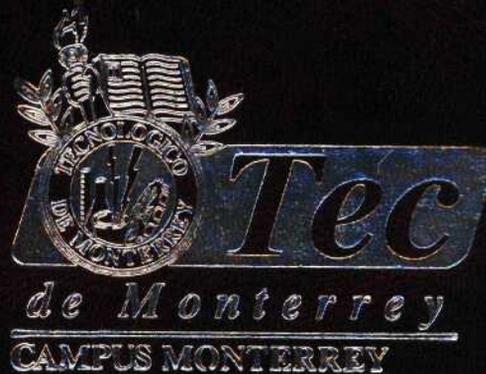


Manejo de incertidumbre por medio de Sistemas de
Lógica Difusa tipo 2 para la modelación de sistemas
no lineales



TESIS

Maestría en Ciencias en Sistemas Inteligentes

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY

Por:

Ing. Germán Montalvo Maldonado

Febrero 2003

© Germán Montalvo Maldonado, 2003

Manejo de incertidumbre por medio de
Sistemas de Lógica Difusa tipo 2 para la
modelación de sistemas no lineales



T E S I S

Maestría en Ciencias en Sistemas Inteligentes

Instituto Tecnológico y de Estudios Superiores de Monterrey

Por

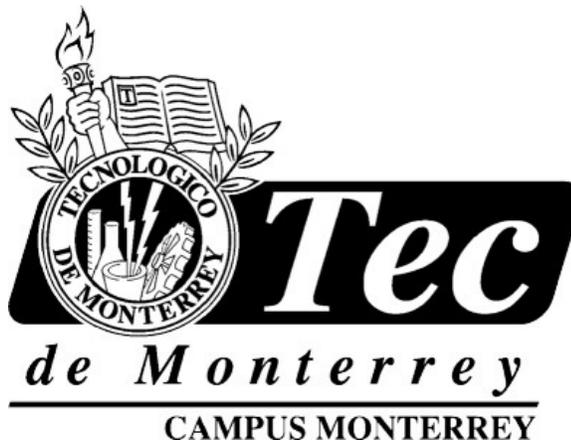
Ing. Germán Montalvo Maldonado

Febrero 2003

Manejo de incertidumbre por medio de Sistemas de Lógica Difusa tipo 2 para la modelación de sistemas no lineales

Por

Ing. Germán Montalvo Maldonado



TESIS

Presentada a la División de Electrónica, Computación, Información y Comunicaciones
Este trabajo es requisito parcial para obtener el grado académico de Maestro en Ciencias
en Sistemas Inteligentes

Instituto Tecnológico y de Estudios Superiores de Monterrey
Campus Monterrey

Monterrey, N.L. Febrero de 2003

**Manejo de incertidumbre por medio de
Sistemas de Lógica Difusa tipo 2 para la
modelación de sistemas no lineales**

TESIS

**Maestría en Ciencias en
Sistemas Inteligentes**

Instituto Tecnológico y de Estudios Superiores de Monterrey

Por

Ing. Germán Montalvo Maldonado

Febrero 2003

Instituto Tecnológico y de Estudios Superiores de Monterrey

División de Graduados en Electrónica, Computación, Información y Comunicaciones

Dirección de Programas de Posgrado en Electrónica, Computación, Información y Comunicaciones

Los miembros del comité de tesis recomendamos que la presente tesis de Germán Montalvo Maldonado sea aceptada como requisito parcial para obtener el grado académico de Maestro en Ciencias en:

Sistemas Inteligentes

Comité de tesis:

Dr. Rogelio Soto Rodríguez

Asesor de la tesis

Dr. Hugo Terashima Marín

Sinodal

Dr. Manuel Valenzuela Rendón

Sinodal

David Garza Salazar, PhD.

Director del Programa de Graduados
en Electrónica, Computación,
Información y Comunicaciones

Febrero de 2003

A mis padres, por su apoyo incondicional.
A mi hermana, por su gran cariño y comprensión.
A mi sobrinita Emmy, por el simple hecho de existir.
A Dios, por poner siempre gente tan buena a mi alrededor.

Reconocimientos

Agradezco a mi asesor, el Dr. Soto, por su guía constante en la realización de esta tesis.

Agradezco a mis sinodales, el Dr. Terashima y el Dr. Valenzuela, por sus críticas constructivas y su disposición.

Agradezco infinitamente al coordinador de la maestría, el Dr. Terashima, por las facilidades otorgadas para mi ingreso a la maestría.

Igualmente, agradezco a mi equipo de trabajo en el Centro de Sistemas Inteligentes, el Dr. Galván, Víctor B., Roberto C. y Leo M., por todo lo aprendido con ustedes.

Un agradecimiento especial para Novelo, cuya ayuda fue primordial en el desarrollo de esta tesis. Muchas gracias por tu tiempo Novelo.

Agradezco también a mi eterno compañero de equipo en clases. Gracias Héctor por todo tu apoyo y amistad.

A Ale, Iván, Griss, Charly, Albores, von Borstel, Mandujano, Uc, Lupita, Roberto G., Luis F., Joaquín, Montelongo, Lucía, Max, Arján, Frydman, Brian, Michell, César, Meade, Martín, Juan del R., Luisandro, Daniel, Chema, Scott, Chacho, Omar, Oscar, Jorge, Hugo, Marc, Doris, Patty, Lucy, Karla, Lily, Yoly y Nadhiely; mil gracias por brindarme su amistad.

Quiero expresar mi admiración por la fortaleza que tuvieron tres amigos en la pérdida de un ser querido. A Ely, Juan Carlos y Mike, mis condolencias. Desde allá arriba, sus padres están muy orgullosos de ustedes.

GERMÁN MONTALVO MALDONADO

*Instituto Tecnológico y de Estudios Superiores de Monterrey
Febrero 2003*

Manejo de incertidumbre por medio de Sistemas de Lógica Difusa tipo 2 para la modelación de sistemas no lineales

Germán Montalvo Maldonado, M.C.
Instituto Tecnológico y de Estudios Superiores de Monterrey, 2003

Asesor de la tesis: Dr. Rogelio Soto Rodríguez

Esta tesis propone una metodología que permite manejar la incertidumbre en sistemas no lineales por medio de Lógica Difusa (LD) tipo 2 —la cual es una extensión de la LD convencional—. Bajo esta metodología se hicieron diversos experimentos de pronóstico de la serie de tiempo caótica *Mackey-Glass* cuando es corrompida por un ruido adicional.

La serie de tiempo en cuestión ha sido muy utilizada por diversas técnicas, especialmente por Redes Neuronales y Lógica Difusa (original), pero la gran mayoría hacían pronósticos donde no se tenía presente la incertidumbre en este sistema no lineal debida al ruido. Cabe mencionar que aquí sólo se hicieron experimentos bajo un marco de LD, tanto convencional como de tipo 2, por lo que sólo se muestra una comparación entre estas dos técnicas.

La tesis propone también una extensión al método de diseño directo convencional, pero utilizando ahora conjuntos difusos tipo 2 de intervalo en vez de conjuntos difusos convencionales. Asimismo, en este trabajo se hicieron otros pronósticos de la serie Mackey-Glass, utilizando los sistemas ya entrenados con la serie con ruido, sólo que en este caso el ruido estaba presente desde sus condiciones iniciales, y se verificó que los sistemas eran robustos a estos cambios.

Los resultados obtenidos pueden ser extendidos a problemas de otras áreas que presenten incertidumbre, tales como: diagnóstico médico, control de procesos industriales complejos, inversiones financieras, etc.

Índice General

Índice de Figuras	iii
Índice de Tablas	vii
Capítulo 1 Introducción	1
1.1 Definición del problema	2
1.2 Objetivos	4
1.3 Hipótesis	4
1.4 Aportaciones de la investigación	5
1.5 Estructura de la tesis	6
Capítulo 2 Sistemas de Lógica Difusa Tipo 1 (SLDT1)	7
2.1 SLDT1 <i>singleton</i>	10
2.1.1 Método de un paso	13
2.1.2 Método de gradiente descendente	14
2.2 SLDT1 no <i>singleton</i>	16
2.2.1 Método de un paso	20
2.2.2 Método de gradiente descendente	20
2.3 Resumen	22
Capítulo 3 Sistemas de Lógica Difusa Tipo 2 (SLDT2)	23
3.1 SLDT2 <i>singleton</i> de intervalo	26
3.1.1 Método de un paso	33
3.1.2 Método de gradiente descendente	33
3.2 SLDT2 no <i>singleton</i> de intervalo	34
3.2.1 Método de un paso	38
3.2.2 Método de gradiente descendente	38
3.3 Resumen	39
Capítulo 4 Experimentos y Resultados	41
4.1 Metodología de experimentación	41
4.2 SLDT1 <i>singleton</i>	43

4.2.1	Método de un paso	46
4.2.2	Método de gradiente descendente	48
4.2.3	Pronóstico con ruido	52
4.3	SLDT1 no <i>singleton</i>	55
4.3.1	Método de un paso	56
4.3.2	Método de gradiente descendente	57
4.4	SLDT2 <i>singleton</i>	61
4.4.1	Método de un paso	61
4.4.2	Método de gradiente descendente	62
4.5	SLDT2 no <i>singleton</i>	66
4.5.1	Método de un paso	66
4.5.2	Método de gradiente descendente	67
4.5.3	Otros diseños de entrenamiento	70
4.6	Pronóstico de serie alterna	74
Capítulo 5 Conclusiones y Trabajo Futuro		77
Apéndice A Pronóstico de Series de Tiempo		81
Apéndice B Conceptos y Operaciones de Conjuntos Difusos Tipo 2		83
B.1	Definiciones y conceptos básicos	83
B.1.1	El concepto de conjunto difuso tipo 2	83
B.1.2	Definición de conjunto difuso tipo 2	84
B.1.3	Función de membresía secundaria	85
B.1.4	Huella de incertidumbre (FOU)	87
B.1.5	Conjunto difuso tipo 2 de intervalo	88
B.1.6	Conjuntos incluidos	89
B.2	Operaciones en conjuntos difusos tipo 2 generales	91
B.2.1	Unión de conjuntos difusos tipo 2	91
B.2.2	Intersección de conjuntos difusos tipo 2	92
B.2.3	Complemento de un conjunto difuso tipo 2	93
B.3	Operaciones en conjuntos difusos tipo 2 de intervalo	94
B.3.1	Operación <i>join</i> en conjuntos de intervalo	94
B.3.2	Operación <i>meet</i> en conjuntos de intervalo	95
B.3.3	Complemento de un conjunto de intervalo	95
Bibliografía		97

Índice de Figuras

1.1	Muestras representativas de la serie de tiempo Mackey-Glass para diferentes retrasos. (a) $\tau = 13$. (b) $\tau = 30$	3
2.1	Estructura de un SLDT1.	8
2.2	Mecanismos de razonamiento difuso comúnmente usados y sus reglas difusas SI-ENTONCES.	10
2.3	Mecanismo de inferencia de un SLD <i>singleton</i> con composición máx-mín.	11
2.4	Mecanismo de inferencia de un SLD <i>singleton</i> con composición máx-prod.	12
2.5	Mecanismo de inferencia de un SLD no <i>singleton</i> con composición máx-mín.	17
3.1	Estructura de un SLDT2. Para enfatizar la importancia del conjunto de tipo reducido, se muestran dos salidas del SLDT2, el conjunto de tipo reducido y el valor nítido defuzificado.	24
3.2	Descripción ilustrada de las operaciones entre las entradas y los antecedentes de un SLDT2 <i>singleton</i> de intervalo (a) usando norma T mínimo, (b) usando norma T producto.	27
3.3	Descripción ilustrada de operaciones de consecuentes en un SLDT2 <i>singleton</i> de intervalo. (a) Conjuntos de salida disparados usando la norma T mínimo. (b) Conjuntos de salida disparados usando la norma T producto.	28
3.4	Descripción ilustrada del: (a) conjunto de salida compuesto por los dos conjuntos de salida disparados que se muestran en la figura 3.3 (a), (b) conjunto de salida compuesto por los dos conjuntos de salida disparados que se muestran en la figura 3.3 (b).	29
3.5	Descripción ilustrada de las operaciones entre las entradas y los antecedentes de un SLDT2 no <i>singleton</i> de intervalo (a) usando norma T mínimo, (b) usando norma T producto.	37
4.1	Diagrama de bloques de la metodología para el diseño de los SLD.	42
4.2	Simulación de la serie Mackey-Glass y su división en conjuntos de entrenamiento y de prueba.	44
4.3	Conjuntos difusos obtenidos de $\mathbf{x}^{(1)}$ para la regla $R^{(1)}$. (a) Antecedentes de $R^{(1)}$. (b) Consecuente de $R^{(1)}$	47

4.4	Gráfica comparativa de la salida del SLDT1 <i>singleton</i> con el método de un paso vs. la serie original.	48
4.5	Conjuntos difusos de la regla $\mathbf{R}_{\text{inicial}}^1$ con desviación estándar de 0.5861. (a) Antecedentes. (b) Consecuente.	51
4.6	Conjuntos difusos de la regla $\mathbf{R}_{\text{final}}^1$ con sus desviaciones estándar finales correspondientes. (a) Antecedentes. (b) Consecuente (conserva su σ_{inicial}).	51
4.7	Gráfica del error por época del SLD <i>singleton</i> diseñado con el método de gradiente descendente.	52
4.8	Una simulación de la serie de tiempo Mackey-Glass con datos de una señal de ruido adicional, $x(1001), x(1002), \dots, x(2000)$	53
4.9	Salidas superpuestas de 50 simulaciones del SLDT1 <i>singleton</i> con el método de un paso, cuando se tiene un ruido adicional en la serie.	54
4.10	Salidas superpuestas de 50 simulaciones del SLDT1 <i>singleton</i> con el método de un paso, cuando se tiene un ruido adicional en la serie.	54
4.11	Gráfica comparativa de los errores de los SLD, obtenidos en cada simulación.	55
4.12	Salidas de los SLDT1 usando el método de diseño de un paso y con fuzificación: (a) <i>singleton</i> , (b) no <i>singleton</i> . En ambos casos hay 50 simulaciones y los datos del 1504 al 2000 están corrompidos por un ruido adicional uniformemente distribuido con $SNR = 0\text{dB}$	58
4.13	Promedios y desviaciones estándar de $RMSE_s(GD)$ y $RMSE_{ns}(GD)$ obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación. (a) Promedios. (b) Desviaciones estándar.	60
4.14	Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$ y $RMSE_{s2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación. (a) Promedios. (b) Desviaciones estándar.	65
4.15	Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$, $RMSE_{s2}(GD)$ y $RMSE_{ns2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación utilizando un diseño de entrenamiento totalmente independiente. (a) Promedios. (b) Desviaciones estándar.	69
4.16	Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$, $RMSE_{s2}(GD)$ y $RMSE_{ns2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación utilizando un diseño de entrenamiento parcialmente independiente. (a) Promedios. (b) Desviaciones estándar.	71
4.17	Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$, $RMSE_{s2}(GD)$ y $RMSE_{ns2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación utilizando un diseño de entrenamiento dependiente. (a) Promedios. (b) Desviaciones estándar.	73

4.18	Serie de tiempo Mackey-Glass con condiciones iniciales con ruido. (a) Datos sin ruido adicional, $s_2(1001), s_2(1002), \dots, s_2(2000)$. (b) Datos con ruido adicional, $x_2(1001), x_2(1002), \dots, x_2(2000)$	75
B.1	(a) Función de membresía tipo 1. (b) Función de membresía tipo 1 con trazo difuso, incluyendo discretización en $x = x'$	84
B.2	Ejemplo de un conjunto difuso tipo 2.	85
B.3	Ejemplo de un corte vertical para la función de membresía tipo 2 mostrada en la figura B.2.	86
B.4	Las áreas sombreadas representan las FOUs, las líneas sólidas denotan las FMs superiores y las punteadas denotan las inferiores. (a) FM Gaussiana con media incierta. (b) FM Gaussiana con desviación estándar incierta. . .	87
B.5	La región sombreada en (a) es la FOU para un conjunto difuso tipo 2. (a) Las membresías primarias, J_{x_1} y J_{x_2} , y sus funciones de membresía secundarias asociadas $\mu_{\tilde{A}}(x_1)$ y $\mu_{\tilde{A}}(x_2)$ se muestran en los puntos x_1 y x_2 . (b) Las funciones de membresía secundarias son conjuntos de intervalo. . .	89
B.6	Ejemplo de un conjunto incluido tipo 2 asociado con la función de membresía de la figura B.2.	90
B.7	(a) Conjuntos difusos tipo 2 de intervalo \tilde{F} y \tilde{G} . (b) Conjuntos tipo 1 de intervalo F y G cuando $x = 0.75$. (c) Unión entre \tilde{F} y \tilde{G} . (d) Intersección entre \tilde{F} y \tilde{G} . (e) Complemento de \tilde{F} . (f) Operación <i>negation</i> de F	96

Índice de Tablas

4.1	Valores iniciales para el centro de las funciones de membresía Gaussianas de los antecedentes y el centro del conjunto consecuente. Note que $m_s - 2\sigma_s = 0.3086$ y $m_s + 2\sigma_s = 1.4809$	49
4.2	Valores finales para el centro de las funciones de membresía Gaussianas de los antecedentes y el centro del conjunto consecuente, después de 6 épocas de ajuste.	50
4.3	Valores finales para la desviación estándar de las funciones de membresía Gaussianas de los antecedentes, después de 6 épocas de ajuste.	50
4.4	Promedios y desviaciones estándar de $RMSE_{s1}(UP)$, $RMSE_{ns1}(UP)$ y $RMSE_{s2}(UP)$, obtenidos en 50 simulaciones de diseño.	62
4.5	Promedios y desviaciones estándar de $RMSE_{s1}(UP)$, $RMSE_{ns1}(UP)$, $RMSE_{s2}(UP)$ y $RMSE_{ns2}(UP)$ obtenidos en 50 simulaciones de diseño.	67
4.6	Promedios y desviaciones estándar de los $RMSEs$ obtenidos al hacer 50 simulaciones de pronóstico de la serie original y de la serie alterna, ambas con ruido adicional.	74
4.7	Resumen de experimentos realizados.	76
5.1	Valores iniciales de los parámetros en cuatro SLDs diseñados con el método de gradiente descendente. Cada antecedente está descrito por dos conjuntos difusos ($i = 1, \dots, M$ y $k = 1, \dots, p$).	78

Capítulo 1

Introducción

El ser humano desde siempre ha desarrollado estrategias para hacer frente a diferentes tipos de incertidumbre que se le presentan, y las ha aplicado aun cuando no ha estado absolutamente seguro de sus consecuencias, con la esperanza de que los efectos dañinos se vean atenuados de manera considerable.

Anteriormente la modelación de incertidumbre en sistemas no lineales se ha manejado con distintas técnicas y métodos, tales como las técnicas Box-Jenkins —también conocidas como modelos ARIMA, por sus siglas en inglés “*AutoRegressive Integrated Moving Average*” [1]—, estadísticas bayesianas [24], métodos difusos [4], redes neuronales [28], sistemas difusos combinados con redes neuronales o algoritmos genéticos [29], etcétera. Particularmente, los campos de redes neuronales y lógica difusa han tratado un problema que ha servido como punto de referencia en la comparación del desempeño de sus sistemas, que es el pronóstico de la serie de tiempo *Mackey-Glass*. Esta serie se caracteriza por ser muy difícil de pronosticar debido a que no tiene períodos, es no lineal, tiene comportamiento caótico (i.e., tiene grandes variaciones en su salida cuando sus condiciones iniciales varían muy poco) y sus observaciones parecen no presentar correlación, por lo que su modelación y pronóstico representan un reto para cualquier técnica.

En este trabajo se hace el pronóstico de dicha serie cuando es corrompida por un ruido adicional incierto, mediante una técnica que permite modelar problemas con incertidumbre, en un marco de Lógica Difusa (LD). Esta técnica es la LD Tipo 2, que es una extensión de la LD original —de ahora en adelante llamada LD Tipo 1—, la cual fue fundada por Lofti Zadeh hace más de 35 años [34]. Dicha extensión surge debido a la necesidad de modelar apropiadamente la incertidumbre, y así, reducir considerablemente sus efectos. Cabe mencionar que si desaparece el término incierto en el problema, la LD Tipo 2 se reduce a LD Tipo 1, de la misma manera que si desaparece la aleatoriedad, la probabilidad se reduce a determinismo.

Los resultados obtenidos podrán extenderse al pronóstico de otras series de tiem-

po que tengan ruido, así como a otros problemas donde se presente la incertidumbre, como es el caso del diagnóstico de fallas en plantas, redes de comunicación, reconocimiento de patrones, control robusto, entre otros.

1.1 Definición del problema

Hoy por hoy, el caos está teniendo un gran impacto en campos muy diferentes, incluyendo la física, la biología, la química, y la medicina [2], [5] y [25]. El comportamiento caótico se puede definir brevemente como fluctuaciones definidas en la salida de un sistema *no lineal* con alto grado de sensibilidad a *condiciones iniciales* [2], i.e., trayectorias con condiciones iniciales casi idénticas pueden diferir mucho entre sí. Un sistema exhibiendo dinámicas caóticas evoluciona de manera *determinística*; y existe un enorme interés en este tipo de sistemas debido a que presentan dos características contrastantes; ya que a pesar de que son impredecibles a largo plazo; sí se pueden realizar predicciones a corto plazo. Lo que significa que series de tiempo que parezcan aleatorias a primera vista, pueden ser de hecho predecibles en periodos de corto tiempo [12]. Por otro lado, la correlación de observaciones de estos sistemas parece ser limitada, y por lo tanto sus observaciones parecen no estar correlacionadas; lo que hace que su predicción sea particularmente difícil [25].

