langle d(L_f^{n-1}h, \mathbf{r}) \rangle \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix}. \quad (\text{A.3})$$

El difeomorfismo es global si además las condiciones (1)-(3) se cumplen en R^n y los campos vectoriales $ad_f^i \mathbf{r}$, $i \in [0, n-1]$ son completos.

A.2. Control por realimentación de salida dinámica

Considérese el sistema de una entrada y una salida

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x})u, \quad \mathbf{x} \in \mathfrak{R}^n, u \in \mathfrak{R}, \\ y &= h(\mathbf{x}), \quad y \in \mathfrak{R} \end{aligned} \quad (\text{A.4})$$

donde \mathbf{x} es el vector de estados, u el control, $h : \mathfrak{R}^n \rightarrow \mathfrak{R}$ es una función suave, $h(0) = 0$, f y g son funciones suaves en R^n , con $f(0) = 0$, $g(0) \neq 0$, con (A.2) observable con $u = 0$.

Teorema A.2. *El sistema (A.2) puede ser localmente linealizado por una realimentación de salida estática si y solo si en un vecindario del origen U_0 se cumple*

1. $\rho\{d(L_f^j h) : j \in [0, n-1]\} = n$,
2. $[ad_f^i \mathbf{r}, ad_f^j \mathbf{r}] = 0, 0 \leq i, j \leq n-1$,
3. $[g, ad_f^k \mathbf{r}] = 0, k \in [0, n-2]$,
4. \exists una función suave $\sigma : \mathfrak{R}^n \rightarrow \mathfrak{R}$ y n constantes reales $b_i, i \in [1, n]$ tal que $g = (\sigma \circ h) \sum_{j=1}^n b_{n-j+1} ad_{-f}^{j-1} \mathbf{r}$,
5. \exists una función suave $\alpha' : \mathfrak{R} \rightarrow \mathfrak{R}$ y n constantes reales $a_i, i \in [1, n]$ tal que $ad_{-f}^n f = -\sum_{i=1}^n a_{n-i+1} ad_{-f}^{i-1} \mathbf{r} + (\alpha' \circ h) \sum_{i=1}^n b_{n-i+1} ad_{-f}^{n-1} \mathbf{r}$,

con \mathbf{r} un campo vectorial solución del sistema

$$\begin{bmatrix} \langle dh, \mathbf{r} \rangle \\ \vdots \\ \langle d(L_f^{n-1}h, \mathbf{r}) \rangle \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix}. \quad (\text{A.5})$$

Si además estas condiciones se satisfacen en \mathfrak{R}^n y los campos vectoriales $ad_f^i \mathbf{r}$, $i \in [0, n-1]$ son completos, entonces el sistema (A.2) es globalmente linealizable por una realimentación de salida estática.

Este resultado implica que el sistema (A.2) puede ser transformado mediante un difeomorfismo $\mathbf{z} = T(\mathbf{x})$ en

$$\begin{aligned}\dot{\mathbf{z}} &= A_0 \mathbf{z} + \beta(y)u + \phi(y) \\ y &= C_0 \mathbf{z}\end{aligned}\tag{A.6}$$

donde

$$\beta(y) = \sigma(y) \begin{bmatrix} b_n \\ \vdots \\ b_2 \\ b_1 \end{bmatrix},\tag{A.7}$$

de donde es posible calcular

$$g = \sigma(y) \sum_{i=1}^n b_{n-i+1} \frac{\partial}{\partial z_i}.\tag{A.8}$$

Con base en el teorema anterior, es posible generalizar el concepto a un difeomorfismo dinámico, donde el control sea a su vez dinámico. En este sentido, se define a un sistema linealizable por realimentación de salida dinámica como aquel para el cual existe una transformación filtrada por realimentación de salida

$$\dot{\xi} = \Delta \xi + \delta(y(t)), \quad \xi(0) = \xi_0,\tag{A.9}$$

$$u = k(y, \xi) + \beta(y)\nu,\tag{A.10}$$

$$\mathbf{z} = T(\mathbf{x}, \xi(t)), \quad \mathbf{x}, \mathbf{z} \in \mathfrak{R}^n, \quad \xi \in \mathfrak{R}^s,\tag{A.11}$$

donde $\beta(y) \neq 0$, $\forall y \in \mathfrak{R}$, de forma que la dinámica en las coordenadas z sea

$$\begin{aligned}\dot{\mathbf{z}} &= A_c \mathbf{z} + a y + b \nu \\ y &= C_c \mathbf{z}\end{aligned}\tag{A.12}$$

con a y b vectores constantes de $n \times 1$.