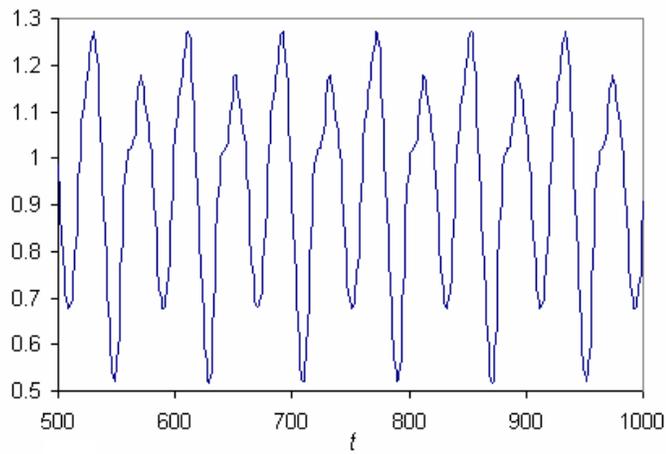
El sistema con dinámicas caóticas que se presenta en esta tesis es una serie de tiempo específica, sobre la cual se desea hacer pronóstico de primera etapa cuando es corrompida por un ruido adicional. Esta serie se obtiene de la ecuación Mackey-Glass, como se describe a continuación.

En 1977, Mackey y Glass [16] publicaron un importante artículo en el que presentaban “la asociación entre el inicio de una enfermedad y las bifurcaciones en las dinámicas de ecuaciones de retraso diferencial de primer orden que modelan sistemas fisiológicos”. La ecuación (4b) de ese artículo se conoce ahora como la *ecuación Mackey-Glass*, que es una ecuación de retraso diferencial no lineal de la forma

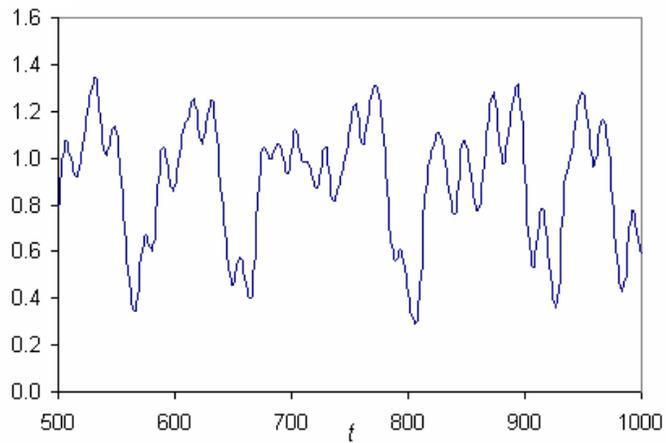
$$\frac{ds(t)}{dt} = \frac{0.2s(t - \tau)}{1 + s^{10}(t - \tau)} - 0.1s(t) \quad (1.1)$$

que exhibe caos para $\tau > 17$.

Para demostrar la naturaleza cualitativa de esta ecuación, en la figura 1.1 (a) y (b) se muestran porciones representativas de la serie de tiempo Mackey-Glass asociada [i.e., la solución de (1.1)] para cada uno de los dos valores de τ usados. En estas gráficas se puede distinguir el comportamiento periódico para valores pequeños de τ y el comportamiento caótico para valores grandes de τ .



(a)



(b)

Figura 1.1: Muestras representativas de la serie de tiempo Mackey-Glass para diferentes retrasos. (a) $\tau = 13$. (b) $\tau = 30$.

Como se puede observar en la figura 1.1 (b), se tiene un comportamiento caótico a pesar de que todavía no se tiene incertidumbre en el sistema; esta incertidumbre se presenta hasta que se adiciona un cierto ruido a la serie, lo que hace aún más difícil realizar el pronóstico. Para más detalles sobre pronósticos de series de tiempo, véase el Apéndice A.

Es importante resaltar que un objetivo de suma importancia al hacer pronósticos, es servir como ayuda para la toma de decisiones; ya que dependiendo de las condiciones que se predigan, serán las acciones que se tomarán para obtener los resultados deseados. Sin embargo, si estas predicciones están mal elaboradas, pueden tomarse acciones que llevarán a resultados no óptimos, o peor aún, a resultados que repre-

senten pérdidas económicas considerables. Por lo tanto, es necesario establecer una metodología que pueda sobrellevar esta característica incierta, para lo cual se aplicará LD Tipo 2, y se observará el desempeño del Sistema de Lógica Difusa (SLD) resultante en la solución del problema.

1.2 Objetivos

El objetivo general es manejar la incertidumbre en sistemas no lineales utilizando Sistemas de Lógica Difusa Tipo 2 (SLDT2) bajo diferentes esquemas de diseño. Para este propósito, se trabajó específicamente con la serie de tiempo caótica Mackey-Glass, que es un sistema no lineal que presenta incertidumbre en su comportamiento cuando sus datos contienen un ruido adicional. Se hizo también un desarrollo aplicando LD convencional —Tipo 1—, ya que el alcance de este trabajo abarca un análisis comparativo del desempeño de dichos SLDT2s contra el desempeño de sistemas de LD convencional, tomándose como punto de referencia el problema anteriormente mencionado.

Para lograr este objetivo general, es necesario establecer los siguientes objetivos específicos:

- Establecer una metodología que permita el diseño de sistemas en los que se tenga presente una característica incierta (e.g., ruido).
- Analizar distintos enfoques de entrenamiento de las reglas en los sistemas de LD.
- Hacer el diseño de los SLDs, tanto de tipo 1 (SLDT1) como de tipo 2 (SLDT2), basándose en los enfoques analizados.
- Realizar los pronósticos con los sistemas diseñados.
- Analizar el desempeño de cada sistema, de manera que los resultados puedan ser comparados correctamente con su contraparte correspondiente.

1.3 Hipótesis

La hipótesis principal aquí planteada es que un SLDT2 tiene un mejor desempeño que un SLDT1 cuando el problema incluye incertidumbre en el entrenamiento de sus reglas, debido a que es difícil para el SLDT1 absorber el ruido de los datos de entrenamiento con las funciones de membresía de sus antecedentes. Sin embargo, debido a que la naturaleza de las funciones de membresía de los conjuntos difusos

tipo 2 es en sí misma difusa, se espera que los SLDT2 tengan menor error al realizar el pronóstico.

Con este trabajo se busca responder las siguientes preguntas:

- ¿Cómo es el desempeño de los SLDT1s cuando no se maneja incertidumbre en el sistema?
- ¿Cómo es el desempeño de los SLDT1s cuando se introduce incertidumbre al hacer pronóstico?
- ¿El ruido en el entrenamiento afecta el desempeño de los SLDT1?
- ¿De qué manera la lógica difusa tipo 2 maneja la incertidumbre?
- ¿Se puede hacer un buen pronóstico de la serie con la LD tipo 2, aun con ruido en los datos de entrenamiento?
- ¿Tiene realmente la LD Tipo 2 un mejor desempeño que la de Tipo 1 en el manejo de incertidumbre en el problema de aplicación?

1.4 Aportaciones de la investigación

En esta tesis se propone una metodología bajo la cual se pueden diseñar SLDs, tanto de tipo 1 como de tipo 2, bajo un esquema de entrenamiento a partir de datos numéricos. La metodología es lo bastante flexible como para poder aplicar diferentes métodos de diseño, así como distintos diseños de entrenamiento, como se muestra en el capítulo 4.

Otra aportación de esta tesis es el método de diseño de un paso en los SLDT2s, con fuzificadores singleton y no singleton, el cuál es una extensión del utilizado para los sistemas de tipo 1. El método propuesto utiliza funciones de membresía tipo 2 (en lugar de tipo 1) en los conjuntos difusos de las reglas del sistema.

Como parte de los experimentos realizados con el método de diseño de gradiente descendente, se presentan los resultados obtenidos con tres diseños de entrenamiento: el dependiente, el parcialmente independiente y el totalmente independiente. Estos diseños de entrenamiento están inspirados en los dos diseños que sugiere Mendel en [17], sólo que ahí se muestran únicamente experimentos utilizando el diseño de entrenamiento dependiente.

1.5 Estructura de la tesis

El presente documento está compuesto por cinco capítulos. El capítulo 2 presenta la estructura básica de los SLDT1, los tipos de sistemas más usados y los métodos de diseño utilizados en este trabajo. El capítulo 3 expone un marco teórico para los SLDT2s, el cual incluye la estructura de estos sistemas y su analogía con la de los SLDT1s, e incluye también sus métodos de diseño. Los experimentos realizados, así como los resultados obtenidos, se presentan en el capítulo 4. Las conclusiones y el trabajo futuro se encuentran en el capítulo 5. En el apéndice A se da una explicación general de lo que involucra el hacer un pronóstico de series de tiempo. Finalmente, el apéndice B explica los conceptos más utilizados con conjuntos difusos tipo 2 y sus operaciones básicas.

Capítulo 2

Sistemas de Lógica Difusa Tipo 1 (SLDT1)

En este capítulo se da una breve introducción a los SLDs en general, se analiza su estructura básica y se presentan los tipos de sistemas más usados en aplicaciones ingenieriles, clasificados de acuerdo a su mecanismo de razonamiento y al tipo de reglas difusas utilizadas. Asimismo, se exponen los tipos de fuzificadores que pueden usarse junto con dos métodos de diseño para construir estos SLDT1s.

Los sistemas de lógica difusa —comúnmente llamados sistemas difusos—, son *sistemas basados en el conocimiento* o *sistemas basados en reglas*, y están compuestos por los siguientes elementos: reglas, fuzificador, máquina de inferencia y procesador de salida, los cuales están conectados como se muestra en la figura 2.1 [17]. Una vez que se han establecido las reglas, las cuales constituyen lo que se conoce como *base de reglas difusas* [31], un SLD puede ser visto como un mapeador de entradas-salida (la ruta marcada por la línea gruesa en la figura 2.1, desde “Entradas Nítidas” hasta “Salida Nítida”). Este mapeo puede ser expresado cuantitativamente como $y = f(\mathbf{x})$. Note que el SLD mostrado, al igual que todos los que aquí se manejan, es de múltiples entradas y una salida. Este tipo de SLD es ampliamente usado en muchas aplicaciones ingenieriles de LD, tales como controladores difusos y procesadores de señal difusos.

Las reglas son el corazón de un SLD, y pueden ser provistas por expertos o pueden ser extraídas de datos numéricos. En cualquier caso, pueden ser expresadas como una colección de oraciones SI-ENTONCES, e.g.,

SI la velocidad del carro es alta, ENTONCES aplicar menos fuerza al acelerador

La parte SI de la regla es su *antecedente*, y la parte ENTONCES de la regla es su *consecuente*.

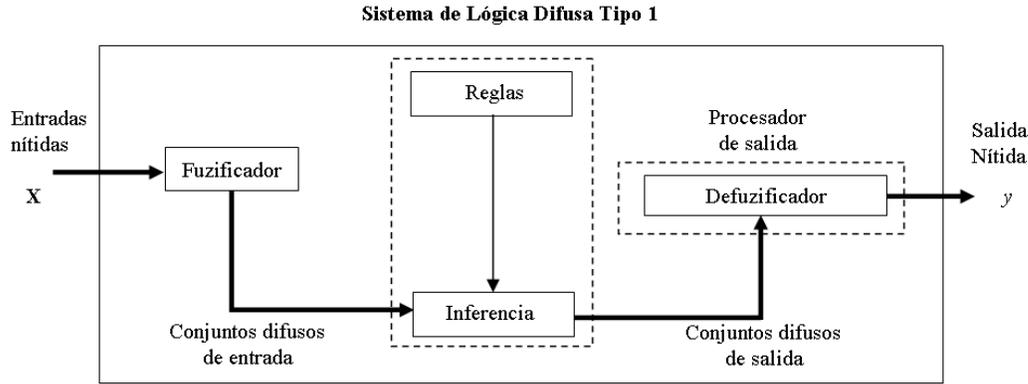


Figura 2.1: Estructura de un SLDT1.

Los conjuntos difusos están asociados a términos que aparecen en los antecedentes o consecuentes de reglas. Para describir estos conjuntos difusos se usan las llamadas *funciones de membresía*, $\mu_F(x)$. Estas funciones de membresía pueden tener diferentes formas, las más usadas son las triangulares, trapezoidales, lineales por segmentos, Gaussianas y de campana.

En un SLD, el *fuzificador* es el que se encarga de mapear los números nítidos de la entrada en conjuntos difusos; lo que permite activar reglas que están en términos de variables lingüísticas, las cuales tienen asociados conjuntos difusos. Las entradas del SLD antes de la fuzificación pueden ser ciertas (e.g., mediciones perfectas) o inciertas (e.g., mediciones ruidosas). Las funciones de membresía para conjuntos difusos tipo 1 manejan apropiadamente las primeras, mientras que los de tipo 2 permiten manejar cualquiera de los dos tipos de mediciones.

La *máquina de inferencia* de la figura 2.1 mapea los conjuntos difusos obtenidos por el fuzificador en otros conjuntos difusos, y es la que administra la manera en la que las reglas son activadas y combinadas. De la misma forma en la que los humanos usan diferentes tipos de procedimientos de inferencia para entender algo o hacer decisiones, existen diferentes procedimientos de inferencia en LD. Los más comunes son el *mínimo* y el *producto*.

En muchas aplicaciones, los conjuntos difusos de salida de la máquina de inferencia de un SLD se deben transformar en un número nítido, mismo que será la salida general del sistema. Este procesamiento de salida es llamado *defuzificación*. El procesador de salida en un sistema difuso tipo 1, está formado únicamente por un *defuzificador*; sin embargo, en un SLDT2, el procesador de salida tiene dos com-

ponentes: el primero mapea un conjunto de tipo 2 en un conjunto de tipo 1, y el segundo se encarga de la defuzificación de este conjunto.

Resumiendo, los pasos del *razonamiento difuso* [7] que lleva a cabo un SLD son:

- Comparar las variables de entrada con las funciones de membresía en la parte de los antecedentes para obtener los valores de membresía de cada variable lingüística. Este paso corresponde a la fuzificación.
- Combinar (mediante un operador de norma T, comúnmente producto o mínimo) los valores de membresía en la parte de los antecedentes para obtener un *nivel de disparo* (grado de activación) de cada regla.
- Generar el consecuente ponderado de cada regla dependiendo de su nivel de disparo. Este paso y el anterior los realiza la máquina de inferencia internamente.
- Agregar los consecuentes ponderados para producir una salida nítida. Este paso corresponde a la defuzificación.

Dependiendo del tipo de razonamiento difuso y de las reglas difusas SI-ENTONCES usadas, la mayoría de los sistemas difusos pueden clasificarse [17] en SLD Mamdani y SLD Takagi-Sugeno-Kang (TSK), que son los más usados en aplicaciones ingenieriles. En la figura 2.2 se muestran sus mecanismos de razonamiento difuso.

SLD Mamdani: En este sistema la salida difusa global se obtiene al aplicar una operación “máx” a las salidas difusas ponderadas (cada una de las cuales es igual al mínimo del nivel de disparo y la función de membresía de salida de cada regla). Existen varios esquemas para obtener la salida nítida final a partir de la salida difusa global, como son el centroide, centro promedio (defuzificador de altura), criterio de máximo, entre otros.

SLD TSK: En este sistema la salida de cada regla es una combinación lineal de variables de entrada más un término constante, y la salida general es el promedio ponderado de la salida de cada regla.

En conclusión, se puede decir que ambos están caracterizados por reglas SI-ENTONCES y tienen las mismas estructuras en sus antecedentes; sólo difieren en las estructuras de sus consecuentes. El consecuente de una regla Mamdani es un conjunto difuso, mientras que el consecuente de una regla TSK es una función.

En este trabajo sólo se utilizan SLDs Mamdani, debido a la sencillez que presentan para establecer un conjunto difuso en el consecuente, siendo necesario definir

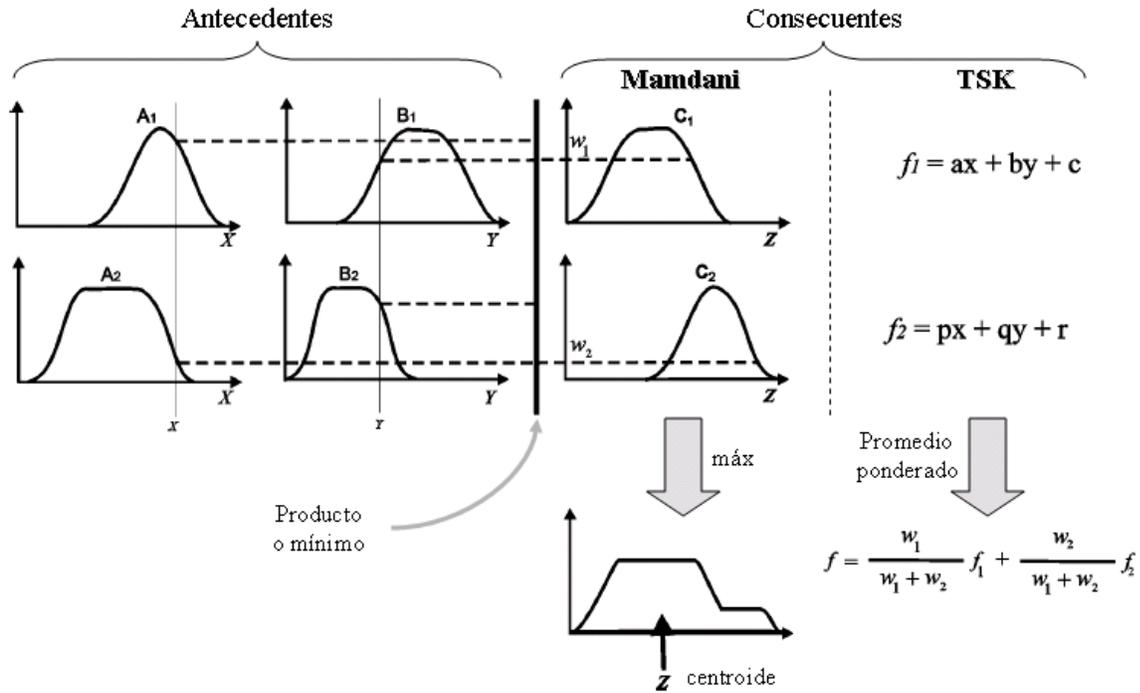


Figura 2.2: Mecanismos de razonamiento difuso comúnmente usados y sus reglas difusas SI-ENTONCES.

únicamente los parámetros de su función de membresía (e.g., la media y la desviación estándar en una Gaussiana). Además, estos sistemas facilitan el proceso de aprendizaje de las reglas cuando se usa el método directo de entrenamiento de un solo paso, ver sección 2.1.1. Por esta razón, de ahora en adelante cuando se mencione un SLD, éste se referirá a un SLD Mamdani.

Los SLDs, dependiendo del tipo de fuzificación [17], se pueden dividir en *singleton* y no *singleton*, los cuales se explican en las siguientes secciones. Recuerde que en este capítulo sólo se exponen SLDs tipo 1, los de tipo 2 se explican en el capítulo 3.

2.1 SLDT1 *singleton*

En un SLD *singleton*, el fuzificador mapea un punto nítido en un *singleton* difuso, el cual es un conjunto que tiene sólo un elemento, la unidad, i.e.,

$$\mu_{A'}(x) = \begin{cases} 1 & \text{si } x = x' \\ 0 & \text{en caso contrario} \end{cases}$$

Por lo tanto, un SLD *singleton* puede hacer un mapeo directo de las entradas nítidas

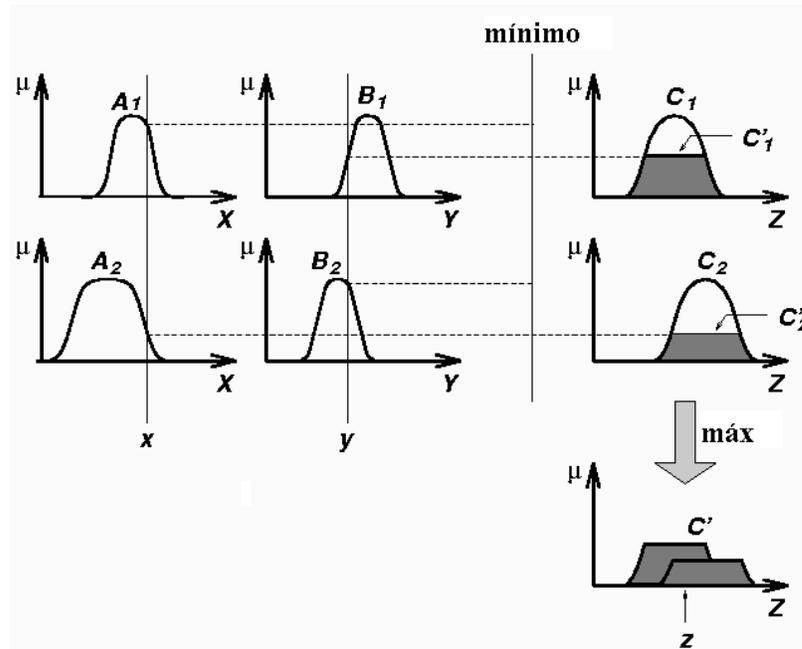


Figura 2.3: Mecanismo de inferencia de un SLD *singleton* con composición máx-mín.

(i.e., los *singleton* difusos) al valor de membresía de los conjuntos difusos de los antecedentes. Con estos valores de membresía, se calcula el nivel de disparo de cada regla usando una norma T (mínimo o producto). Luego, se aplica una composición (máx-mín o máx-prod) con este nivel de disparo al conjunto difuso del consecuente respectivo. Posteriormente, se combinan estos consecuentes ponderados mediante la norma S (máximo) para obtener un conjunto difuso de salida; al cual se le aplica un proceso de defuzificación para calcular el valor nítido, que será la salida general del sistema.

La forma de la salida inferida en cada regla depende de la composición que utilice el sistema [9]; por ejemplo, cuando se usa la composición máx-mín, la salida es un conjunto difuso recortado por el nivel de disparo de la regla respectiva, ver figura 2.3. Cuando se usa la composición máx-prod, la salida es un conjunto difuso escalado por el nivel de disparo de su regla correspondiente, ver figura 2.4.

Los SLDs *singleton* son los más utilizados, y existen muchos métodos de diseño para su construcción. Más adelante se describen brevemente dos de estos métodos de diseño que son útiles para tratar el problema de pronóstico, el cual puede plantearse de la siguiente manera:

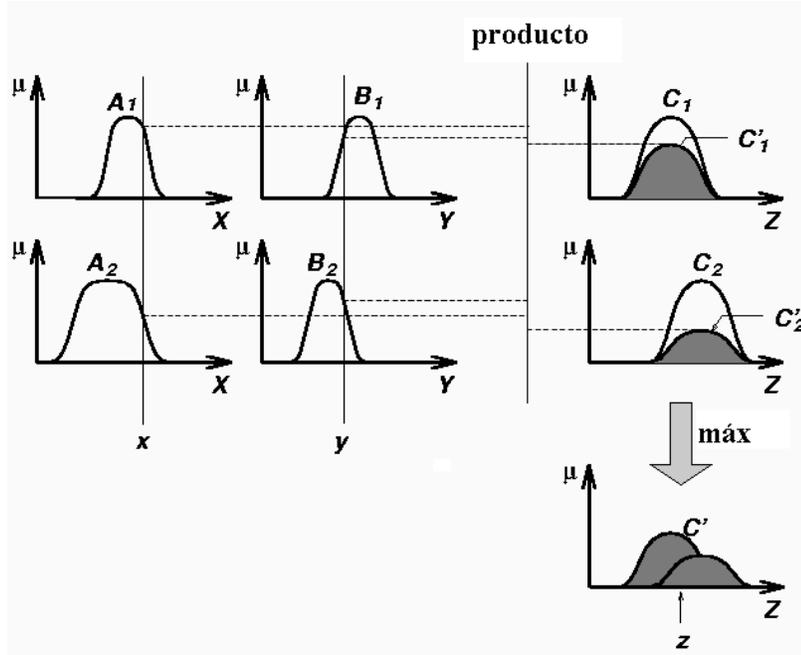


Figura 2.4: Mecanismo de inferencia de un SLD *singleton* con composición máx-prod.

Dada una colección de N pares de datos de entrada–salida de entrenamiento, $(\mathbf{x}^{(1)} : y^{(1)}), (\mathbf{x}^{(2)} : y^{(2)}), \dots, (\mathbf{x}^{(N)} : y^{(N)})$, donde \mathbf{x} es el vector de entrada y y es la salida escalar de un SLD *singleton*, la meta es especificar completamente el SLD usando los datos de entrenamiento.

Esta especificación implica hacer una selección en cada uno de los elementos que aparecen en la figura 2.1; por ejemplo, si se decide utilizar un fuzificador *singleton*, una composición máx-prod, una implicación producto y un defuzificador de centro promedio (dejando abierta la selección de la función de membresía), se puede especificar la salida del SLD en función de su entrada \mathbf{x} con la siguiente expresión:

$$y(\mathbf{x}) = f_s(\mathbf{x}) = \frac{\sum_{l=1}^M \bar{y}^l \prod_{i=1}^p \mu_{F_i^l}(x_i)}{\sum_{l=1}^M \prod_{i=1}^p \mu_{F_i^l}(x_i)} \quad (2.1)$$

donde p es el número de entradas, M es el número de reglas, l es el índice de la regla, \bar{y}^l es el centro del conjunto difuso del consecuente y $\mu_{F_i^l}(x_i)$ son las funciones de membresía de los conjuntos difusos de los antecedentes. Note el uso del subíndice “s” en $f_s(x)$ para indicar que es un SLD *singleton*.

Los datos de entrenamiento pueden ser interpretados como una colección de reglas SI-ENTONCES. Cada regla tiene la forma

$$R^l: \text{SI } x_1 \text{ es } F_1^l \text{ y } x_2 \text{ es } F_2^l \text{ y } \cdots \text{ y } x_p \text{ es } F_p^l, \text{ ENTONCES } y \text{ es } G^l \quad l = 1, 2, \dots, M \quad (2.2)$$

donde F_i^l son los conjuntos difusos de las variables de entrada y G^l los conjuntos difusos de la variable de salida, todos descritos por las funciones de membresía que se seleccionen. Cada método de diseño establece cómo especificar la media $m_{F_i^l}$ y la desviación estándar $\sigma_{F_i^l}$ de las funciones de membresía de los antecedentes, así como los centros de las funciones de membresía de los consecuentes, la \bar{y}^l en (2.1), usando los pares de entrenamiento $(\mathbf{x}^{(1)} : y^{(1)}), (\mathbf{x}^{(2)} : y^{(2)}), \dots, (\mathbf{x}^{(N)} : y^{(N)})$.

El número de reglas que puede haber en el SLD depende del método de diseño que se use para construirlo. Si no se hace ajuste en los parámetros del SLD, entonces pueden haber tantas reglas como pares de datos de entrada–salida, i.e., a lo más pueden haber $M = N$ reglas. Sin embargo, si se hace ajuste, tomando en cuenta que comúnmente hay menos parámetros de diseño que pares de entrenamiento, entonces se usan menos de N reglas. En las secciones siguientes se explican dos métodos de diseño, cuya diferencia principal se basa precisamente en que uno hace ajuste en los parámetros del sistema y el otro no.

2.1.1 Método de un paso

En este método [17] no se hace ajuste en los parámetros del SLD, ya que una vez que las reglas son obtenidas de los datos del conjunto de entrenamiento, éstas ya no son objeto de cambio alguno. De hecho, los datos establecen directamente los centros de los conjuntos difusos de los antecedentes y consecuentes de las reglas. Por lo tanto, la cantidad de reglas depende del número de entradas p del sistema y del número de datos de entrenamiento D ; así, hay $D - p$ reglas que se pueden extraer de los $D - p$ pares de entrenamiento (apéndice A), $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(D-p)}$:

$$R^1: \text{SI } x_1 \text{ es } F_1^1 \text{ y } x_2 \text{ es } F_2^1 \text{ y } \cdots \text{ y } x_p \text{ es } F_p^1, \text{ ENTONCES } y \text{ es } G^1$$

En esta regla, la cual es obtenida de $\mathbf{x}^{(1)}$, F_1^1 es un conjunto difuso cuya función de membresía está centrada en $x(1)$, F_2^1 es un conjunto difuso cuya función de membresía está centrada en $x(2)$, \dots , F_p^1 es un conjunto difuso cuya función de membresía está centrada en $x(p)$, y G^1 es un conjunto difuso cuya función de membresía está centrada en $x(p + 1)$.

$$R^2: \text{SI } x_1 \text{ es } F_1^2 \text{ y } x_2 \text{ es } F_2^2 \text{ y } \cdots \text{ y } x_p \text{ es } F_p^2, \text{ ENTONCES } y \text{ es } G^2$$

En esta regla, la cual es obtenida de $\mathbf{x}^{(2)}$, F_1^2 es un conjunto difuso cuya función de membresía está centrada en $x(2)$, F_2^2 es un conjunto difuso

cuya función de membresía está centrada en $x(3), \dots, F_p^2$ es un conjunto difuso cuya función de membresía está centrada en $x(p+1)$, y G^2 es un conjunto difuso cuya función de membresía está centrada en $x(p+2)$. Y así sucesivamente, hasta llegar a la regla siguiente:

R^{D-p} : SI x_1 es F_1^{D-p} y x_2 es F_2^{D-p} y \dots y x_p es F_p^{D-p} , ENTONCES y es G^{D-p}

En esta regla, la cual es obtenida de $\mathbf{x}^{(D-p)}$, F_1^{D-p} es un conjunto difuso cuya función de membresía está centrada en $x(D-p+1)$, F_2^{D-p} es un conjunto difuso cuya función de membresía está centrada en $x(2), \dots, F_p^{D-p}$ es un conjunto difuso cuya función de membresía está centrada en $x(D-1)$, y G^{D-p} es un conjunto difuso cuya función de membresía está centrada en $x(D)$.

En este primer esquema para obtener reglas a partir de datos numericos, se puede ver que los centros de las funciones de membresía de los antecedentes y consecuentes están completamente determinadas por los datos que son usados para crear las reglas. Típicamente, el resto de los parámetros de las funciones de membresía (e.g., la desviación estándar en las Gaussianas) son preestablecidos por el diseñador.

2.1.2 Método de gradiente descendente

A diferencia del método de un paso de la sección anterior, en este método¹ sí se hace un ajuste en los parámetros del SLD, usando un algoritmo de gradiente descendente [17], [31] y los N pares de entrenamiento. El objetivo de este método es minimizar la siguiente función de error:

$$e^{(i)} = \frac{1}{2}[f_s(\mathbf{x}^{(i)}) - y^{(i)}]^2 \quad i = 1, \dots, N \quad (2.3)$$

donde $f_s(\mathbf{x}^{(i)})$ es el valor obtenido al entrenar el sistema definido en (2.1) con el par de entrada-salida $(\mathbf{x}^{(i)} : y^{(i)})$. Si para completar la definición de dicho sistema se eligen funciones de membresía Gaussianas de la forma

$$\mu_{F_k^l}(x_k) = \exp \left\{ -\frac{1}{2} \left(\frac{x_k^{(i)} - m_{F_k^l}}{\sigma_{F_k^l}} \right)^2 \right\} \quad k = 1, \dots, p \text{ (entradas)} \quad (2.4)$$

¹También se le conoce como método de Retropropagación

entonces, se puede reexpresar (2.1) como

$$y(\mathbf{x}^{(i)}) = f_s(\mathbf{x}^{(i)}) = \frac{\sum_{l=1}^M \bar{y}^l \prod_{k=1}^p \exp \left[-\frac{1}{2} \left(\frac{x_k^{(i)} - m_{F_k^l}}{\sigma_{F_k^l}} \right)^2 \right]}{\sum_{l=1}^M \prod_{k=1}^p \exp \left[-\frac{1}{2} \left(\frac{x_k^{(i)} - m_{F_k^l}}{\sigma_{F_k^l}} \right)^2 \right]} \quad i = 1, \dots, N \quad (2.5)$$

de donde se puede ver que f_s está caracterizado completamente por \bar{y}^l , $m_{F_k^l}$ y $\sigma_{F_k^l}$, los cuales son los parámetros que se van a ajustar para minimizar el error en (2.3) usando las fórmulas siguientes:

$$m_{F_k^l}(i+1) = m_{F_k^l}(i) - \alpha [f_s(\mathbf{x}^{(i)}) - y^{(i)}] [\bar{y}^l(i) - f_s(\mathbf{x}^{(i)})] \times \frac{[x_k^{(i)} - m_{F_k^l}(i)]}{\sigma_{F_k^l}^2(i)} \phi_l(\mathbf{x}^{(i)}) \quad (2.6)$$

$$\bar{y}^l(i+1) = \bar{y}^l(i) - \alpha [f_s(\mathbf{x}^{(i)}) - y^{(i)}] \phi_l(\mathbf{x}^{(i)}) \quad (2.7)$$

$$\sigma_{F_k^l}(i+1) = \sigma_{F_k^l}(i) - \alpha [f_s(\mathbf{x}^{(i)}) - y^{(i)}] [\bar{y}^l(i) - f_s(\mathbf{x}^{(i)})] \times \frac{[x_k^{(i)} - m_{F_k^l}(i)]^2}{\sigma_{F_k^l}^3(i)} \phi_l(\mathbf{x}^{(i)}) \quad (2.8)$$

donde $i = 0, 1, \dots, (N-1)$ (pares de entrada-salida), $k = 1, \dots, p$ (entradas), $l = 1, \dots, M$ (reglas), y

$$\phi_l(\mathbf{x}^{(i)}) = \frac{\prod_{k=1}^p \exp \left(-\frac{(x_k^{(i)} - m_{F_k^l}(i))^2}{2\sigma_{F_k^l}^2(i)} \right)}{\sum_{l=1}^M \prod_{k=1}^p \exp \left(-\frac{(x_k^{(i)} - m_{F_k^l}(i))^2}{2\sigma_{F_k^l}^2(i)} \right)} \quad (2.9)$$

Estos tres parámetros pueden ser inicializados aleatoriamente; sin embargo, se recomienda escoger valores iniciales o “semillas” de manera inteligente (i.e., usar cantidades con significado físico) para que el algoritmo converja más rápido.

Los parámetros se van ajustando para reducir el error a medida que se avanza en el número de época, donde una época se define como el ajuste realizado en un ciclo que recorre una sola vez el conjunto completo de los N pares de entrenamiento. El algoritmo se detiene cuando llega a un número preestablecido de épocas o cuando se ha alcanzado un error menor a un valor arbitrario de $e^{(i)}$.

2.2 SLDT1 no *singleton*

Un SLD (tipo 1) no *singleton* [17] es aquel cuyas entradas son modeladas como números difusos (i.e., conjuntos difusos); por lo tanto puede ser usado para manejar la incertidumbre que se presenta cuando se aplican entradas inciertas al sistema, por ejemplo cuando se tienen mediciones ruidosas. Esto no quiere decir que ésta sea la única manera de modelar dicha incertidumbre o que sea la más adecuada, si no que, aunque sea poca la mejoría, permite atenuar el ruido que presentan las entradas. Sin embargo, el desempeño del sistema puede mejorar aún más usando un SLD tipo 2, como se verá en el capítulo 5.

Un SLD tipo 1 no *singleton* tiene la misma estructura que un *singleton* (figura 2.1), y comparten el mismo tipo de reglas; la diferencia entre ellos radica en el fuzificador. En un SLD tipo 1 no *singleton* el fuzificador trata las entradas como conjuntos difusos tipo 1, lo cual produce un efecto al hacer la inferencia que será explicado más adelante. La salida del bloque de inferencia nuevamente es un conjunto difuso tipo 1, por lo que el proceso de defuzificación descrito para un SLD tipo 1 *singleton* aplica también para uno no *singleton*.

En la fuzificación no *singleton*, las mediciones $x_i = x'_i$ se mapean en números difusos; i.e., se asocian con una función de membresía. Específicamente hablando:

Un fuzificador no *singleton* es aquel para el cual $\mu_{x_i}(x'_i) = 1$, donde $i = 1, \dots, p$ y $\mu_{x_i}(x_i)$ decrece (de la unidad) a medida que x_i se aleja de x'_i .

En concepto, el fuzificador no *singleton* implica que el valor de entrada x'_i es el valor más posible de ser el correcto de entre los valores en su vecindad inmediata; sin embargo, debido a que la entrada es corrompida por ruido, los puntos vecinos posiblemente también son valores correctos, pero en menor grado.

La forma de la función de membresía $\mu_{x_i}(x_i)$ de estos fuzificadores no *singleton*² suele ser simétrica en x'_i , ya que lo más probable es que el efecto del ruido sea equivalente en todos los puntos. Ejemplos de estas de funciones de membresía son:

1. Gaussiana: $\mu_{x_i}(x_i) = \exp[-x(x_i - x'_i)/2\sigma^2]$
2. Triangular: $\mu_{x_i}(x_i) = \max(0, 1 - |(x_i - x'_i)/c|)$
3. Sin nombre: $\mu_{x_i}(x_i) = 1 / (1 + |(x_i - x'_i)/c|^n)$

²El fuzificador no *singleton* puede adoptar el nombre de la forma de la función de membresía que utilice [31].

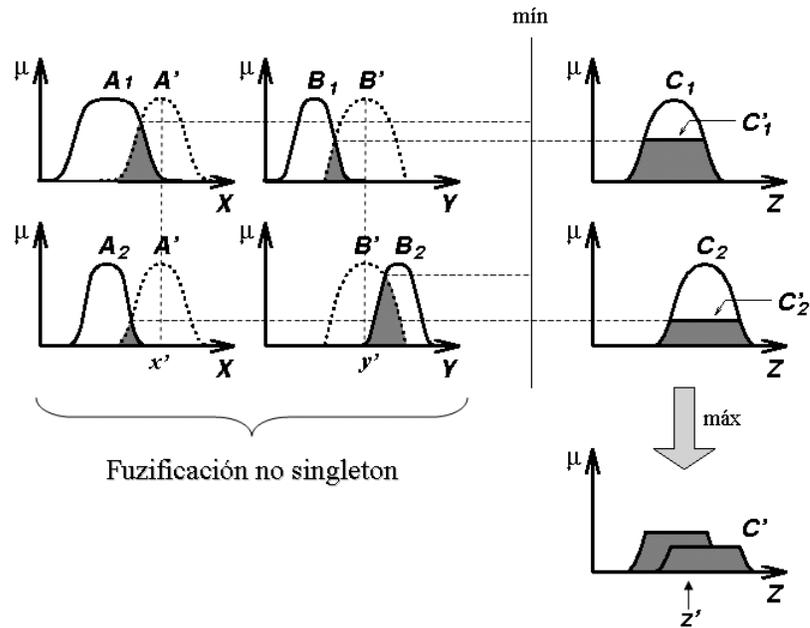


Figura 2.5: Mecanismo de inferencia de un SLD no *singleton* con composición máx-mín.

donde x'_i es el centro de los conjuntos difusos y $\sigma(c)$ es la dispersión de estos conjuntos. Mayores valores de la dispersión de estas funciones de membresía indican que se espera mayor ruido en los datos. Una vez seleccionada la forma de estas funciones de membresía (incluyendo sus parámetros), se puede realizar el proceso de razonamiento difuso como se explica a continuación.

El mecanismo de inferencia de un SLD no *singleton* se muestra en la figura 2.5, donde se puede observar que las entradas nítidas al sistema x' y y' establecen los centros de las funciones de membresía de los conjuntos difusos de entrada A' y B' , respectivamente; es decir, se modelan como número difusos. Éstos se utilizan para calcular el nivel de disparo de cada regla usando una norma T (mínimo o producto). A partir de aquí, el comportamiento del SLD no *singleton* es muy parecido al de un *singleton*. Aquí también se aplica una composición (máx-mín o máx-prod) con dicho nivel de disparo al conjunto difuso del consecuente respectivo. Luego, se combinan los consecuentes ponderados mediante una norma S (máximo) para así obtener el conjunto difuso de salida. Al igual que antes, se aplica una defuzificación para calcular el valor nítido de salida del sistema.

De manera similar, la salida inferida en cada regla depende de la composición utilizada [9]. Con el fin de recalcar la diferencia entre el fuzificador *singleton* (figura

2.3) y el no *singleton*, en la figura 2.5 se muestra el proceso de razonamiento resaltando la fuzificación no *singleton*. En este caso sólo se muestra el mecanismo de inferencia cuando se usa la composición máx-mín (con sus conjuntos de salida recortados), ya que la forma de la salida cuando se usa la composición máx-prod es muy similar a la que se muestra en la figura 2.4, (con sus conjuntos de salida escalados). Recuerde que en esa figura la fuzificación es distinta, pero el resto del mecanismo es igual.

El diseño de un SLD no *singleton* está asociado con el siguiente problema:

Dada una colección de N pares de datos *ruidosos* de entrada-salida de entrenamiento, $(\mathbf{x}^{(1)} : y^{(1)}), (\mathbf{x}^{(2)} : y^{(2)}), \dots, (\mathbf{x}^{(N)} : y^{(N)})$, donde \mathbf{x} es el vector *ruidoso* de entrada y y es la salida escalar de un SLD no *singleton*, la meta es especificar completamente el SLD no *singleton* usando los datos de entrenamiento.

Esta especificación, al igual que en un SLD *singleton*, implica hacer una selección de los componentes del sistema de la figura 2.1. Para ser consistentes con la selección hecha en el SLD *singleton*, en este sistema se utiliza una composición máx-prod, una implicación producto y un defuzificador de centro promedio (sin especificar una función de membresía), sólo que ahora se usa un fuzificador no *singleton*. Por lo tanto, la salida del sistema³ queda en función de su entrada \mathbf{x} con la siguiente expresión:

$$y(\mathbf{x}) = f_{ns}(\mathbf{x}) = \frac{\sum_{l=1}^M \bar{y}^l \prod_{i=1}^p \mu_{Q_k^l}(x_{k,\max}^l)}{\sum_{l=1}^M \prod_{i=1}^p \mu_{Q_k^l}(x_{k,\max}^l)} \quad (2.10)$$

donde p es el número de entradas, M es el número de reglas, l es el índice de la regla, \bar{y}^l es el centro del conjunto difuso del consecuente y $\mu_{Q_k^l}(x_{k,\max}^l)$ es una función auxiliar que depende de

$$\mu_{Q_k^l}(x_k) \equiv \mu_{x_k}(x_k) \star \mu_{F_k^l}(x_k) \quad (2.11)$$

donde $k = 1, \dots, p$; $l = 1, \dots, M$ y \star indica una norma T (mínimo o producto). Si además se define $x_{k,\max}^l$ como el valor de x_k donde $\mu_{Q_k^l}(x_k)$ alcanza su valor máximo⁴, entonces $\mu_{Q_k^l}(x_{k,\max}^l)$ queda definido como el máximo valor de membresía del conjunto resultante de la intersección (norma T o AND difuso) entre el número difuso de entrada y el conjunto difuso del antecedente de la variable correspondiente, y representa el nivel de disparo de la regla.