Referencias

- ANDERBERG, M. (1973). Cluster analysis for applications. *New York*. 27
- ANDERSON, B. & MOORE, J. (1979). *Optimal filtering*. Prentice-Hall Englewood Cliffs, NJ. 63
- AREM, B.V. & HOGEMA, J. (1995). The microscopic simulation model mixic 1.2. Tech. rep., Delft. 95
- ASOCIATION, A.D. (2009). *Standards of Medical Care in Diabetes*. ADA. 106
- BACK, A.D. & TSOI, A.C. (1990). A time series modeling methodology using fir and iir synapses. *Proc. Workshop on Neural Networks for Statistical and Economic Data*, 187–194. 7
- BASTIN, G. & GEVERS, M. (1988). Stable adaptive observers for nonlinear time-varying systems. *Automatic Control, IEEE Transactions on*, **33**, 650–658. 29, 60
- BENGIO, Y., SIMARD, P. & FRASCONI, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, **5**, 157–166. 22
- BERENJI, H., KHEDKAR, P., CENTER, N. & VIEW, M. (1992). Learning and tuning fuzzy logic controllers through reinforcements. *IEEE Transactions on Neural Networks*, **3**, 724–740. 6
- BERGMAN, R.N., IDER, Y.Z., BOWDEN, C.R. & COBELLI, C. (1979). Quantitative estimation of insulin sensitivity. *Am. J. Physiology, Endocrinology and Metabolism*, **235**, E667–E677. 106, 107
- BERROSPE, E., GONZALEZ-OLVERA, M. & TANG, Y. (2009). Identification and speed control of a dc motor using an input-output recurrent neurofuzzy network. In *International Workshop on Advanced Computational Intelligence*. 10
- BEZDEK, J. (1981). Pattern Recognition with Fuzzy Objective Function. *Ed. Plenum Press, NY, USA*. 6, 27, 28
- BOLIE, V. (1961). Coefficients of normal blood glucose regulation. *Journal of Applied Physiology*, **16**, 783–788. 106
- CHAUMETTE, F. & HUNTHINSON, S. (2006). Visual servo control part i: Basic approaches. *IEEE Robotics & Automation Magazine*, 82–90. 102
- CHEN, C. & KAK, A. (1989). A robot vision system for recognizing 3D objects in low-order polynomial time. *IEEE Transactions on Systems, Man and Cybernetics*, **19**, 1535–1563. 102
- CHIU, S.L. (1994). Fuzzy model identification based on cluster estimation. *Journal of Intelligent and Fuzzy Systems*, **2**, 267–278. 6, 28
- COMBASTEL, C., ZHANG, Q. & LALAMI, A. (2006). Fault Diagnosis based on the Enclosure of Parameters Estimated with an Adaptive Observer. *Sixth IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*. 86
- CYBENKO, G. (1989). Approximation by superpositions of a sigmoidal function. *Math. Con. Signal Syst.*, **2**, 303–314. 5, 19
- DAGANZO, C. (1994). The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation research. Part B: methodological*, **28**, 269–287. 94, 95
- DE JESÚS RUBIO, J. & YU, W. (2007). Nonlinear system identification with recurrent neural networks and dead-zone Kalman filter algorithm. *Neurocomputing*, **70**, 2460–2466. 25
- DRIANKOV, D., HELLENDOORN, H. & REINFRANK, M. (1996). *An introduction to fuzzy control*. Springer. 17
- DUNN, J. (1973). A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, **3**, 32–57. 6, 28
- FU, K., GONZALEZ, R. & LEE, C. (1987). *Robotics: control, sensing, vision, and intelligence*. McGraw-Hill, Inc. New York, NY, USA. 102