³Note el uso del subíndice “ns” en $f_{ns}(x)$ para indicar que es un SLD no *singleton*.

⁴Suponiendo que la operación *supremum* (máximo) ocurre sólo en un punto.

Cada regla, al igual que en la sección 2.1, tiene la forma dada en (2.2), donde F_i^l son los conjuntos difusos de los antecedentes que están asociados con los elementos de \mathbf{x} y están descritos por las funciones de membresía que se seleccionen; por ejemplo, si se eligen funciones de membresía Gaussianas como se hizo en la sección 2.1.2, se tiene

$$\mu_{F_i^l}(x_i) = \exp \left\{ -\frac{1}{2} \left(\frac{x_i - m_{F_i^l}}{\sigma_{F_i^l}} \right)^2 \right\} \quad (2.12)$$

donde $i = 1, \dots, p$ y $l = 1, \dots, M$. Además, cada entrada al SLD es modelada como un número difuso Gaussiano cuya función de membresía es

$$\mu_{X_k}(x_k) = \exp \left\{ -\frac{1}{2} \left(\frac{x_k - m_{X_k}}{\sigma_{X_k}} \right)^2 \right\} \quad (2.13)$$

donde $k = 1, \dots, p$ y $m_{X_k} = x^l$. Aquí se está suponiendo que la incertidumbre en cada entrada es la misma (como cuando se tiene un error estacionario); sin embargo, si no es el caso, entonces simplemente se debe reemplazar σ_X en (2.13) por σ_{X_k} . Cada método de diseño establece cómo especificar los parámetros $m_{F_i^l}$, $\sigma_{F_i^l}$ y σ_{X_k} de estas funciones de membresía, así como los centros de las funciones de membresía de los consecuentes, la \bar{y}^l en (2.10), usando los pares de entrenamiento $(\mathbf{x}^{(1)} : y^{(1)})$, $(\mathbf{x}^{(2)} : y^{(2)})$, \dots , $(\mathbf{x}^{(N)} : y^{(N)})$.

Antes de pasar a los métodos de diseño que se describieron en la sección 2.1, cabe mencionar que puede haber diferentes *diseños de entrenamiento* o de *ajuste* en un SLD no *singleton*, entre los que destacan los siguientes:

Diseño Dependiente: En este diseño, primero se obtiene un SLD *singleton* con el mejor desempeño posible, haciendo el ajuste de todos sus parámetros y entonces actualizar el sistema: (a) manteniendo todos los parámetros que sean compartidos por ambos sistemas, fijándolos en los valores obtenidos del mejor SLD *singleton* posible, y (b) ajustando sólo los parámetros nuevos del SLD no *singleton*. En este caso, sólo se ajustaría la desviación estándar σ_X .

Diseño Independiente: En este diseño, todos los parámetros del SLD no *singleton* son ajustados. Si por casualidad ya se ha diseñado un SLD *singleton*, entonces se pueden usar sus parámetros como valores iniciales o “semillas” en los algoritmos para realizar el ajuste de los parámetros; en cuyo caso podría llamarse *diseño parcialmente independiente*, y el que no utiliza semillas *diseño totalmente independiente*.

Podría pensarse que se obtiene el mejor desempeño con el diseño independiente; sin embargo, algunas veces es útil aplicar el diseño dependiente para observar la mejora marginal que se puede ganar del diseño anterior al diseño nuevo.

En las secciones siguientes se presentan las similitudes y diferencias con la sección 2.1 debidas a la naturaleza del SLD considerado en esta sección.

2.2.1 Método de un paso

En este método no se tienen cambios significativos, ya que se basa completamente en los datos de entrenamiento, es decir, el entrenamiento se sigue haciendo directamente con los pares de entrada–salida. Por lo que nuevamente se pueden diseñar hasta N reglas a partir de los datos de entrenamiento ahora ruidosos. Para completar este diseño, se debe determinar un valor para σ_X . Esto requiere que se sepa de antemano la varianza del ruido adicional en las mediciones, o que éste pueda ser estimado a partir de los datos; en su defecto, se debe usar otro método de diseño para determinar el valor de σ_X .

2.2.2 Método de gradiente descendente

En este método, se sigue haciendo un ajuste de todos los parámetros del sistema mediante un algoritmo de gradiente descendente de manera similar a la sección 2.1.2, sólo que debido al nuevo parámetro introducido, σ_X , se tienen unas pequeñas diferencias, por lo que se explica más a detalle que el de la sección 2.2.1. El objetivo de este método nuevamente es el de minimizar una función de error, en este caso:

$$e^{(i)} = \frac{1}{2}[f_{ns}(\mathbf{x}^{(i)}) - y^{(i)}]^2 \quad i = 1, \dots, N \quad (2.14)$$

donde $f_{ns}(\mathbf{x}^{(i)})$ es el valor obtenido al entrenar el sistema definido en (2.10) con el par de entrada–salida $(\mathbf{x}^{(i)} : y^{(i)})$. Al igual que en la sección 2.1.2, se completa la definición del sistema eligiendo funciones de membresía Gaussianas de la forma expresada en (2.12) y en (2.13), para los conjuntos difusos de los antecedentes y de las entradas, respectivamente. Por lo tanto, (2.10) se puede reexpresar como

$$y(\mathbf{x}^{(i)}) = f_{ns}(\mathbf{x}^{(i)}) = \frac{\sum_{l=1}^M \bar{y}^l \prod_{i=1}^p \mu_{Q_k^l}(x_{k,\max}^{l,(i)})}{\sum_{l=1}^M \prod_{i=1}^p \mu_{Q_k^l}(x_{k,\max}^{l,(i)})}$$

$$y(\mathbf{x}^{(i)}) = \frac{\sum_{l=1}^M \bar{y}^l \prod_{k=1}^p \exp \left[-\frac{1}{2} \frac{(x_k^{(i)} - m_{F_k^l})^2}{\sigma_X^2 + \sigma_{F_k^l}^2} \right]}{\sum_{l=1}^M \prod_{k=1}^p \exp \left[-\frac{1}{2} \frac{(x_k^{(i)} - m_{F_k^l})^2}{\sigma_X^2 + \sigma_{F_k^l}^2} \right]} \quad (2.15)$$

donde $i = 1, \dots, N$. A partir de (2.15), se puede ver que f_{ns} está caracterizado completamente por \bar{y}^l , $m_{F_k^l}$, $\sigma_{F_k^l}$ y σ_X , los cuales son los parámetros que se van a ajustar para minimizar el error en (2.14) usando las fórmulas siguientes:

$$\begin{aligned} m_{F_k^l}(i+1) &= m_{F_k^l}(i) - \alpha [f_s(\mathbf{x}^{(i)}) - y^{(i)}] [\bar{y}^l(i) - f_s(\mathbf{x}^{(i)})] \\ &\quad \times \frac{[x_k^{(i)} - m_{F_k^l}(i)]}{\sigma_X^2(i) + \sigma_{F_k^l}^2(i)} \phi_l(\mathbf{x}^{(i)}) \end{aligned} \quad (2.16)$$

$$\bar{y}^l(i+1) = \bar{y}^l(i) - \alpha [f_s(\mathbf{x}^{(i)}) - y^{(i)}] \phi_l(\mathbf{x}^{(i)}) \quad (2.17)$$

$$\begin{aligned} \sigma_{F_k^l}(i+1) &= \sigma_{F_k^l}(i) - \alpha [f_s(\mathbf{x}^{(i)}) - y^{(i)}] [\bar{y}^l(i) - f_s(\mathbf{x}^{(i)})] \\ &\quad \times \sigma_{F_k^l}(i) \left[\frac{x_k^{(i)} - m_{F_k^l}(i)}{\sigma_X^2(i) + \sigma_{F_k^l}^2(i)} \right]^2 \phi_l(\mathbf{x}^{(i)}) \end{aligned} \quad (2.18)$$

$$\begin{aligned} \sigma_X(i+1) &= \sigma_X(i) - \alpha [f_s(\mathbf{x}^{(i)}) - y^{(i)}] [\bar{y}^l(i) - f_s(\mathbf{x}^{(i)})] \\ &\quad \times \sigma_X(i) \left[\frac{x_k^{(i)} - m_{F_k^l}(i)}{\sigma_X^2(i) + \sigma_{F_k^l}^2(i)} \right]^2 \phi_l(\mathbf{x}^{(i)}) \end{aligned} \quad (2.19)$$

donde $i = 0, 1, \dots, (N-1)$ (pares de entrada-salida), $k = 1, \dots, p$ (entradas), $l = 1, \dots, M$ (reglas), y

$$\phi_l(\mathbf{x}^{(i)}) = \frac{\prod_{i=1}^p \mu_{Q_k^l}(x_{k,\max}^{l,(i)})}{\sum_{l=1}^M \prod_{i=1}^p \mu_{Q_k^l}(x_{k,\max}^{l,(i)})} \quad (2.20)$$

Las observaciones hechas en la sección 2.1.2 también aplican para esta sección.

2.3 Resumen

En este capítulo se estudiaron los pasos del razonamiento difuso de un SLDT1 típico representado por la figura 2.1. Se mostró que la diferencia entre los sistemas difusos más utilizados radica principalmente en los consecuentes de las reglas difusas; i.e., en los SLDs Mamdani los consecuentes son conjuntos difusos y en los SLDs TSK son funciones lineales que dependen de sus antecedentes. Habiendo establecido que los sistemas usados en este trabajo son únicamente SLDs Mamdani, se procedió a explicar los fuzificadores que estos sistemas pueden usar —singleton y no *singleton*—, mostrando con ejemplos gráficos los mecanismos de inferencia de cada uno de ellos. En estos ejemplos se ve claramente como la composición máx-mín recorta los conjuntos difusos consecuentes, mientras que la composición máx-prod los escala. Además, se presentaron dos métodos de diseño para construir los SLDT1s: el método de un paso (el cual no hace ajuste alguno en los parámetros del sistema) y el de gradiente descendente (que va ajustando los parámetros mientras se incrementa el número de época).

Capítulo 3

Sistemas de Lógica Difusa Tipo 2 (SLDT2)

Este capítulo da un marco teórico relativo a los SLDT2s, donde se expone la estructura de estos sistemas y su analogía con la de los SLDT1s. Además, se analizan los tipos de fuzificadores que pueden usarse junto con dos métodos de diseño para construir estos SLDT2s.

A menudo, el conocimiento usado para construir reglas en un sistema de lógica difusa (SLD) es incierto. Esta incertidumbre lleva a reglas que tienen antecedentes y/o consecuentes inciertos, lo cual a su vez se traduce en funciones de membresía de antecedentes y/o consecuentes inciertos [14]. Existen (al menos) cuatro formas en las que pueden ocurrir tales reglas con incertidumbre:

1. Cuando el significado de las palabras que son usadas en los antecedentes y consecuentes de reglas pueden ser inciertos (las palabras significan diferentes cosas para diferentes personas).
2. Cuando se encuesta a un grupo de expertos a menudo se obtienen consecuentes diferentes para la misma regla, porque los expertos no necesariamente estarán de acuerdo.
3. Cuando se usan datos de entrenamiento con ruido, como es el caso de toma de mediciones no estables [10].
4. Cuando los datos que son usados para ajustar los parámetros de un SLD tipo 1 también tienen ruido.

Todas estas fuentes de incertidumbre se traducen en incertidumbre acerca de las funciones de membresía de conjuntos difusos. Los conjuntos difusos tipo 1 no son capaces de modelar directamente dicha incertidumbre debido a que sus funciones de

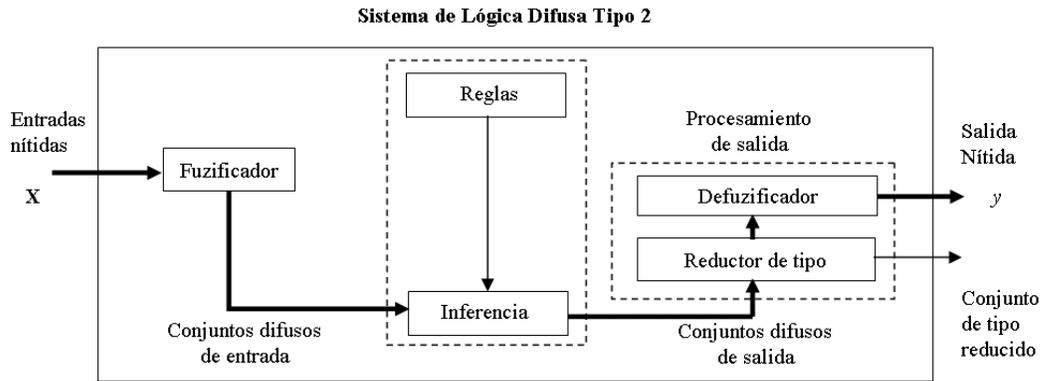


Figura 3.1: Estructura de un SLDT2. Para enfatizar la importancia del conjunto de tipo reducido, se muestran dos salidas del SLDT2, el conjunto de tipo reducido y el valor nítido defuzificado.

membresía son totalmente nítidas. Por otro lado, los conjuntos difusos tipo 2 son capaces de modelar dicha incertidumbre porque sus funciones de membresía son por sí mismas difusas. Las funciones de membresía de conjuntos difusos tipo 1 son bidimensionales, mientras que las de tipo 2 son tridimensionales. Es esta nueva tercera dimensión de los conjuntos difusos tipo 2 la que provee grados adicionales de libertad para hacer posible modelar directamente la incertidumbre.

La necesidad de esta nueva dimensión se comprende mejor haciendo una analogía entre estadística y lógica difusa. Por ejemplo, así como la varianza provee una medida de dispersión del promedio y es casi siempre usada para capturar incertidumbre probabilística en diseños prácticos basados en estadística, así también los sistemas de lógica difusa (SLD) necesitan alguna medida de dispersión, y parece ser tan fundamental para el diseño de sistemas que incluyen incertidumbre lingüística y/o numérica, que se traduce en reglas de incertidumbre como la varianza es a la media. De la misma manera que uno puede trabajar en modelación probabilística con momentos de orden mayor al segundo, también se pueden usar conjuntos mayores al tipo 2 en modelación difusa; pero a medida que se avanza a tipos mayores, la complejidad del sistema crece rápidamente. Es por eso que en este trabajo se utilizarán conjuntos tipo 2 máximo.

En la Figura 3.1 se muestra la estructura de un SLDT2 [11], la cual es muy similar a la de un tipo SLDT1 (figura 2.1); la única diferencia es que en este último, el bloque de *procesamiento de salida* sólo contiene el defuzificador. A continuación se hace una comparación entre de ambos esquemas, haciendo notar las diferencias en cada bloque.

En el SLDT1, el fuzificador mapea la entrada nítida a un conjunto difuso tipo 1; de igual manera, en el SLDT2 el conjunto obtenido es de tipo 2.

En el bloque central, la diferencia entre el tipo 1 y el tipo 2 está asociada con la naturaleza de las funciones de membresía, lo cual no es importante al formar las reglas; por lo tanto, la estructura de las reglas permanece exactamente igual en el caso del tipo 2, la única diferencia es que ahora algunos o todos los conjuntos involucrados son de tipo 2.

En un SLDT1, la máquina de inferencia combina reglas y da un mapeo de los conjuntos difusos tipo 1 de entrada a los conjuntos difusos tipo 1 de salida. Múltiples antecedentes en reglas están conectados por la norma t (intersección de conjuntos). Los grados de membresía en los conjuntos de entrada son combinados con los conjuntos de salida usando la composición *sup-star*. Múltiples reglas pueden ser combinadas usando la operación de conorma t (unión de conjuntos) o durante la defuzificación por suma ponderada. En el caso de tipo 2, el proceso de inferencia es muy similar. La máquina de inferencia combina reglas y da un mapeo de conjuntos difusos tipo 2 de entrada a conjuntos difusos tipo 2 de salida. Para hacer esto se necesita encontrar uniones e intersecciones de conjuntos tipo 2, así como también composiciones de relaciones tipo 2.

En un SLDT1, el defuzificador produce una salida nítida del conjunto difuso que es la salida de la máquina de inferencia. En un tipo 2, la salida de la máquina de inferencia es un conjunto tipo 2; así que se pueden usar “versiones extendidas” de los métodos de defuzificación tipo 1. Esta defuzificación extendida da un conjunto difuso tipo 1, y debido a que lleva de conjuntos de salida tipo 2 del SLD a un conjunto de tipo 1, a esta operación se le ha llamado “reducción de tipo” y al conjunto obtenido “conjunto de tipo reducido”.

Para obtener una salida nítida de un SLDT2, se puede defuzificar el conjunto de tipo reducido. La manera más natural de hacerlo parece ser encontrando el centroide del conjunto de tipo reducido, sin embargo, existen otras posibilidades, como escoger el punto de membresía más alto en este conjunto.

De la discusión anterior, se puede ver que para desarrollar un SLDT2, se necesita:

- llevar a cabo operaciones teóricas de conjuntos como unión, intersección y complemento en conjuntos de tipo 2;
- conocer propiedades¹ (e.g., leyes de conmutatividad, asociatividad, identidad)

¹Se puede encontrar una lista de estas propiedades en [10].

de grados de membresía de conjuntos tipo 2;

- lidiar con relaciones difusas tipo 2 y sus composiciones; y
- hacer reducción de tipo y defuzzificación para obtener una salida nítida del SLD.

Debido a que los SLDT2s generales son computacionalmente muy costosos, en este trabajo se utilizarán únicamente SLDT2s de intervalo², los cuales simplifican las operaciones entre conjuntos difusos tipo 2. Éstos conjuntos distribuyen de manera uniforme la incertidumbre entre todas las membresías primarias permisibles, tal como se requiere para este trabajo; además, según [17], no parece haber una justificación razonable para no usar conjuntos difusos tipo 2 de intervalo.

3.1 SLDT2 *singleton* de intervalo

El proceso de inferencia difusa en estos sistemas se resume en el siguiente teorema:

Teorema 3.1 [14] *En un SLDT2 singleton de intervalo con operación meet bajo norma T producto o mínimo: (a) el resultado de las operaciones entre las entradas y los antecedentes, las cuales están contenidas en el conjunto de disparo $\prod_{i=1}^p \mu_{\tilde{F}_i}(x'_i) \equiv F^l(\mathbf{x}')$, es un conjunto tipo 1 de intervalo, i.e.,*

$$F^l(\mathbf{x}') = [\underline{f}^l(\mathbf{x}'), \bar{f}^l(\mathbf{x}')] \equiv [\underline{f}^l, \bar{f}^l] \quad (3.1)$$

donde

$$\underline{f}^l(\mathbf{x}') = \underline{\mu}_{\tilde{F}_1}(x'_1) \star \cdots \star \underline{\mu}_{\tilde{F}_p}(x'_p) \quad (3.2)$$

y

$$\bar{f}^l(\mathbf{x}') = \bar{\mu}_{\tilde{F}_1}(x'_1) \star \cdots \star \bar{\mu}_{\tilde{F}_p}(x'_p) \quad (3.3)$$

(b) el conjunto consecuente de salida producido por la regla disparada R^l , es el conjunto difuso tipo 1 $\mu_{\tilde{B}^l}(y)$ expresado por

$$\mu_{\tilde{B}^l}(y) = \int_{b^l \in [\underline{f}^l \star \underline{\mu}_{\tilde{G}^l}(y), \bar{f}^l \star \bar{\mu}_{\tilde{G}^l}(y)]} 1/b^l, \quad y \in Y \quad (3.4)$$

²Recuerde que en este trabajo todos los SLDT2s utilizados son SLDT2s Mamdani, SLDT2s TSK se pueden encontrar en [13].

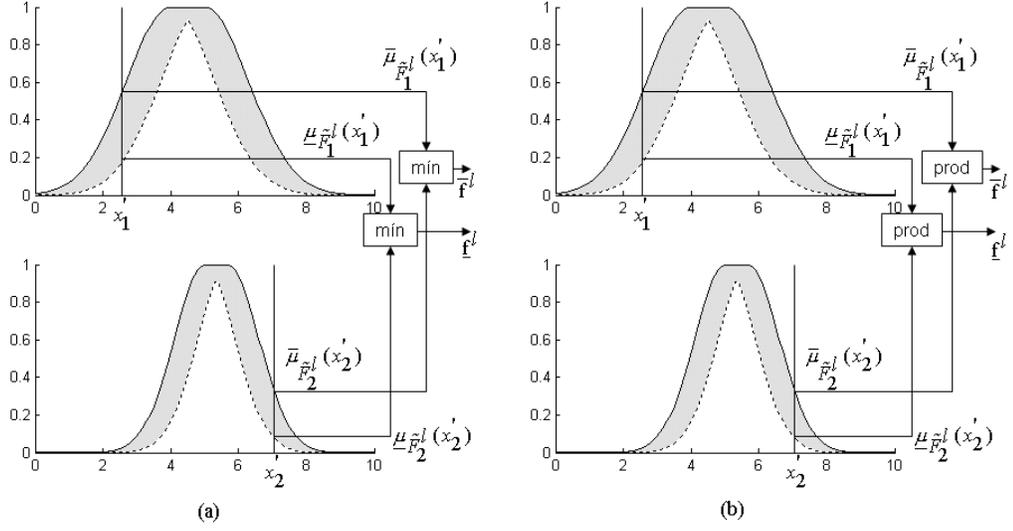


Figura 3.2: Descripción ilustrada de las operaciones entre las entradas y los antecedentes de un SLDT2 *singleton* de intervalo (a) usando norma T mínimo, (b) usando norma T producto.

donde $\underline{\mu}_{\tilde{G}^i}(y)$ y $\bar{\mu}_{\tilde{G}^i}(y)$ son los grados de membresía inferior y superior de $\mu_{\tilde{G}^i}(y)$.

(c) suponga que se disparan N de las M reglas en el SLD, donde $N \leq M$, y el conjunto difuso tipo 1 de salida compuesto se obtiene combinando los conjuntos consecuentes de salida disparados; i.e., $\mu_{\tilde{B}}(y) = \sqcup_{i=1}^N \mu_{\tilde{B}^i}(y) \quad y \in Y$; entonces,

$$\mu_{\tilde{B}}(y) = \int_{b^l \in [\underline{f}^1 \star \underline{\mu}_{\tilde{G}^1}(y)] \vee \dots \vee [\underline{f}^N \star \underline{\mu}_{\tilde{G}^N}(y)], [\bar{f}^1 \star \bar{\mu}_{\tilde{G}^1}(y)] \vee \dots \vee [\bar{f}^N \star \bar{\mu}_{\tilde{G}^N}(y)]} 1/b, \quad y \in Y \quad (3.5)$$

En la figura 3.2 se muestran las operaciones entre las entradas y los antecedentes [3.2 y 3.3] cuando se tienen reglas con dos antecedentes y un consecuente, una fuzificación *singleton* y normas T mínimo y producto. En todos los casos, en nivel de disparo es un conjunto tipo 1 de intervalo $[\underline{f}^l, \bar{f}^l]$, donde $\underline{f}^l = \underline{\mu}_{\tilde{F}_1^l}(x'_1) \star \underline{\mu}_{\tilde{F}_2^l}(x'_2)$ y $\bar{f}^l = \bar{\mu}_{\tilde{F}_1^l}(x'_1) \star \bar{\mu}_{\tilde{F}_2^l}(x'_2)$. Observe, por ejemplo, que $\underline{\mu}_{\tilde{F}_1^l}(x'_1)$ ocurre en la intersección de la línea vertical en x'_1 con $\underline{\mu}_{\tilde{F}_1^l}(x'_1)$. Para la norma T mínimo, se tiene que

$$\underline{f}^l = \min \left[\underline{\mu}_{\tilde{F}_1^l}(x'_1), \underline{\mu}_{\tilde{F}_2^l}(x'_2) \right] \quad (3.6)$$

y

$$\bar{f}^l = \min \left[\bar{\mu}_{\tilde{F}_1^l}(x'_1), \bar{\mu}_{\tilde{F}_2^l}(x'_2) \right] \quad (3.7)$$

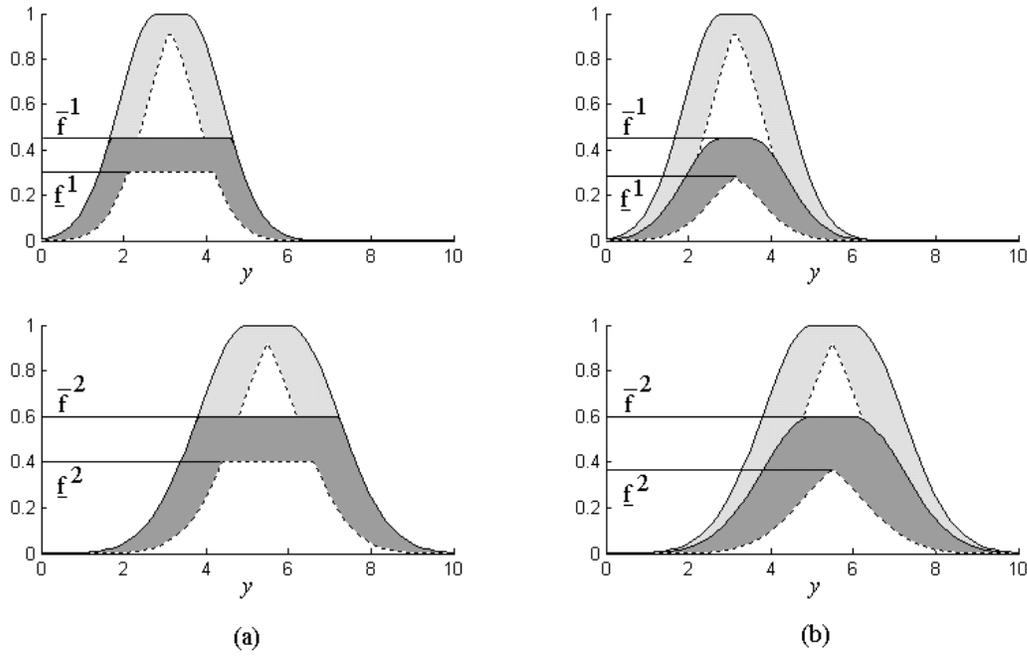


Figura 3.3: Descripción ilustrada de operaciones de consecuentes en un SLDT2 *singleton* de intervalo. (a) Conjuntos de salida disparados usando la norma T mínimo. (b) Conjuntos de salida disparados usando la norma T producto.

mientras que para la norma T producto, se tiene:

$$\underline{f}^l = \underline{\mu}_{\tilde{F}_1}(x'_1) \times \underline{\mu}_{\tilde{F}_2}(x'_2) \quad (3.8)$$

y

$$\overline{f}^l = \overline{\mu}_{\tilde{F}_1}(x'_1) \times \overline{\mu}_{\tilde{F}_2}(x'_2) \quad (3.9)$$

Lo más importante a observar en la figura 3.2 (a) y (b) es que, independientemente de la norma T utilizada, el resultado de las operaciones entre las entradas y los antecedentes es un intervalo —el intervalo de disparo—.

La figura 3.3 muestra al conjunto de salida de cada regla $\mu_{\tilde{B}^l}$ para un SLDT2 *singleton* de dos reglas ($l = 1, 2$). Cada conjunto se obtiene implementando (3.4) en cada $y \in Y$. Por ejemplo, a \overline{f}^1 se le aplica la norma T junto con la función de membresía superior $\overline{\mu}_{\tilde{G}_1}$ para dar la curva sólida $\overline{f}^1 \star \overline{\mu}_{\tilde{G}_1}$ para $\forall y \in Y$. La función de membresía primaria $\mu_{\tilde{B}^l}(y) \quad \forall y \in Y$ [i.e., la $FOU(\tilde{B}^l)$] es el área entre estas dos funciones, representada por el área oscura.

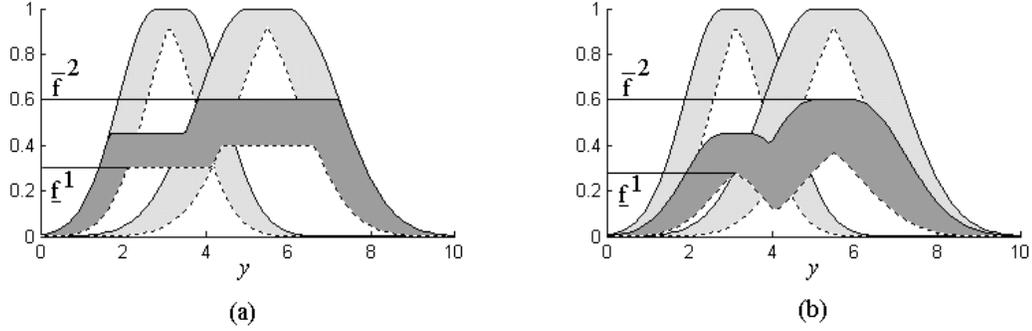


Figura 3.4: Descripción ilustrada del: (a) conjunto de salida compuesto por los dos conjuntos de salida disparados que se muestran en la figura 3.3 (a), (b) conjunto de salida compuesto por los dos conjuntos de salida disparados que se muestran en la figura 3.3 (b).

La figura 3.4 muestra el conjunto de salida tipo 2 combinado del SLDT2 *singleton* de dos reglas, donde los conjuntos de salidas disparados se combinan como en (3.5) usando la conorma T máximo. La curva sólida superior corresponde a $\left[\overline{f}^1 \star \overline{\mu}_{\tilde{G}^1}\right] \vee \left[\overline{f}^2 \star \overline{\mu}_{\tilde{G}^2}\right]$ para $\forall y \in Y$, mientras que la curva punteada corresponde a $\left[\underline{f}^1 \star \underline{\mu}_{\tilde{G}^1}\right] \vee \left[\underline{f}^2 \star \underline{\mu}_{\tilde{G}^2}\right]$ para $\forall y \in Y$. La función de membresía primaria $\mu_{\tilde{B}}(y) \quad \forall y \in Y$ [i.e., la $FOU(\tilde{B})$] es el área entre estas dos funciones, representada por el área oscura. Posteriormente se le aplica una reducción de tipo a $\mu_{\tilde{B}}(y) \quad \forall y \in Y$.

Basándose en el teorema 3.1, se sabe que para un SLDT2 *singleton* de intervalo que usa la norma T mínimo o producto, el resultado de las operaciones entre las entradas y los antecedentes es un conjunto tipo 1 de intervalo —el intervalo de disparo— el cual es determinado por sus puntos izquierdo y derecho, \underline{f}^l y \overline{f}^l , respectivamente. El conjunto consecuente de salida activo $\mu_{\tilde{B}^l}(y)$ de la regla R^l puede obtenerse a partir del intervalo de disparo usando (3.4), y es también un conjunto de intervalo.

El siguiente paso, después de la inferencia difusa, es la reducción de tipo. Sin importar qué método de reducción de tipo se elija, el conjunto de tipo reducido es también un conjunto de intervalo, y tiene la siguiente estructura:

$$Y_{RT} = [y_l, y_r] \quad (3.10)$$

Si se utiliza el método de los centros de conjuntos en la reducción de tipo, Y_{cdc} , se tiene la siguiente expresión:

$$Y_{cdc}(\mathbf{x}) = \int_{y^1 \in [y_l^1, y_r^1]} \cdots \int_{y^M \in [y_l^M, y_r^M]} \int_{f^1 \in [f_l^1, f_r^1]} \cdots \int_{f^M \in [f_l^M, f_r^M]} 1 \left/ \frac{\sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i} \right. \quad (3.11)$$

donde: $Y_{cdc}(\mathbf{x})$ es un conjunto de intervalo determinado por sus dos puntos, y_l y y_r ; y $[y_l^i, y_r^i]$ corresponde al centroide del conjunto consecuente tipo 2 de intervalo \tilde{G}^i , el cual se puede obtener de

$$C_{\tilde{G}^i} = \int_{\theta_1 \in J_{y_1}} \cdots \int_{\theta_N \in J_{y_N}} 1 \left/ \frac{\sum_{i=1}^N y_i \theta_i}{\sum_{i=1}^N \theta_i} \right. = [y_l^i, y_r^i] \quad (3.12)$$

Note que $[y_l^i, y_r^i]$ ($i = 1, \dots, M$) deben ser precalculados *antes* de realizar el cálculo de $Y_{cdc}(\mathbf{x})$.

Claramente, para calcular $Y_{cdc}(\mathbf{x})$ sólo es necesario calcular sus dos puntos límite, y_l y y_r . Si los valores de f^i y y^i que están asociados con y_l se hacen denotar por f_l^i y y_l^i , respectivamente, y además los valores de f^i y y^i que están asociados con y_r se hacen denotar por f_r^i y y_r^i , respectivamente; entonces, a partir de (3.11), se puede ver que

$$y_l = \frac{\sum_{i=1}^M f_l^i y_l^i}{\sum_{i=1}^M f_l^i} \quad (3.13)$$

y

$$y_r = \frac{\sum_{i=1}^M f_r^i y_r^i}{\sum_{i=1}^M f_r^i} \quad (3.14)$$

Note que, a pesar de que no se ha mostrado explícitamente, y_l , f_l^i , y_r y f_r^i dependen de la entrada \mathbf{x} del SLD; i.e., un cambio en \mathbf{x} produce valores diferentes para estas cantidades.

Para calcular y_l es necesario determinar $\{f_l^i, i = 1, \dots, M\}$ y sus $\{y_l^i, i = 1, \dots, M\}$ asociados, y para calcular y_r es necesario determinar $\{f_r^i, i = 1, \dots, M\}$ y sus $\{y_r^i, i = 1, \dots, M\}$ asociados. A continuación se presenta un procedimiento para calcular y_r .

Sin pérdida de generalidad, suponga que los y_r^i precalculados están arreglados en *orden ascendente*; i.e., $y_r^1 \leq y_r^2 \leq \cdots \leq y_r^M$. Entonces,

1. Calcule y_r en (3.14) inicializando $f_r^i = (\underline{f}^i + \bar{f}^i)/2$ para $i = 1, \dots, M$, donde \underline{f}^i y \bar{f}^i han sido previamente calculados usando (3.2) y (3.3), respectivamente, y haciendo $y_r' \equiv y_r$.

2. Encuentre R ($1 \leq R \leq M-1$) tal que $y_r^R \leq y_r' \leq y_r^{R+1}$.
3. Calcule y_r en (3.14) con $f_r^i = \underline{f}^i$ para $i \leq R$ y $f_r^i = \overline{f}^i$ para $i > R$, y haga $y'' \equiv y_r$.
4. Si $y_r'' \neq y_r'$, entonces vaya al paso 5. Si $y_r'' \equiv y_r'$
5. Haga y_r' igual a y_r'' , y regrese al paso 2.

Observe que en este procedimiento, el número R es muy importante. Para $i \leq R$, $f_r^i = \underline{f}^i$, mientras que para $i > R$, $f_r^i = \overline{f}^i$; por lo tanto, y_r en (3.14) puede ser representado como

$$y_r = y_r(\underline{f}^1, \dots, \underline{f}^R, \overline{f}^{R+1}, \dots, \overline{f}^M, y_r^1, \dots, y_r^M) \quad (3.15)$$

El procedimiento para calcular y_l es muy similar al de y_r , sólo se debe remplazar y_r^i por y_l^i , y, en el paso 2 encontrar L ($1 \leq L \leq M-1$) tal que $y_l^L \leq y_l' \leq y_l^{L+1}$. Adicionalmente, en el paso 3, ahora se debe calcular y_l con (3.13) con $f_l^i = \overline{f}^i$ para $i \leq L$ y $f_l^i = \underline{f}^i$ para $i > L$. Entonces y_l en (3.13) se puede representar como

$$y_l = y_l(\overline{f}^1, \dots, \overline{f}^L, \underline{f}^{L+1}, \dots, \underline{f}^M, y_l^1, \dots, y_l^M) \quad (3.16)$$

Esto completa la reducción de tipo para un SLDT2 *singleton* de intervalo.

Debido a que Y_{cdc} es un conjunto de intervalo, éste se puede defuzificar usando el promedio de y_l y y_r ; por lo que finalmente se puede obtener la salida defuzificada de un SLDT2 *singleton* de intervalo con

$$y(\mathbf{x}) = f_{s2}(\mathbf{x}) = \frac{y_l + y_r}{2} \quad (3.17)$$

Los SLDT2s de intervalo aquí diseñados, al igual que en la sección 2.1, están asociados con el siguiente problema:

Dada una colección de N pares de datos de entrada–salida de entrenamiento, $(\mathbf{x}^{(1)} : y^{(1)})$, $(\mathbf{x}^{(2)} : y^{(2)})$, \dots , $(\mathbf{x}^{(N)} : y^{(N)})$, donde \mathbf{x} es el vector de entrada y y es la salida escalar de un SLDT2 *singleton* de intervalo, la meta es especificar completamente el SLDT2 usando los datos de entrenamiento.

En esta sección se utilizan funciones de membresía Gaussianas con media incierta y funciones de membresía secundarias de intervalo [ver figura B.4 (a)], implicación producto, norma T producto, reducción de tipo usando centro de conjuntos, y defuzificación usando el centroide del conjunto de tipo reducido.

Como en la sección 2.1, los datos de entrenamiento pueden ser interpretados como una colección de reglas SI-ENTONCES. Ahora, cada regla tiene la forma

$$R^l: \text{ SI } x_1 \text{ es } \tilde{F}_1^l \text{ y } x_2 \text{ es } \tilde{F}_2^l \text{ y } \dots \text{ y } x_p \text{ es } \tilde{F}_p^l, \text{ ENTONCES } y \text{ es } \tilde{G}^l \quad l = 1, 2, \dots, M \quad (3.18)$$

donde \tilde{F}_i^l son conjuntos difusos tipo 2 asociados con los elementos de $\mathbf{x}^{(l)}$ y están descritos por funciones de membresía primarias Gaussianas con media incierta, i.e.,

$$\mu_i^l(x_i) = \exp \left[-\frac{1}{2} \left(\frac{x_i - m_i^l}{\sigma_i^l} \right)^2 \right] \quad m_i^l \in [m_{i1}^l, m_{i2}^l] \quad (3.19)$$

donde $i = 1, \dots, p$ y $l = 1, \dots, M$. Los conjuntos consecuentes también son funciones de membresía primarias de intervalo Gaussianas con media incierta, i.e.,

$$\mu^j(y) = \exp \left[-\frac{1}{2} \left(\frac{y - m^j}{\sigma^j} \right)^2 \right] \quad m^j \in [m_1^j, m_2^j] \quad j = 1, \dots, M \quad (3.20)$$

Note que el centroide de cada $\mu^j(y)$, $C_{\tilde{G}^j}$ es un conjunto tipo 1 de intervalo, i.e.,

$$C_{\tilde{G}^j} = [y_l^j, y_r^j] \quad j = 1, \dots, M \quad (3.21)$$

Cada método de diseño establece cómo especificar los parámetros de las funciones de membresía en (3.19) y (3.20) usando los pares de entrenamiento $(\mathbf{x}^{(1)} : y^{(1)})$, $(\mathbf{x}^{(2)} : y^{(2)})$, \dots , $(\mathbf{x}^{(N)} : y^{(N)})$. Asimismo, el número de reglas que puede haber en un SLDT2 *singleton* de intervalo también depende del método de diseño usado para construir el sistema. Si no se hace ajuste en los parámetros del SLD, pueden haber hasta $M = N$ reglas. Si se hace ajuste, y tomando en cuenta que típicamente hay menos parámetros de diseño ajustados que pares de entrenamiento, entonces se deben tener menos de N reglas.

Antes de describir los métodos de diseño, cabe mencionar que en los SLDT2s *singleton* de intervalo, como en los no *singleton* tipo 1, pueden haber distintos diseños de entrenamiento (o de ajuste del sistema), los cuales se describen a continuación:

Diseño Dependiente: En este diseño, primero se obtiene el SLDT1 con el mejor desempeño posible haciendo el ajuste de todos sus parámetros, y entonces se usan estos parámetros de alguna manera para *inicializar* los parámetros del SLDT2 *singleton* de intervalo; por ejemplo, haciendo unas pequeñas modificaciones a los parámetros finales de las funciones de membresía tipo 1 para inicializar los parámetros de los conjuntos difusos tipo 2.

Diseño Independiente: En este diseño, los parámetros del SLDT2 *singleton* de intervalo son ajustados: (a) sin el beneficio de un diseño anterior de tipo 1, en cuyo caso podría llamarse *diseño totalmente independiente*; y (b) usando (de manera indirecta) parámetros ya ajustados de sistemas tipo 1, por ejemplo, usando σ_X del SLDT1 no *singleton* para calcular los valores de los parámetros iniciales de los conjuntos difusos tipo 2; a este diseño se le llama *diseño parcialmente independiente*.

En el resto de la sección se describen los dos métodos de diseño presentados en la sección 2.1, sólo que ahora se aplican a SLDT2s *singleton* de intervalo.

3.1.1 Método de un paso

Este método se propone como una extensión del método de un paso expuesto en la sección 2.1, utilizando ahora conjuntos difusos tipo 2 de intervalo, por lo que es necesario especificar un rango de incertidumbre, i.e., definir los límites izquierdo y derecho de la media de cada función de membresía primaria Gaussiana de la ecuación (3.19) usando los datos de entrenamiento. Como el entrenamiento se hace directamente con los N pares de entrada–salida ruidosos, se generan N reglas.

3.1.2 Método de gradiente descendente

En este método se hace un ajuste de los parámetros del SLDT2 *singleton* de intervalo usando un algoritmo de gradiente descendente. El entrenamiento del sistema se basa en pares de entrada–salida $(\mathbf{x}^{(t)} : y^{(t)})$, y el objetivo es minimizar la siguiente función de error:

$$e^{(t)} = \frac{1}{2}[f_{s2}(\mathbf{x}^{(t)}) - y^{(t)}]^2 \quad t = 1, \dots, N \quad (3.22)$$

Los únicos que determinan $f_{s2}(\mathbf{x}^{(t)})$ son las funciones de membresía superior e inferior de los antecedentes y los puntos límite del centroide de cada consecuente; por lo tanto, se puede pensar conceptualmente en minimizar (3.22) con respecto a estas funciones de membresía superior e inferior y a los parámetros del consecuente y_l^j y y_r^j . Adicionalmente, para cada $\mathbf{x}^{(t)}$ el sistema puede ser caracterizado por dos funciones [(3.16) y (3.15)], las cuales generan los puntos izquierdo y derecho del conjunto de tipo reducido; por lo que para cada $\mathbf{x}^{(t)}$, uno puede enfocarse únicamente en el ajuste de los parámetros de esos dos SLDs tipo 1. Note que los diseños de estos SLDT1s están *acoplados* debido a que $f_{s2}(\mathbf{x}^{(t)})$ en (3.22) depende tanto de $y_l(\mathbf{x}^{(t)})$ como de $y_r(\mathbf{x}^{(t)})$. Hecho esto, se tiene el conjunto de tipo reducido, al cual se le aplica una defuzificación para obtener finalmente la salida nítida del sistema.