REFERENCIAS

- GALLARDO-HERNANDEZ, G., FRIDMAN, L., ISLAS-ANDRADE, S. & SHTESSEL, Y. (2008). Quasi-continuous high order sliding modes controllers applied to glucose-insulin regulatory system models. In *Proceedings of Decision and Control Conference*. 106, 108, 109, 110
- GONZALEZ OLVERA, M. & TANG, Y. (2007). A new recurrent neurofuzzy network for identification of dynamic systems. *Fuzzy Sets and Systems*, **158**, 1023–1035. 9, 37, 44
- GONZALEZ-OLVERA, M. & TANG, Y. (2009a). Identification of a class of nonlinear systems by a continuous-time recurrent neurofuzzy network. In *American Control Conference 2009. ACC '09*. 10
- GONZALEZ-OLVERA, M. & TANG, Y. (2009b). Identification of a class of nonlinear systems by a recurrent neurofuzzy network. *IEEE Trans. on Neural Networks*. In revision.. 10
- GONZALEZ-OLVERA, M. & TANG, Y. (2009c). Modeling, identification and control based on recurrent neural networks of a class of nonlinear systems. In *15th IFAC Symposium on System Identification SYSID 2009. Saint-Malo, France..* 10
- GONZALEZ-OLVERA, M., RODRIGUEZ-MORALES, A.L. & TANG, Y. (2009a). Black-box modeling of a 2-dof manipulator in the image plane using recurrent neurofuzzy networks. In *Control and Decision Conference*. 10
- GONZALEZ-OLVERA, M., TANG, Y., GALLARDO-HERNÁNDEZ, A.G., REVILLA-MONSALVE, M.C. & ISLAS-ANDRADE, S. (2009b). Black-box modeling of insulin dynamics using recurrent neurofuzzy networks. In *International Workshop on Advanced Computational Intelligence*. 10
- GONZALEZ-OLVERA, M.A. & TANG, Y. (2007). Identificación de sistemas dinámicos por una red neurodifusa recurrente e/s mediante mínimos cuadrados. In *Memorias del Congreso AMCA*. 9
- GONZALEZ-OLVERA, M.A. & TANG, Y. (2008a). Identification and Control of Nonlinear Dynamic Systems via a Constrained Input-Output Neurofuzzy Network. *Advances in Control Systems Theory and Applications (Edited by G. Tao and J. Sun)*. 10, 21, 47
- GONZALEZ-OLVERA, M.A. & TANG, Y. (2008b). Nonlinear system identification and control using an input-output recurrent neurofuzzy network. In M.P. Chung Myung Jin, ed., *Proceedings of the 17th IFAC World Congress*, vol. 17. 9
- GONZALEZ-OLVERA, M.A. & TANG, Y. (2008c). Velocity estimation using adaptive observers. In *Memorias del Congreso AMCA*. 10
- GONZALEZ-OLVERA, M.A. & TANG, Y. (2009d). *Advances in Control Systems Theory and Applications*, vol. 1, chap. Identification and Control of Nonlinear Dynamic Systems via a Constrained Input-Output Neurofuzzy Network, 191–210. University of Science and Technology of China. 22
- GONZALEZ-OLVERA, M.A. & TANG, Y. (2009e). Modeling of a traffic cell based on a recurrent neurofuzzy network. *International Journal of Modelling, Identification and Control*. 10
- GORRINI, V. & BERSINI, H. (1994). Recurrent fuzzy systems. *Proc. IEEE Int. Conference on Fuzzy Systems*, 193–198. 6, 20
- GUO, L. (1990). Estimating time-varying parameters by the Kalman filter based algorithm: stability and convergence. *Automatic Control, IEEE Transactions on*, **35**, 141–147. 25
- GUYTON, A. & HALL, J. (2000). *Textbook of Medical Physiology*. W. B. Saunders. 106
- HAGAN, M. & MENHAJ, M. (1994). Training feed-forward networks with the Marquardt algorithm. *IEEE transactions on Neural Networks*, **5**, 989–993. 22
- HAGNER, D.G., HASSOUN, M.H. & WATTA, P.B. (2000). *Comparison of Recurrent Networks for Trajectory Generation*, chap. 10, 243–276. CRC Press. 25
- HAYKIN, S. (2001). *Kalman Filtering and Neural Networks*. Wiley Chichester. 7, 25
- HELBING, D. (1996). Gas-kinetic derivation of navier-stokes-like traffic equations. *Physical Review*, **E53**, 2266–2381. 94
- HO, F. & IOANNOU, P. (1996). Traffic flow modeling and control using artificial neural networks. *Control Systems Magazine, IEEE*, **16**, 16–26. 96