3.2 SLDT2 no singleton de intervalo

El proceso de inferencia difusa en estos sistemas se resume en el siguiente teorema:

Teorema 3.2 [14] *En un SLDT2 no singleton de intervalo con operación meet bajo norma T producto o mínimo: (a) el resultado de las operaciones entre las entradas y los antecedentes, las cuales están contenidas en el conjunto de disparo $\sqcup_{\mathbf{x} \in \mathbf{X}} [\prod_{k=1}^p \mu_{\tilde{Q}_k^l}(x_k)] = \sqcup_{\mathbf{x} \in \mathbf{X}} \mu_{\tilde{Q}^l}(\mathbf{x}) \equiv F^l(\mathbf{x}')$, es un conjunto tipo 1 de intervalo, i.e.,*

$$F^l(\mathbf{x}') = [\underline{f}^l(\mathbf{x}'), \bar{f}^l(\mathbf{x}')] \equiv [\underline{f}^l, \bar{f}^l] \quad (3.23)$$

donde

$$\underline{f}^l(\mathbf{x}') = \sup_{\mathbf{x}} \int_{x_1 \in X_1} \cdots \int_{x_p \in X_p} [\mu_{X_1}(x_1) \star \underline{\mu}_{\tilde{F}_1^l}(x_1)] \star \cdots \star [\mu_{X_p}(x_p) \star \underline{\mu}_{\tilde{F}_p^l}(x_p)] / \mathbf{x} \quad (3.24)$$

y

$$\bar{f}^l(\mathbf{x}') = \sup_{\mathbf{x}} \int_{x_1 \in X_1} \cdots \int_{x_p \in X_p} [\mu_{X_1}(x_1) \star \bar{\mu}_{\tilde{F}_1^l}(x_1)] \star \cdots \star [\mu_{X_p}(x_p) \star \bar{\mu}_{\tilde{F}_p^l}(x_p)] / \mathbf{x} \quad (3.25)$$

y el supremum es alcanzado cuando cada término en los corchetes alcanza su supremum.

(b) el conjunto consecuente de salida producido por la regla disparada R^l , $\mu_{\tilde{B}^l}(y) = \mu_{\tilde{G}^l}(y) \cap F^l(\mathbf{x}') \ y \in Y$, es un conjunto difuso tipo 1, donde

$$\mu_{\tilde{B}^l}(y) = \int_{b^l \in [\underline{f}^l \star \underline{\mu}_{\tilde{G}^l}(y), \bar{f}^l \star \bar{\mu}_{\tilde{G}^l}(y)]} 1/b^l, \ y \in Y \quad (3.26)$$

donde $\underline{\mu}_{\tilde{G}^l}(y)$ y $\bar{\mu}_{\tilde{G}^l}(y)$ son los grados de membresía inferior y superior de $\mu_{\tilde{G}^l}(y)$.

(c) suponga que se disparan N de las M reglas en el SLD, donde $N \leq M$, y el conjunto difuso tipo 1 de salida compuesto se obtiene combinando los conjuntos consecuentes de salida disparados; i.e., $\mu_{\tilde{B}}(y) = \sqcup_{i=1}^N \mu_{\tilde{B}^i}(y) \ y \in Y$; entonces,

$$\mu_{\tilde{B}}(y) = \int_{b^l \in [\underline{f}^1 \star \underline{\mu}_{\tilde{G}^1}(y) \vee \cdots \vee \underline{f}^N \star \underline{\mu}_{\tilde{G}^N}(y), \bar{f}^1 \star \bar{\mu}_{\tilde{G}^1}(y) \vee \cdots \vee \bar{f}^N \star \bar{\mu}_{\tilde{G}^N}(y)]} 1/b, \ y \in Y \quad (3.27)$$

Cabe resaltar que como parte del proceso para encontrar la salida de un SLDT2 no *singleton* de intervalo, es necesario realizar la operación *meet* entre un conjunto tipo 1 de entrada y un conjunto tipo 2 antecedente, para lo cual basta con aplicar la norma T entre la función de membresía de entrada y las funciones de membresía inferior y superior del antecedente. El resultado es un conjunto de intervalo representado por $\mu_{\tilde{Q}_k^l}(x_k)$, el cual puede expresarse en términos de sus funciones de membresía inferior y superior como

$$\underline{\mu}_{\tilde{Q}_k^l}(x_k) = \int_{x_k \in X_k} \left[\mu_{X_k}(x_k) \star \underline{\mu}_{\tilde{F}_k^l}(x_k) \right] / x_k = \sup_{x_k} \underline{\mu}_{\tilde{Q}_k^l}(x_k) \quad (3.28)$$

y

$$\bar{\mu}_{\tilde{Q}_k^l}(x_k) = \int_{x_k \in X_k} \left[\mu_{X_k}(x_k) \star \bar{\mu}_{\tilde{F}_k^l}(x_k) \right] / x_k = \sup_{x_k} \bar{\mu}_{\tilde{Q}_k^l}(x_k) \quad (3.29)$$

Por su parte, se sabe que al evaluar (3.24) y (3.25), el *supremum* es alcanzado cuando cada término en los corchetes alcanza su *supremum*; por lo tanto, en el bloque del mecanismo de inferencia de un SLDT2 no *singleton* de intervalo se examinará

$$\underline{f}_k^l(x'_k) \equiv \sup_{x_k} \int_{x_k \in X_k} \left[\mu_{X_k}(x_k) \star \underline{\mu}_{\tilde{F}_k^l}(x_k) \right] / x_k = \sup_{x_k} \underline{\mu}_{\tilde{Q}_k^l}(x_k) \quad (3.30)$$

y

$$\bar{f}_k^l(x'_k) \equiv \sup_{x_k} \int_{x_k \in X_k} \left[\mu_{X_k}(x_k) \star \bar{\mu}_{\tilde{F}_k^l}(x_k) \right] / x_k = \sup_{x_k} \bar{\mu}_{\tilde{Q}_k^l}(x_k) \quad (3.31)$$

donde $k = 1, \dots, p$, y se han usado (3.28) y (3.29).

Si $\underline{x}_{k,\max}^l$ y $\bar{x}_{k,\max}$ denotan los valores de x_k que están asociados con $\sup_{x_k} \underline{\mu}_{\tilde{Q}_k^l}(x_k)$ y $\sup_{x_k} \bar{\mu}_{\tilde{Q}_k^l}(x_k)$, respectivamente, entonces (3.30 y (3.31) se pueden reexpresar como

$$\underline{f}_k^l(x'_k) = \underline{\mu}_{\tilde{Q}_k^l}(\underline{x}_{k,\max}^l) \quad (3.32)$$

y

$$\bar{f}_k^l(x'_k) = \bar{\mu}_{\tilde{Q}_k^l}(\bar{x}_{k,\max}) \quad (3.33)$$

Esto significa, por supuesto, que $\underline{f}^l(\mathbf{x}')$ y $\bar{f}^l(\mathbf{x}')$ en (3.24) y (3.25) pueden ser reexpresados como

$$\underline{f}^l(\mathbf{x}') = T_{k=1}^p \underline{f}_k^l(x'_k) = T_{k=1}^p \underline{\mu}_{\tilde{Q}_k^l}(\underline{x}_{k,\max}^l) \quad (3.34)$$

y

$$\bar{f}^l(\mathbf{x}') = T_{k=1}^p \bar{f}_k^l(x'_k) = T_{k=1}^p \bar{\mu}_{\bar{Q}_k^l}(\bar{x}_{k,\max}^l) \quad (3.35)$$

El siguiente procedimiento resume los pasos para calcular $F^l(\mathbf{x}')$ en (3.23):

1. Escoger una norma T (producto o mínimo) y crear las funciones $\underline{\mu}_{\bar{Q}_k^l}(x_k)$ y $\bar{\mu}_{\bar{Q}_k^l}(x_k)$ usando (3.28) y (3.29), respectivamente.
2. Calcular $\underline{x}_{k,\max}^l$ y $\bar{x}_{k,\max}$ maximizando $\underline{\mu}_{\bar{Q}_k^l}(x_k)$ y $\bar{\mu}_{\bar{Q}_k^l}(x_k)$, respectivamente.
3. Evaluar $\underline{f}_k^l(x'_k)$ y $\bar{f}_k^l(x'_k)$ usando (3.32) y (3.33), respectivamente.
4. Calcular $\underline{f}^l(\mathbf{x}')$ y $\bar{f}^l(\mathbf{x}')$ usando (3.34) y (3.35), respectivamente.

En la figura 3.5 se muestran las operaciones [(3.30)–(3.35)] entre las entradas y los antecedentes cuando se tienen reglas con dos antecedentes y un consecuente, una fuzificación no *singleton*, y normas T mínimo o producto. En todos los casos, la fuerza de activación o disparo es un conjunto tipo 1 de intervalo $[\underline{f}^l, \bar{f}^l]$, donde $\underline{f}^l = \underline{f}_1^l \star \underline{f}_2^l$ y $\bar{f}^l = \bar{f}_1^l \star \bar{f}_2^l$. De acuerdo con (3.30), \underline{f}_k^l es el *supremum* de la fuerza de disparo entre la norma T de $\mu_{X_k}(x_k)$ y la función de membresía inferior $\underline{\mu}_{\bar{F}_k^l}(x_k)$, y de acuerdo con (3.31), \bar{f}_k^l es el *supremum* de la fuerza de disparo entre la norma T de $\mu_{X_k}(x_k)$ y la función de membresía superior $\bar{\mu}_{\bar{F}_k^l}(x_k)$ ($k = 1, 2$). Note que $\mu_{X_k}(x_k)$ está centrada en $x_k = x'_k$. Estas normas T se muestran como curvas gruesas en la figura 3.5 (a) y (b). A partir de estas curvas gruesas es fácil elegir sus supremos.

Lo más importante a observar en la figura 3.5 (a) y (b) es que, independientemente de la norma T utilizada, el resultado de las operaciones entre las entradas y los antecedentes es un intervalo —el intervalo de disparo—, como en la figura 3.2. Como consecuencia de esto, los resultados mostrados en las figuras 3.3 [$\mu_{\bar{B}^l}(y)$, $y \in Y$, para un SLD de dos reglas] y 3.4 (conjunto de salida combinado para un SLD de dos reglas), se aplican de igual manera a un SLDT2 no *singleton* de intervalo. Así, la única diferencia entre un SLDT2 no *singleton* de intervalo y un SLDT2 *singleton* de intervalo está en sus intervalos de disparo. Note que el presente caso de fuzificación no *singleton* produce un intervalo de disparo que incluye la *incertidumbre adicional* de las entradas no *singleton* de tipo 1.

En cuanto al diseño de SLDT2s no *singleton* de intervalo, los métodos aquí utilizados (como en la sección 3.1), están asociados con el siguiente problema:

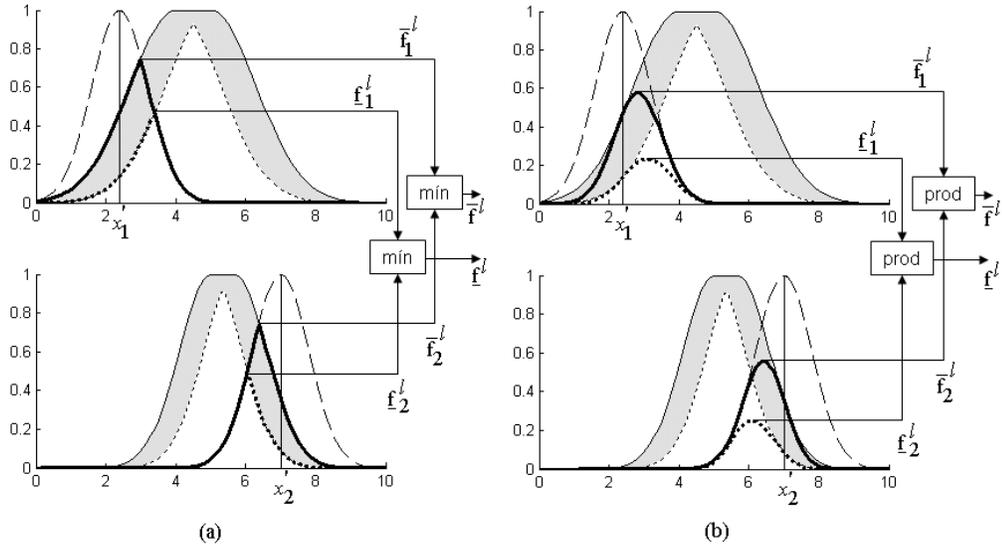


Figura 3.5: Descripción ilustrada de las operaciones entre las entradas y los antecedentes de un SLDT2 no *singleton* de intervalo (a) usando norma T mínimo, (b) usando norma T producto.

Dada una colección de N pares de datos de entrada–salida de entrenamiento, $(\mathbf{x}^{(1)} : y^{(1)}), (\mathbf{x}^{(2)} : y^{(2)}), \dots, (\mathbf{x}^{(N)} : y^{(N)})$, donde \mathbf{x} es el vector de entrada y y es la salida escalar de un SLDT2 no *singleton* de intervalo, la meta es especificar completamente el SLDT2 usando los datos de entrenamiento.

Para ser consistentes con la sección 3.1, en esta sección se utilizan funciones de membresía Gaussianas con media incierta y funciones de membresía secundarias de intervalo, implicación producto, norma T producto, reducción de tipo usando centro de conjuntos, y defuzzificación usando el centroide del conjunto de tipo reducido. Adicionalmente, la naturaleza de tipo 1 de las mediciones de entrada se describe por funciones de membresía Gaussianas que están centradas en los valores de medición, i.e., las entradas se modelan como números difusos Gaussianos cuyas funciones de membresía están dadas por (2.13), donde $m_{x_k} = x^l$.

Los datos de entrenamiento se pueden interpretar como una colección de reglas SI–ENTONCES de la forma (3.18), de la misma manera que en la sección 3.1. La definición de los parámetros de las funciones de membresía, así como el número de reglas del sistema, dependen del método de diseño (e.g., el método de un paso o el de gradiente descendente). Independientemente del método de diseño, los SLDT2s no *singleton* de intervalo pueden ser ajustados mediante los siguientes diseños de

entrenamiento:

Diseño Dependiente: En este diseño, primero se obtiene el SLDT2 *singleton* de intervalo con el mejor desempeño posible haciendo el ajuste de todos sus parámetros, y entonces se actualiza el sistema: (a) manteniendo todos los parámetros que sean compartidos por ambos sistemas, fijándolos en los valores obtenidos del mejor SLDT2 *singleton* de intervalo, y (b) ajustando sólo los parámetros nuevos del SLDT2 no *singleton* de intervalo. En este caso, sólo se ajustaría la desviación estándar σ_X .

Diseño Independiente: En este diseño, todos los parámetros del SLDT2 no *singleton* de intervalo son ajustados. Si se cuenta con el diseño de un SLDT2 *singleton* de intervalo ya terminado, entonces se pueden usar sus parámetros finales como semillas en los algoritmos para realizar el ajuste de los parámetros; a este caso se le llamó *diseño parcialmente independiente*, y al que no utiliza semillas *diseño totalmente independiente*.

A continuación se presentan las similitudes y diferencias con la sección 3.1 debidas a la naturaleza del SLDT2 no *singleton* de intervalo.

3.2.1 Método de un paso

En este método no se tienen cambios significativos, ya que se basa completamente en los datos de entrenamiento, es decir, el entrenamiento se sigue haciendo directamente con los pares de entrada–salida, por lo que nuevamente se pueden diseñar hasta N reglas a partir de los datos de entrenamiento ruidosos. Para completar este diseño, se debe determinar un valor para σ_X . Esto requiere que se sepa de antemano la varianza del ruido adicional en las mediciones, o que éste pueda ser estimado a partir de los datos; en su defecto, se debe usar otro método de diseño para determinar el valor de σ_X .

3.2.2 Método de gradiente descendente

En este método se hace un ajuste de los parámetros del SLDT2 no *singleton* de intervalo usando un algoritmo de gradiente descendente.

Al igual que en la sección 3.1.2, se tienen N pares de entrenamiento de entrada–salida $(\mathbf{x}^{(t)} : y^{(t)})$, $t = 1, \dots, N$; y el objetivo es ajustar los parámetros de diseño de tal manera que se minimice la función de error siguiente:

$$e^{(t)} = \frac{1}{2} [f_{ns2}(\mathbf{x}^{(t)}) - y^{(t)}]^2 \quad t = 1, \dots, N \quad (3.36)$$

donde $f_{ns2}(\mathbf{x}^{(t)})$ depende, nuevamente, de las funciones de membresía superior e inferior de los antecedentes y los puntos límite del centroide de cada consecuente; por lo tanto, se puede pensar conceptualmente en minimizar (3.36) con respecto a estas funciones de membresía superior e inferior y a los parámetros del consecuente y_l^j y y_r^j , incluyendo la función de membresía de la entrada dada por (2.13). Al igual que en la sección 3.1.2, el cálculo de $f_{ns2}(\mathbf{x}^{(t)})$ en (3.36) depende tanto de $y_l(\mathbf{x}^{(t)})$ como de $y_r(\mathbf{x}^{(t)})$. Calculando ambos términos, se tiene el conjunto de tipo reducido, al cual se le aplica una defuzificación para obtener finalmente la salida nítida del sistema.

3.3 Resumen

En este capítulo se justificó el uso de los SLDT2s, los cuales surgen debido a la necesidad de modelar mejor la incertidumbre causada por el ruido en los datos de entrenamiento de los sistemas. Además, se hizo una analogía de la estructura de los SLDT2s típicos con la de los SLDT1s, resaltando el hecho de que el único bloque que se agrega en los de tipo 2 es el “Reductor de tipo”, ver figura 3.1. Asimismo, se mostró cómo, tanto en los SLDT2 de intervalo *singleton* como en los no *singleton*, los mecanismos de inferencia dependen principalmente de los valores de membresía superior e inferior de los conjuntos difusos tipo 2 que forman parte de las reglas. Por último, se presentaron los mismos métodos de diseño que en el capítulo 2 (i.e., el método de un paso y el de gradiente descendente), sólo que en este caso se usaron para construir SLDT2s.

Capítulo 4

Experimentos y Resultados

En este capítulo se exponen los experimentos realizados en el diseño de los SLDs, así como los resultados obtenidos en cada uno de ellos. Recuerde que, como se menciona en el capítulo 2, todos los sistemas aquí utilizados son SLDs Mamdani, los cuales fueron diseñados bajo una metodología que se explica en la sección 4.1.

El primer diseño que se presenta en este capítulo es el de los SLDT1s *singleton*, donde se tiene un entrenamiento sin ruido y dos tipos de pronóstico, uno sin ruido y otro con ruido. Enseguida aparece el diseño de los SLDT1s no *singleton*, donde tanto el entrenamiento y como el pronóstico son con ruido. Luego, se muestran los diseños de los SLDT2s *singleton* y no *singleton*, secciones 4.4 y 4.5, respectivamente; donde nuevamente el entrenamiento y el pronóstico se hacen con ruido. Por último, se tiene un experimento adicional donde se utilizan los sistemas ya entrenados en las secciones anteriores, para hacer el pronóstico de una serie Mackey-Glass con condiciones iniciales corrompidas por ruido, lo que genera una serie con apariencia totalmente distinta a la forma de la serie utilizada para el entrenamiento de dichos sistemas. Además, esta nueva serie sigue presentando ruido adicional a lo largo del tiempo.

4.1 Metodología de experimentación

Para el diseño de los SLDs (sin importar su tipo), se siguió la metodología mostrada en el diagrama de bloques de la figura 4.1. A continuación se da una descripción de cada uno de sus pasos:

- 1. Simulación de la serie.** Primero se obtienen los datos de la serie de tiempo a partir de la ecuación Mackey-Glass.
- 2. Conjuntos de entrenamiento y prueba.** A partir de los datos obtenidos en el paso anterior, se hace una división para obtener el conjunto de entrenamiento y el conjunto de prueba. (Ver apéndice A).

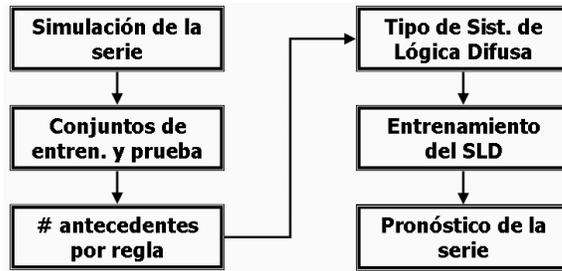


Figura 4.1: Diagrama de bloques de la metodología para el diseño de los SLD.

3. **Número de antecedentes por regla.** En este paso se define el número de antecedentes que tendrá cada regla en todos los sistemas. Recuerde que el número de antecedentes depende del número de variables de entrada al SLD. Este valor se mantiene fijo para todos los diseños, así que es el único paso que no se vuelve a hacer.
4. **Definición del tipo de SLD.** Aquí se especifican los datos del sistema que se vaya a utilizar; estos datos incluyen el tipo de fuzificador (*singleton*, no *singleton*) que se va a usar, las formas de las funciones de membresía (triangular, trapezoidal, Gaussiana, lineal por segmentos), el tipo de composición (máx-mín, máx-producto), el tipo de implicación (mínimo, producto) y el tipo de defuzificador (centroide, centro promedio, etc.).
5. **Entrenamiento del SLD.** El entrenamiento del sistema depende del método de diseño que se esté utilizando; lo que implica que la elaboración de sus reglas depende también de dicho método. Así, cuando se usa el método de un solo paso, el entrenamiento de las reglas es un simple aprendizaje (directo) de los datos del conjunto de entrenamiento, por lo que el número de reglas dependerá del tamaño de dicho conjunto. Por su parte, cuando se usa el método de gradiente descendente, el entrenamiento consiste en predefinir un número constante de reglas, cuyos parámetros se irán ajustando a partir de los datos de entrenamiento; aquí el número de reglas dependerá del número de conjuntos difusos que se elija para cada uno de sus antecedentes. Se pone más atención en este método, debido a que típicamente tiene mucho menos reglas que el método de un paso, por lo que hace un pronóstico mucho más rápido, fomentando así su uso en una aplicaciones de tiempo real.
6. **Pronóstico de la serie.** Cuando se ha entrenado el sistema, se procede a pronosticar la serie usando ahora el conjunto de prueba. De los resultados obtenidos,

se calcula el error de los datos pronosticados con respecto a los datos deseados para verificar el desempeño del sistema.