- HOPFIELD, J. (1982). Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proceedings of the National Academy of Sciences*, **79**, 2554–2558. 6
- HOVORKA, R., CANONICO, V., CHASSIN, L., HAUETER, U., MASSI-BENEDETTI, M., FEDERICI, M., PIEBER, T., SCHALLER, H., SCHAUPP, L., VERING, T. & WILINSKA, M. (2004). Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiological Measurements*, **25**, 905–920. 106
- IOANNOU, P. & SUN, J. (1996). *Robust adaptive control*. PTR Prentice-Hall Upper Saddle River, NJ. 29
- ISIDORI, A. (1995). *Nonlinear control systems*. Springer. 91
- JANG, J.S.R. (1993). Anfis: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man and Cybernetics*, **23**, 665–685. 6, 18, 37, 119
- JOHANSSON, R. (1993). *System Modeling and Identification*. Prentice Hall Information and System Sciences, Prentice Hall, 1st edn. 7, 24, 39
- JUANG, C.F. (2002). A tsf-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms. *IEEE Transactions of Fuzzy Systems*, **10**, 155–170. 21, 44
- JUANG, C.F. & LIN, C.T. (1999). A recurrent self-organizing neural fuzzy inference network. *IEEE Trans. Neural Networks*, **10**, 828–845. 7, 21, 119
- KAVEH, P. & SHTESSEL, Y. (2007). Blood glucose regulation using higher-order sliding mode control. *Int. J. of Robust and Nonlinear Control*, **18**, 557–569. 106, 108
- KELLY, R. (1996). Robust asymptotically stable visual servoing of planar robots. *IEEE Transactions on Robotics and Automation*, **12**, 759–766. 102, 104
- KEMPF, R. & ADAMY, J. (2003). Equilibria of recurrent fuzzy systems. *Fuzzy Sets and Systems*, **140**, 231–257. 22
- KOHONEN, T. (2001). *Self-organizing maps*. Springer. 28
- KOSKO, B. (1992). *Fuzzy Engineering*. Prentice Hall. 17, 22, 38
- KREISSELMEIER, G. (1977). Adaptive observers with exponential rate of convergence. *IEEE Transactions on Automatic Control*, **22**, 2–8. 29
- LARSEN, KRONENBERG, MELMED & POLONSKY (2003). *Williams Textbook of Endocrinology*. Saunders, USA. 107
- LECUN, Y., BOSER, B., DENKER, J., HENDERSON, D., HOWARD, R., HUBBARD, W. & JACKEL, L. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, **1**, 541–551. 5
- LIN, C. & HSU, C. (2001). Identification of dynamic systems using recurrent fuzzy neuralnetwork. In *IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th*, 2671–2675. 20
- LIN, C.T. (1996). *Neural Fuzzy Systems: a neuro-fuzzy synergism to intelligent systems*. Prentice Hall, 1st edn. 20, 21
- LJUNG, L. (1986). *System identification: theory for the user*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA. 2
- LÜDERS, G. & NARENDRA, K. (1973). An adaptive observer and identifier for a linear system (Asymptotically stable adaptive observer and identification scheme for linear system, using Liapunov canonical state representation). *IEEE Transactions on Automatic Control*, **18**, 496–499. 29
- MAN, C.D., RIZZA, R. & COBELLI, C. (2007). Meal simulation model of the glucose-insulin system. *IEEE Transaction on Biomedical Engineering*, **54**, 1740–1749. 106
- MARINO, R. (1990). Adaptive observers for single output nonlinear systems. *Automatic Control, IEEE Transactions on*, **35**, 1054–1058. 58, 60, 68, 78
- MARINO, R. & TOMEI, P. (1993). Global adaptive output-feedback control of nonlinear systems. I. Linear parameterization. *Automatic Control, IEEE Transactions on*, **38**, 17–32. 70, 78, 91
- MARINO, R. & TOMEI, P. (1996). *Nonlinear control design: geometric, adaptive and robust*. 29, 58, 80, 81, 123

REFERENCIAS

- MASTOROCOSTAS, P.A. & THEOCHARIS, J.B. (2002). A recurrent fuzzy-neural model for dynamic system identification. *IEEE Trans. on Systems, Man and Cybernetics - Part B: Cybernetics*, **32**, 176–190. 7
- MCCULLOCH, W. & PITTS, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biology*, **5**, 115–133. 5
- MINDERHOUD, M.M. (1999). Supported driving: Impacts on motorway traffic flow. Dissertation thesis, Delft University of Technology. 95
- NAGEL, K. (1996). Particle hopping models and traffic flow theory. *Physical Review*, **E53**, 4655–4672. 94
- NARENDRA, K. & PARTHASARATHY, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Tras. Neural Networks*, **1**, 4–27. 5, 7, 42, 63
- NAUCK, D., NURNBERGER, A., BT, R. & VENTURING, I. (2005). The evolution of neuro-fuzzy systems. In *Fuzzy Information Processing Society, 2005. NAFIPS 2005. Annual Meeting of the North American*, 98–103. 6, 18, 20
- NELLES, O. (2001). *Nonlinear System Identification*. Springer, Berlin. 3, 4, 17, 18, 25, 35
- NURNBERGER, A. (2001). A hierarchical recurrent neuro-fuzzy system. In *IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th*, vol. 3. 21
- NÜRNBERGER, A. & KRUSE, R. (2002). A neuro-fuzzy approach to optimize hierarchical recurrent fuzzy systems. *Fuzzy Optimization and Decision Making*, **1**, 221–248. 21
- PAPAGEORGIU, M. (1990). Modelling and real-time control of traffic flow on the southern park of Boulevard Peripherique in Paris: Part II: Coordinated on-ramp metering. *TRANSP. RES.*, **24**, 361–370. 96
- PARKER, R., DOYLE III, F., WARD, J. & PEPPAS, N. (2004). Robust h_{∞} glucose control in diabetes using a physiological model. *American Institute of Chemical Engineers (AIChE)*, **46**, 2537–2549. 106
- PASSINO, K.M. (1998). *Fuzzy Control*. Addison Wesley Longman, Menlo, CA. 6, 7
- PAYNE, H. (1971). Models of freeway traffic and control. *Mathematical Models of Public Systems*, **1**, 51–61. 94
- PIPES, L.A. (1953). An operational analysis of traffic dynamics. *Journal of Applied Physics*, **24**, 274–287. 95
- POZNYAK, A., SANCHEZ, E. & YU, W. (2001). *Differential Neural Networks for Robust Nonlinear Control: Identification, State Estimation and Trajectory Tracking*. World Scientific. 21
- POZNYAK, A., POZNYAK, T. & CHAIREZ, I. (2006). Dynamic neural observers and their application for identification and purification of water by ozone. *Automation and Remote Control*, **67**, 887–899. 7, 21, 29
- RODRIGUEZ, L. & TANG, Y. (2009). Dynamic visual servoing of robot manipulators based on passivity. Internal report, Faculty of Engineering, UNAM. 104, 105
- ROSAS-JAIMES, O. & ALVAREZ-ICAZA, L. (2007). Vehicle density and velocity estimation on highways for on-ramp metering control. *Nonlinear Dynamics*, **49**, 555–566. 96, 98, 99
- ROSENBLATT, F. (1957). *The Perceptron, a Perceiving and Recognizing Automaton Project Para*. Cornell Aeronautical Laboratory. 5
- RUANO, A. (2005). *Intelligent control systems using computational intelligence techniques*. Iet. 22
- RUBIO, J. & YU, W. (2007). Learning and Approximation-Neural Networks Training with Optimal Bounded Ellipsoid Algorithm. *Lecture Notes in Computer Science*, **4491**, 1173–1182. 7, 21, 24, 25
- RUMELHART, D., HINTON, G. & WILLIAMS, R. (1986). *Learning Internal Representations by Error Propagation*, chap. 8. MIT Press, Cambridge. 5, 7
- SORENSEN, J.T. (1985). *A physiologic model of glucose metabolism in man and its use to design and assess improved insulin therapies for diabetes*. Ph.D. thesis, Massachusetts Institute of Technology. Dept. of Chemical Engineering. 106, 107
- SPONG, M. (1989). *Robot Dynamics and Control*. John Wiley & Sons, Inc. New York, NY, USA. 104