Cabe mencionar que cuando se trabaja con SLDs del mismo tipo¹, pero con métodos de diseño distintos (e.g., método de un paso y método de gradiente descendente), los primeros cuatro pasos de la metodología son comunes, la única diferencia se tiene en los dos últimos. En consecuencia, por cada tipo de SLD, primero se exponen los pasos comunes, y enseguida se especifican los pasos correspondientes a cada método de diseño.

4.2 SLDT1 *singleton*

En esta sección se examinan dos diseños de SLDT1 *singleton* que pronostican la serie de tiempo Mackey-Glass. Uno se basa en el método de diseño de un paso descrito en la sección 2.1.1, y el otro se basa en el método de gradiente descendente de la sección 2.1.2.

Para hacer más evidente la metodología anteriormente descrita, en esta sección se expone por separado cada uno de sus pasos. Sin embargo, las demás secciones de diseño omiten dicha separación para hacer la lectura más fluida y mantener la estructura del documento sencilla y fácil de seguir.

Simulación de la serie Mackey-Glass

Para obtener los datos del conjunto de entrenamiento y del conjunto de prueba que se van a usar en el diseño de los pronosticadores, se convirtió la ecuación (1.1) en una ecuación de tiempo discreto, usando el método de aproximación que se usó en [14], [17], [23]; el método de Euler [22]:

$$f(s, n) = \frac{0.2s(n - \tau)}{1 + s^{10}(n - \tau)} - 0.1s(n) \quad (4.1)$$

y

$$s(n + 1) = s(n) + h f(s, n) \quad (4.2)$$

donde el paso fue de tamaño $h = 1$. En este caso, al igual que en [6], [26], [32], se hizo $\tau = 30$.

Debido al retraso de 30 unidades de tiempo, se requieren 31 valores iniciales de $s(k)$, i.e., $s(1), s(2), \dots, s(31)$, los cuales fueron generados aleatoriamente un cierto

¹Sistemas con la misma definición en el paso 4

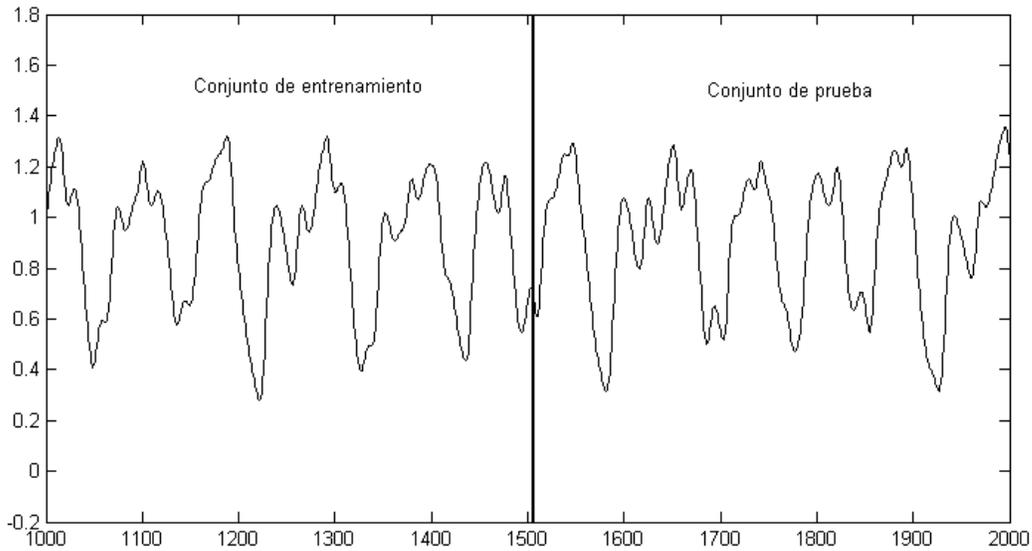


Figura 4.2: Simulación de la serie Mackey-Glass y su división en conjuntos de entrenamiento y de prueba.

número de veces hasta que la serie de tiempo obtenida se vio interesante (ver figura 4.2). Una vez obtenidos estos valores iniciales, se deben calcular los valores siguientes. Para calcular $s(32)$, por ejemplo, se tiene que:

$$s(31 + 1) = s(31) + (1) \left[\frac{0.2s(31 - 30)}{1 + s^{10}(31 - 30)} - 0.1s(31) \right]$$

$$s(32) = s(31) + \left[\frac{0.2s(1)}{1 + s^{10}(1)} - 0.1s(31) \right]$$

luego se calculan $s(32 + 1)$, $s(33 + 1)$, y así sucesivamente, hasta obtener un buen número de datos de la serie, suficientes para que de ahí salgan el conjunto de entrenamiento y el conjunto de prueba.

Establecimiento de conjuntos de entrenamiento y de prueba

Para el diseño de los sistemas, se utilizaron $N = 1000$ datos² libres de ruido, $s(k)$, con $k = 1001, 1002, \dots, 2000$, i.e.,

$$s(1001), s(1002), \dots, s(2000)$$

Note el uso de la notación $s(k)$ aquí en vez de $x(k)$ para enfatizar el hecho de que las mediciones son libres de ruido; por lo tanto, son la señal misma. Más adelante se

²1000 datos como en [6], [8], [15], [27], [32]

hará un análisis con versiones de $s(k)$ corrompidas por ruido, las cuales se denotan por $x(k)$, i.e.,

$$x(k) = s(k) + n(k), \quad k = 1001, 1002, \dots, 2000$$

Se usaron los primeros $D = 504$ datos para diseñar el SLD pronosticador; así, el conjunto de entrenamiento se formó de los siguientes datos

$$s(1001), s(1002), \dots, s(1504)$$

Los $N - D = 496$ datos restantes se usaron para probar el diseño; así, el conjunto de prueba quedó compuesto de los siguientes datos

$$s(1505), s(1506), \dots, s(2000)$$

En la figura 4.2 se puede observar la gráfica de los 1000 datos obtenidos en la simulación, así como la división entre los conjuntos de entrenamiento y de prueba.

Establecimiento del número de antecedentes por regla

El número de antecedentes por regla que se usó para hacer el pronóstico de la serie fue de $p = 4$ como en [3], [8], [15], [27], [33], i.e.,

$$s(k - 3), s(k - 2), s(k - 1), s(k)$$

A partir de los cuales se obtiene $s(k + 1)$. Como se puede apreciar, los datos tomados de $s(k)$ son consecutivos como en [17], [23], [30], [31]. En otros artículos [3], [21], los datos estaban espaciados uno de otro por un cierto valor Δ . Sin embargo, en lo que sí coincidían los artículos es en que todos realizaban un pronóstico de una etapa solamente (ver apéndice A); es decir, sólo hacían predicciones a corto plazo, ya que aunque el sistema tenga un error muy pequeño, hacer predicciones iterativas causaría un incremento en dicho error a medida que se avanza en el tiempo, hasta llegar a valores no deseados [26], [27].

En resumen, todos los sistemas de este trabajo, sin importar su tipo, hacen pronósticos de una etapa, utilizando reglas de cuatro antecedentes y un consecuente.

Definición del tipo de SLD

En esta etapa se diseñaron dos sistemas tipo 1, ambos bajo la misma definición de sistema pero con métodos distintos. Esta definición es como sigue:

- Fuzificación: *singleton*
- Funciones de membresía: Gaussianas

- Implicación: producto
- Composición: máx-producto
- Defuzificación: centro promedio

Con esta definición, y sabiendo que el número de antecedentes por regla es $p = 4$, se puede reexpresar (2.5) como

$$y(\mathbf{x}) = f_s(\mathbf{x}) = \frac{\sum_{l=1}^M \bar{y}^l \prod_{k=1}^4 \exp \left[-\frac{1}{2} \left(\frac{x_k - m_{F_k^l}}{\sigma_{F_k^l}} \right)^2 \right]}{\sum_{l=1}^M \prod_{k=1}^4 \exp \left[-\frac{1}{2} \left(\frac{x_k - m_{F_k^l}}{\sigma_{F_k^l}} \right)^2 \right]} \quad (4.3)$$

donde l es el índice de la regla. El número de reglas M , depende de los métodos de diseño, por lo que para la definición completa del sistema es necesario primero especificar el método. De manera similar, el entrenamiento del sistema y el pronóstico de la serie también dependen de dicha especificación. Es por ello que las secciones siguientes se basan en el método seleccionado, y presentan los tres pasos finales de la metodología que le corresponden a cada método.

4.2.1 Método de un paso

En este método, el entrenamiento del sistema consiste en obtener las reglas directamente de los datos del conjunto de entrenamiento, como se explica a continuación.

Entrenamiento del SLD

Los datos establecen los centros de los conjuntos difusos de los antecedentes y consecuentes de las reglas. Las desviaciones estándar de las funciones de membresía Gaussianas se preestablecieron en este caso a $\sigma = 0.1$.

Hay $D - p = 504 - 4 = 500$ reglas que se pueden extraer de los 500 pares de entrenamiento, $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(D-p)}$. Cada regla $R^{(l)}$ se obtiene de su respectivo par de entrenamiento $x^{(l)}$, y se puede expresar como

$$R^l: \text{SI } x_1 \text{ es } F_1^l \text{ y } x_2 \text{ es } F_2^l \text{ y } \dots \text{ y } x_4 \text{ es } F_4^l, \text{ ENTONCES } y \text{ es } G^l$$

donde $l = 1, 2, \dots, 500$. F_1^l es un conjunto difuso cuya función de membresía está centrada en $x(1)$, F_2^l es un conjunto difuso cuya función de membresía está centrada en $x(2)$, \dots , F_p^l es un conjunto difuso cuya función de membresía está centrada en

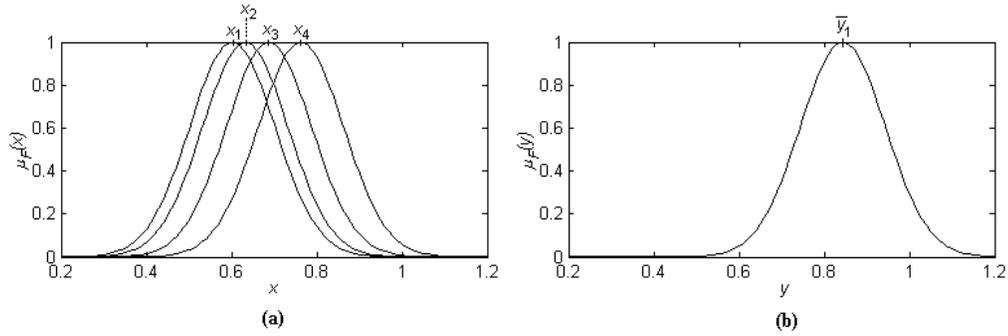


Figura 4.3: Conjuntos difusos obtenidos de $\mathbf{x}^{(1)}$ para la regla $R^{(1)}$. (a) Antecedentes de $R^{(1)}$. (b) Consecuente de $R^{(1)}$.

$x(p)$ y G^l es un conjunto difuso cuya función de membresía está centrada en $x(p+1)$, ver sección 2.1.1.

Por ejemplo, para el par de entrenamiento $\mathbf{x}^{(1)} = [0.5940 \ 0.6019 \ 0.6323 \ 0.6878 \ 0.7628]^T$ se tienen los conjuntos difusos de los antecedentes mostrados en la figura 4.3 (a), y el conjunto difuso del consecuente se muestra en la figura 4.3 (b).

Pronóstico de la serie

Una vez que se tienen las 500 reglas, se hace el pronóstico de la serie con los 496 pares de datos del conjunto de prueba. En la figura 4.4 se muestra el pronóstico obtenido con este sistema junto con la gráfica de la serie original.

Para evaluar el desempeño de cada sistema se utilizó el error cuadrático medio, el cual se representa normalmente por sus siglas en inglés RMSE – *Root Mean-Squared Error*. Este error se calcula con la fórmula

$$RMSE_s = \sqrt{\frac{1}{496} \sum_{k=1504}^{1999} [s(k+1) - f_s(\mathbf{x}^{(k)})]^2} \quad (4.4)$$

donde $s^{(k)} = [s(k-3), s(k-2), s(k-1), s(k)]^T$ y el subíndice “s” en RMSE indica que es un diseño singleton. El error obtenido al hacer el pronóstico con el método de un paso fue de

$$RMSE_s(UP) = 0.0485$$

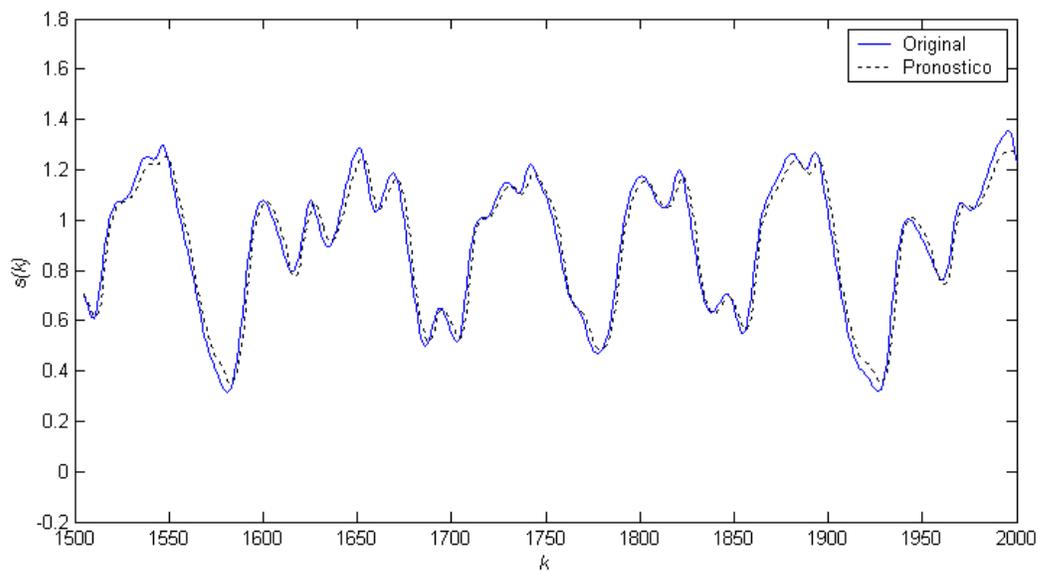


Figura 4.4: Gráfica comparativa de la salida del SLDT1 *singleton* con el método de un paso vs. la serie original.

4.2.2 Método de gradiente descendente

En este método, el entrenamiento del sistema consiste en ajustar los parámetros de las reglas usando los datos del conjunto de entrenamiento como se explica a continuación.

Entrenamiento del SLD

Debido a que al inicio no se tienen reglas que ajustar, es necesario seguir un proceso de elaboración de reglas, el cual exige la preespecificación del número de conjuntos difusos que se van a utilizar en cada variable de entrada. En este caso se usaron solamente dos conjuntos difusos por cada uno de los 4 antecedentes, por lo tanto se obtuvieron $2^4 = 16$ reglas. Cada regla tiene 8 parámetros en las funciones de membresía de los antecedentes (2 por cada función de membresía Gaussiana, m y σ) y 1 parámetro en el consecuente (\bar{y}).

La ubicación inicial de cada Gaussiana se basó en la media m_s y desviación estándar σ_s de los 504 datos de entrenamiento. Específicamente, se inicializó la media de cada una de las dos Gaussianas de cada antecedente como $m_s - 2\sigma_s = 0.3086$ y $m_s + 2\sigma_s = 1.4809$, respectivamente; y sus desviaciones estándar como $2\sigma_s = 0.5861$. El centro de cada consecuente \bar{y}^i ($i = 1, \dots, 16$) se inicializó con un número aleatorio del intervalo $[0,1]$. El valor del parámetro de aprendizaje fue de $\alpha = 0.3$.

Tabla 4.1: Valores iniciales para el centro de las funciones de membresía Gaussianas de los antecedentes y el centro del conjunto consecuente. Note que $m_s - 2\sigma_s = 0.3086$ y $m_s + 2\sigma_s = 1.4809$.

# Regla	Valores iniciales de los centros					$\bar{y}_{\text{inicial}}^i$
1	0.3086	0.3086	0.3086	0.3086	0.3086	0.0620
2	0.3086	0.3086	0.3086	1.4809	1.4809	0.6082
3	0.3086	0.3086	1.4809	0.3086	0.3086	0.5390
4	0.3086	0.3086	1.4809	1.4809	1.4809	0.9515
5	0.3086	1.4809	0.3086	0.3086	0.3086	0.6065
6	0.3086	1.4809	0.3086	1.4809	1.4809	0.1518
7	0.3086	1.4809	1.4809	0.3086	0.3086	0.2511
8	0.3086	1.4809	1.4809	1.4809	1.4809	0.6249
9	1.4809	0.3086	0.3086	0.3086	0.3086	0.3346
10	1.4809	0.3086	0.3086	1.4809	1.4809	0.3744
11	1.4809	0.3086	1.4809	0.3086	0.3086	0.4640
12	1.4809	0.3086	1.4809	1.4809	1.4809	0.1918
13	1.4809	1.4809	0.3086	0.3086	0.3086	0.0015
14	1.4809	1.4809	0.3086	1.4809	1.4809	0.7000
15	1.4809	1.4809	1.4809	0.3086	0.3086	0.5217
16	1.4809	1.4809	1.4809	1.4809	1.4809	0.3614

La Tabla 4.1 muestra los valores iniciales de los centros de las funciones de membresía Gaussianas así como los valores iniciales de todas las \bar{y}^i . Los valores iniciales de las desviaciones estándar de estas funciones de membresía Gaussianas se preestablecieron con el mismo valor de 0.5861. Los valores finales de estos parámetros después de 6 épocas de ajuste se muestran en las Tablas 4.2 y 4.3.

Para hacer más ilustrativo el ajuste que se realiza en este método, se presentan las gráficas de los conjuntos difusos correspondientes a la primera regla, i.e., R^1 . Los valores iniciales de esta regla son

$$\mathbf{R}_{\text{inicial}}^1 = [0.3086 \ 0.3086 \ 0.3086 \ 0.3086 \ 0.0620]$$

por lo tanto, los conjuntos difusos de los antecedentes son iguales, como se puede observar en la figura 4.5 (a). En la figura 4.5 (b) se presenta el conjunto difuso del consecuente, cuya media corresponde al último elemento de $\mathbf{R}_{\text{inicial}}^1$, i.e., 0.0620.

Al finalizar las 6 épocas, las medias de los conjuntos difusos de esta regla quedan como

Tabla 4.2: Valores finales para el centro de las funciones de membresía Gaussianas de los antecedentes y el centro del conjunto consecuente, después de 6 épocas de ajuste.

# Regla	Valores finales de los centros					\bar{y}_{final}^k
1	0.3796	0.3319	0.2578	0.0957	0.0986	
2	0.2484	0.2630	0.2837	1.4489	0.8882	
3	0.2771	0.2780	1.5244	0.2756	0.6482	
4	0.5321	0.5622	1.3077	1.4436	1.7395	
5	0.2979	1.5406	0.3003	0.2976	0.6225	
6	0.2221	1.5548	0.1947	1.5326	0.2390	
7	0.2799	1.5125	1.5026	0.2470	0.3064	
8	0.2755	1.4554	1.4656	1.4820	1.2281	
9	1.4875	0.3397	0.3186	0.2799	0.2446	
10	1.5368	0.2424	0.2282	1.5070	0.4717	
11	1.5082	0.3040	1.4922	0.2751	0.5088	
12	1.5443	0.1557	1.5348	1.5303	0.3051	
13	1.4279	1.4251	0.3968	0.3401	-0.0755	
14	1.4458	1.4590	0.3457	1.5193	1.2796	
15	1.4903	1.4856	1.4746	0.3002	0.6679	
16	1.6465	1.6368	1.6359	1.6442	1.3455	

Tabla 4.3: Valores finales para la desviación estándar de las funciones de membresía Gaussianas de los antecedentes, después de 6 épocas de ajuste.