- TANAKA, K. & WANG, H. (2001). *Fuzzy control systems design and analysis: a linear matrix inequality approach*. Wiley-Interscience. 18, 19
- THEOCHARIS, J. & VACHTSEVANOS, G. (1996a). Adaptive fuzzy neural networks as identifiers of discrete-time nonlinear dynamic systems. *Journal of Intelligent and Robotic Systems*, **17**, 119–168. 20
- THEOCHARIS, J. & VACHTSEVANOS, G. (1996b). Recursive learning algorithms for training fuzzy recurrent models. *International Journal of Intelligent Systems*, **11**, 1059–1098. 20
- WAN, E.A. (1990). Temporal back-propagation for fir neural networks. *Proc. of the International Joint Conference on Neural Networks*, 575–580. 7
- WANG, L.X. (1994). *Adaptive Fuzzy Systems and Control*. Prentice Hall. 6, 18, 19, 33, 51, 78, 80
- WEIBEL, A., HANAZAWA, T., HINTON, G., SHIKANO, K. & LANG, K.J. (1989). Phenomene recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **37**, 328–339. 7
- WERBOS, P. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral sciences*. Harvard University. 5, 22, 25
- WILINSKA, M., CHASSIN, L.J., SCHALLER, H., SCHAUPP, L., PIEBER, T. & HOVORKA, R. (2005). Insulin kinetics in type-1 diabetes: continuous and bolus delivery/insulin kinetics in type-1 diabetes: continuous and bolus delivery. *Biomedical Engineering, IEEE Transactions*, **52-1**, 3–12. 106
- WILLIAMS, R.J. & ZIPSER, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, **2**, 270–280. 7
- WILSON, W., HULLS, C. & BELL, G. (1996). Relative end-effector control using Cartesian position based visual servoing: Special section on vision-based control of robot manipulators. *IEEE Transactions on Robotics and Automation*, **12**, 684–696. 102
- XU, A. & ZHANG, Q. (2004). Nonlinear system fault diagnosis based on adaptive estimation. *Automatica*, **40**, 1181–1193. 29
- YASDI, R. (1999). Prediction of Road Traffic using a Neural Network Approach. *Neural Computing & Applications*, **8**, 135–142. 96
- YASUNOBU, S. & MIYAMOTO, S. (1985). Automatic train operation system by predictive fuzzy control. *Industrial Applications of Fuzzy Control*, 1–18. 6
- YU, E. & CHEN, Y. (1993). Traffic prediction using neural networks. *Global Telecommunications Conference, 1993, including a Communications Theory Mini-Conference. Technical Program Conference Record, IEEE in Houston. GLOBECOM'93., IEEE*, 991–995. 96
- YU, W. & LI, X. (2004). Fuzzy identification using fuzzy neural network. *IEEE Trans. on Fuzzy Systems*, **12**. 7, 35
- ZADEH, L. (1965). Fuzzy sets. *Information and control*, **8**, 338–353. 5, 14
- ZADEH, L. (1973). Outline of a new approach to the analysis of complex systems and decision-making approach. *IEEE Trans. on Systems, Man, and Cybernetics SME-3 (1)*, 28–45. 5
- ZHANG, J. & MORRIS, A. (1999). Recurrent neuro-fuzzy networks for nonlinear process modeling. *IEEE Transactions on Neural Networks*, **10**, 313–326. 20, 21
- ZHANG, Q. (2002). Adaptive observer for multiple-input-multiple-output (MIMO) linear-time-varying systems. *Automatic Control, IEEE Transactions on*, **47**, 525–529. 59, 60, 63
- ZHANG, Q. (2005). Revisiting different adaptive observers through a unified formulation. In *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05*, 3067–3072. 29
- ZHANG, Q. (2009). *Advances in Control Systems: Theory and Applications*, chap. 11: From Adaptive Observers to Decoupled State and Parameter Estimations. USTC Press. 30