# Regla	Valores finales de las σ				
1	0.6307	0.5998	0.5406	0.3032	
2	0.5380	0.5454	0.5510	0.6195	
3	0.5677	0.5627	0.5204	0.5295	
4	0.8227	0.8503	0.7401	0.5003	
5	0.5963	0.4852	0.5806	0.5591	
6	0.4693	0.4941	0.4010	0.5459	
7	0.5576	0.5475	0.5686	0.4612	
8	0.5135	0.6241	0.6025	0.5664	
9	0.5413	0.6245	0.5896	0.5159	
10	0.5142	0.4922	0.4542	0.5705	
11	0.5447	0.5861	0.5671	0.4986	
12	0.5240	0.2840	0.5421	0.5506	
13	0.6321	0.6477	0.6716	0.5374	
14	0.6300	0.6119	0.6290	0.5104	
15	0.5693	0.5708	0.5810	0.4891	
16	0.4733	0.4853	0.4827	0.4637	

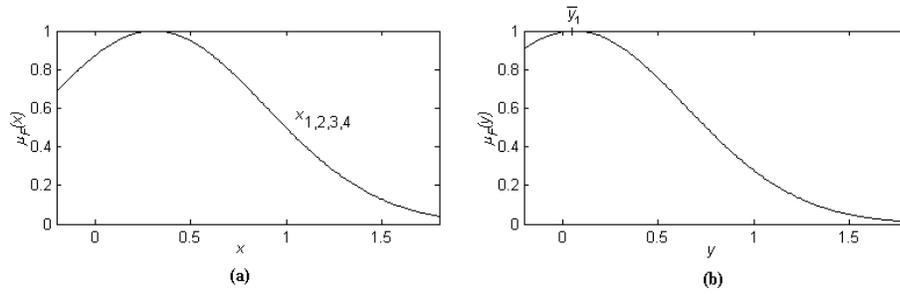


Figura 4.5: Conjuntos difusos de la regla $\mathbf{R}_{\text{inicial}}^1$ con desviación estándar de 0.5861. (a) Antecedentes. (b) Consecuente.

$$\mathbf{R}_{\text{final}}^1 = [0.3796 \ 0.3319 \ 0.2578 \ 0.0957 \ 0.0986]$$

y sus desviaciones estándar respectivas son

$$\sigma_{\text{final}}^1 = [0.6307 \ 0.5998 \ 0.5406 \ 0.3032]$$

como se observa en la figura 4.6. Note que la desviación estándar del consecuente no se incluye aquí, pues en este caso es irrelevante debido al tipo de defuzificador utilizado. Este defuzificador sólo toma en cuenta el valor donde la función de membresía es mayor (\bar{y}^l), sin tomar en cuenta su dispersión. Sin embargo, para efectos ilustrativos, aquí se considera como 0.5861.

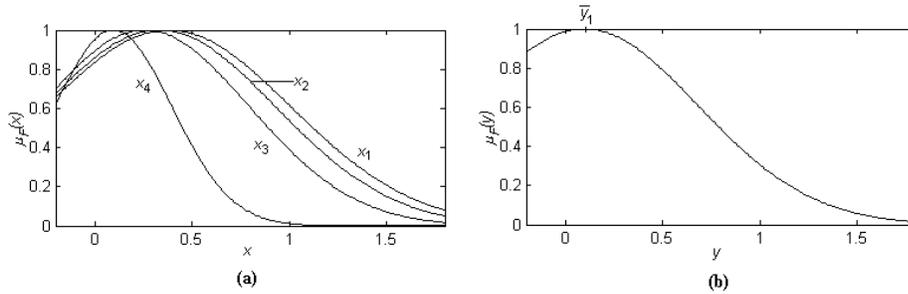


Figura 4.6: Conjuntos difusos de la regla $\mathbf{R}_{\text{final}}^1$ con sus desviaciones estándar finales correspondientes. (a) Antecedentes. (b) Consecuente (conserva su σ_{inicial}).

Pronóstico de la serie

Una vez ajustados los parámetros de las 16 reglas, se hace el pronóstico de la serie con los 496 pares del conjunto de prueba. Para calcular el error del pronóstico,

se utilizó nuevamente la ecuación (4.4). Los errores en cada una de las 6 épocas de ajuste fueron:

$$RMSE_s(GD) = \{0.0701 \quad 0.0523 \quad 0.0393 \quad 0.0345 \quad 0.0330 \quad 0.0322\}$$

Estos valores fueron obtenidos haciendo el ajuste para una época; entonces se probaba el diseño con los datos de prueba, $s(1505), s(1506), \dots, s(2000)$. Luego, se hacía el ajuste para una segunda época y otra vez se comparaba con los datos de prueba; y así sucesivamente, por 6 épocas. A partir de estos valores obtenidos se obtuvo la gráfica de la figura 4.7, donde se observa una reducción sustancial en el RMSE de la primera a la sexta época. Note que alrededor de la quinta época el error se empieza a estabilizar; por lo que es razonable detener el algoritmo en la sexta época.

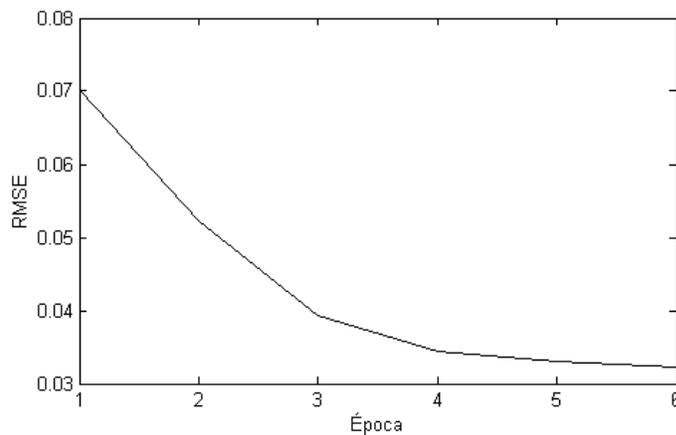


Figura 4.7: Gráfica del error por época del SLD *singleton* diseñado con el método de gradiente descendente.

4.2.3 Pronóstico con ruido

Los sistemas en las secciones 4.2.1 y 4.2.2 se diseñaron utilizando mediciones perfectas de la serie $s(k)$, en espera de que trabajen bajo el mismo ambiente libre de ruido. Sin embargo, si el ambiente cambia y deben operar ahora con mediciones con ruido, los sistemas tienen un desempeño distinto. Para probar qué tan robustos son estos sistemas, se realizó un pronóstico con datos de prueba que tienen ruido, i.e.,

$$x(k) = s(k) + n(k), \quad k = 1505, 1506, \dots, 2000$$

donde $n(k)$ representa una señal de ruido adicional a la señal de la serie, como se muestra en la figura 4.8. Note el uso de $x(k)$ en el eje de las ordenadas para enfatizar

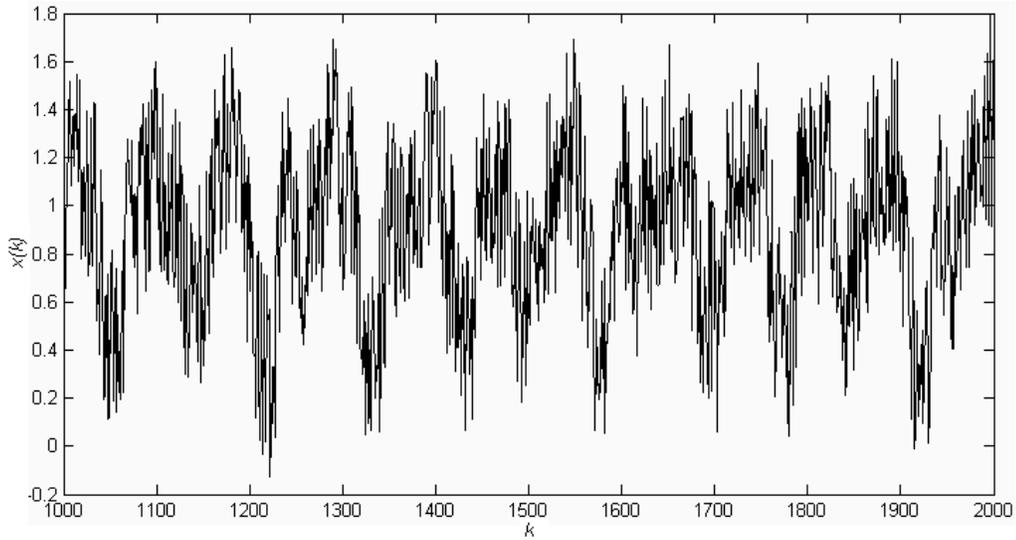


Figura 4.8: Una simulación de la serie de tiempo Mackey-Glass con datos de una señal de ruido adicional, $x(1001), x(1002), \dots, x(2000)$.

el hecho de que los datos son corrompidos por ruido; además, se grafica con la misma escala que la figura 4.2 para que la comparación entre ambas gráficas sea más sencilla. Note también que aquí se grafican 1000 datos, i.e., $k = 1001, 1002, \dots, 2000$; pero en realidad los que se utilizan para hacer el pronóstico son los del conjunto de prueba.

Se hicieron 50 simulaciones con ruido, de manera que se tuvieran 50 gráficas similares a la de la figura 4.8, las cuales se conservaron para aplicarlas a los demás diseños y que todos tuvieran exactamente las mismas señales de ruido. El pronóstico obtenido con el método de un paso se muestra en la figura 4.9, la cual es una superposición de 50 gráficas. Los RMSEs obtenidos en cada iteración se muestran gráficamente en la figura 4.11.

El pronóstico obtenido con el método de gradiente descendente se muestran en la figura 4.10. Los RMSEs que se tienen en cada iteración se presentan gráficamente en la figura 4.11, donde se observa que para todas las simulaciones fueron mayores los errores de este método con respecto al de un paso, lo que indica que el pronosticador diseñado con el método de un paso es más robusto al ruido inesperado que el de gradiente descendente. Esto también se puede deducir de las gráficas de las figuras 4.9 y 4.10, ya que la desviación estándar (i.e., el grosor de las gráficas superpuestas) es visiblemente menor en el método de un paso; así, el hecho de tener más reglas lo hace menos susceptible a las mediciones ruidosas.

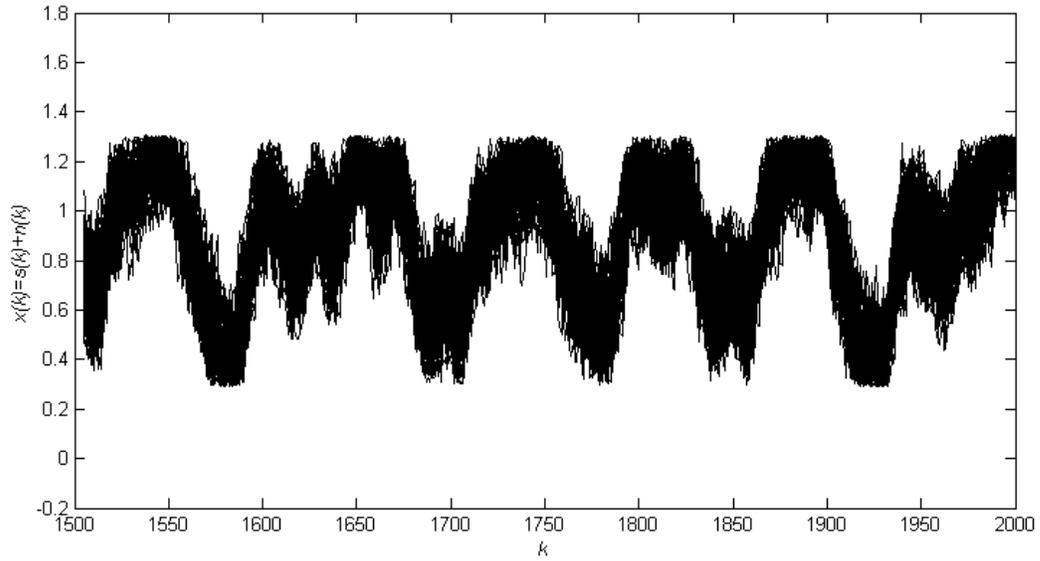


Figura 4.9: Salidas superpuestas de 50 simulaciones del SLDT1 *singleton* con el método de un paso, cuando se tiene un ruido adicional en la serie.

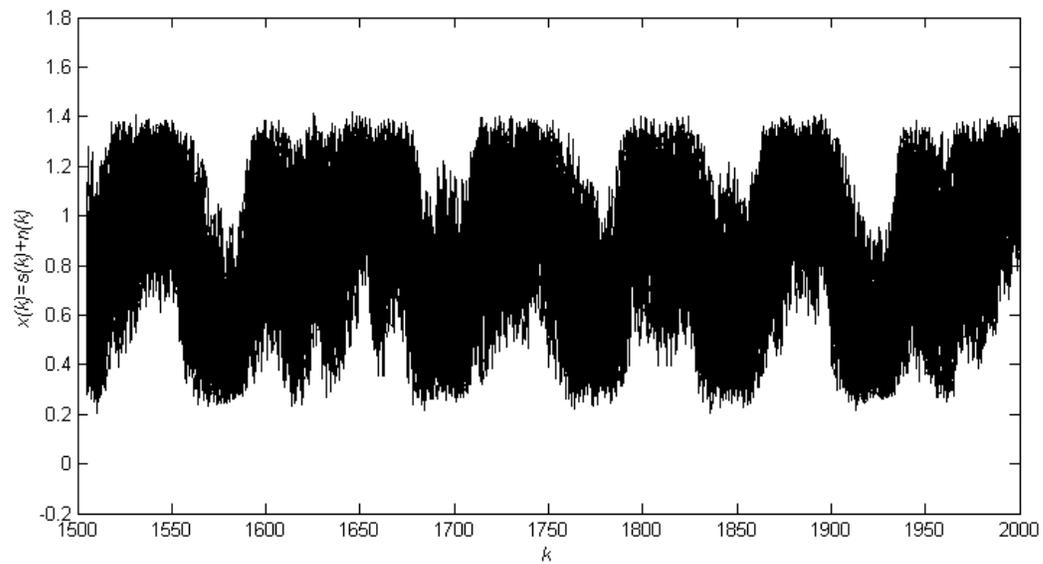


Figura 4.10: Salidas superpuestas de 50 simulaciones del SLDT1 *singleton* con el método de un paso, cuando se tiene un ruido adicional en la serie.

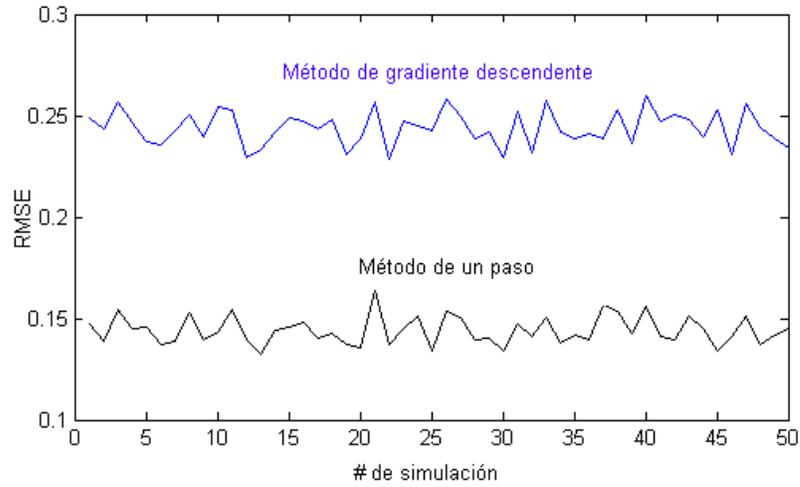


Figura 4.11: Gráfica comparativa de los errores de los SLD, obtenidos en cada simulación.

4.3 SLDT1 no *singleton*

En esta sección se presentan dos diseños de SLDT1 no *singleton* que pronostican la serie de tiempo Mackey-Glass. Uno se basa en el método de diseño de un paso descrito en la sección 2.2.1, y el otro se basa en el método de gradiente descendente de la sección 2.2.2. Ambos diseños se basan en los siguientes 1000 pares de entrada ruidosos:

$$x(1001), x(1002), \dots, x(2000)$$

Los primeros 504 datos con ruido, $x(1001), x(1002), \dots, x(1504)$, se usaron para el entrenamiento de los sistemas; y los 496 datos restantes, $x(1505), x(1506), \dots, x(2000)$, se usaron para probar los diseños.

Para la obtención de los datos con ruido, se tomó la serie de tiempo Mackey-Glass sin ruido de la figura 4.2. Se consideró que esta serie, $s(k)$, es corrompida por un ruido adicional estacionario uniformemente distribuido [19], $n(k)$,

$$x(k) = s(k) + n(k) \quad k = 1001, 1002, \dots, 2000$$

donde la razón de señal a ruido (Signal-to-Noise Ratio) es $SNR = 0\text{dB}$. La figura 4.8 muestra una simulación de $x(1001), x(1002), \dots, x(2000)$. La diferencia entre los diseños de esta sección y los de la sección 4.2 es que aquí el entrenamiento se hizo

con datos con ruido, mientras que en la 4.2 se usaron datos de entrenamiento perfectos.

Al igual que en la sección 4.2, se usaron cuatro antecedentes para hacer el pronóstico; específicamente, $x(k-3)$, $x(k-2)$, $x(k-1)$ y $x(k)$, para predecir $x(k+1)$. Las funciones de membresía usadas en ambos diseños fueron Gaussianas. El desempeño de cada diseño no *singleton* se evaluó usando la siguiente fórmula

$$RMSE_{ns} = \sqrt{\frac{1}{496} \sum_{k=1504}^{1999} [s(k+1) - f_{ns}(\mathbf{x}^{(k)})]^2} \quad (4.5)$$

donde $\mathbf{x}^{(k)} = [x(k-3), x(k-2), x(k-1), x(k)]^T$

4.3.1 Método de un paso

En este método se usaron datos de entrenamiento con ruido para la elaboración de 500 reglas, i.e., $x(1001), x(1002), \dots, x(1504)$. En cada una de las reglas, las funciones de membresía Gaussianas de los antecedentes y de los consecuentes tenían sus centros en los valores de las mediciones ruidosas de $x(k)$, y las desviaciones estándar se preestablecieron a 0.1. Para el cálculo de la desviación estándar de las Gaussianas que modelan las mediciones de entrada ruidosas, se utilizó el hecho de que la razón de señal a ruido es $SNR = 10 \log_{10} (\sigma_s^2 / \sigma_n^2)$, de donde se obtuvo que $\sigma_n = \sigma_s / 10^{SNR/20}$, y se estableció $\sigma_X = \sigma_n$. Con σ_X establecida, se completó la definición del SLD no *singleton* $f_{ns}(\mathbf{x})$ y se procedió a calcular el $RMSE_{ns}$ del sistema, aplicando (4.5) a los datos de prueba $x(1505), x(1506), \dots, x(2000)$. Todo este proceso se repitió 50 veces usando 50 conjuntos independientes de 1000 datos, por lo que al final se tenían 50 valores de $RMSE_{ns}(UP)$. El promedio y la desviación estándar de los 50 $RMSE_{ns}$ son

$$\overline{RMSE}_{ns}(UP) = 0.1371 \quad \text{y} \quad \sigma_{RMSE_{ns}(UP)} = 0.0082$$

Las reglas generadas con el método de un paso en el diseño del SLD no *singleton* son exactamente iguales a las generadas en el *singleton* en cada una de las 50 simulaciones; sin embargo, las reglas en el diseño no *singleton* incluyen el efecto de las mediciones ruidosas por medio de su prefiltrado hecho en la fuzificación, mientras que el *singleton* no hace tal prefiltrado. El $RMSE_s$ de cada simulación en el diseño *singleton* se calculó usando (4.4). El promedio y la desviación estándar de los 50 $RMSE_s$ son

$$\overline{RMSE}_s(UP) = 0.1954 \quad \text{y} \quad \sigma_{RMSE_s(UP)} = 0.0075$$

Observe que el error obtenido en el diseño *singleton* fue casi 30% mayor que el obtenido en el diseño no *singleton*. Por lo tanto, el tomar en cuenta el ruido usando fuzificación no *singleton* puede resultar en una mejora substancial en el desempeño del SLD, inclusive para un SLD con diseño de un paso.

La figura 4.12 ilustra más fácilmente la mejora del diseño del SLD no *singleton* sobre el *singleton*. Ahí se muestran los pronósticos de cada diseño obtenidos en las 50 simulaciones (cada inciso es una sobreposición de 50 gráficas). Se observa claramente que el diseño no *singleton* tiene menor desviación estándar que el *singleton*, i.e., el ancho de la gráficas sobrepuestas es menor en el sistema con diseño no *singleton*.

4.3.2 Método de gradiente descendente

En esta sección se compara un SLDT1 no *singleton* con un SLDT1 *singleton* cuando usan el método de diseño de gradiente descendente. Para el entrenamiento del SLD no *singleton* se usó un diseño totalmente independiente (ver sección 2.2), en el cual todos los parámetros del sistema son ajustados y sus valores iniciales no dependen de diseños anteriores.

Al igual que en la sección 4.2.2, se usan sólo dos conjuntos difusos para cada uno de los cuatro antecedentes, por lo que hay sólo 16 reglas. Cada regla está caracterizada por ocho parámetros en las funciones de membresía de los antecedentes (la media y la desviación estándar por cada una de las cuatro funciones de membresía Gaussianas) y por un parámetro en el consecuente, \bar{y} . La ubicación inicial de cada función de membresía Gaussiana de los antecedentes se basó en la media m_x y la desviación estándar σ_x de los 504 datos de entrenamiento ruidosos, $x(1001), x(1002), \dots, x(1504)$. Específicamente, se inicializó la media de cada una de las dos Gaussianas de cada antecedente como $m_s - 2\sigma_x$ o $m_s + 2\sigma_x$, respectivamente; y sus desviaciones estándar como $2\sigma_x$. El centro de cada consecuente \bar{y}^i ($i = 1, \dots, 16$) se inicializó con un número aleatorio del intervalo $[0,1]$.

En este método, como en la sección 4.3.1, se modelaron las cuatro mediciones de entrada ruidosas usando funciones de membresía Gaussianas. Para ajustar la desviación estándar de estas funciones de membresía se tienen dos opciones: (1) usar una desviación estándar diferente para cada una de las cuatro funciones de membresía, o (2) usar la misma desviación estándar para todas ellas. En este trabajo se experimentó con ambas opciones y se obtuvieron resultados muy similares; pero como el ruido adicional utilizado es estacionario (i.e., su desviación estándar no varía en el tiempo), resulta más natural usar la misma desviación estándar para cada función de membresía de las mediciones de la entrada, por lo que sólo se presentan los resultados

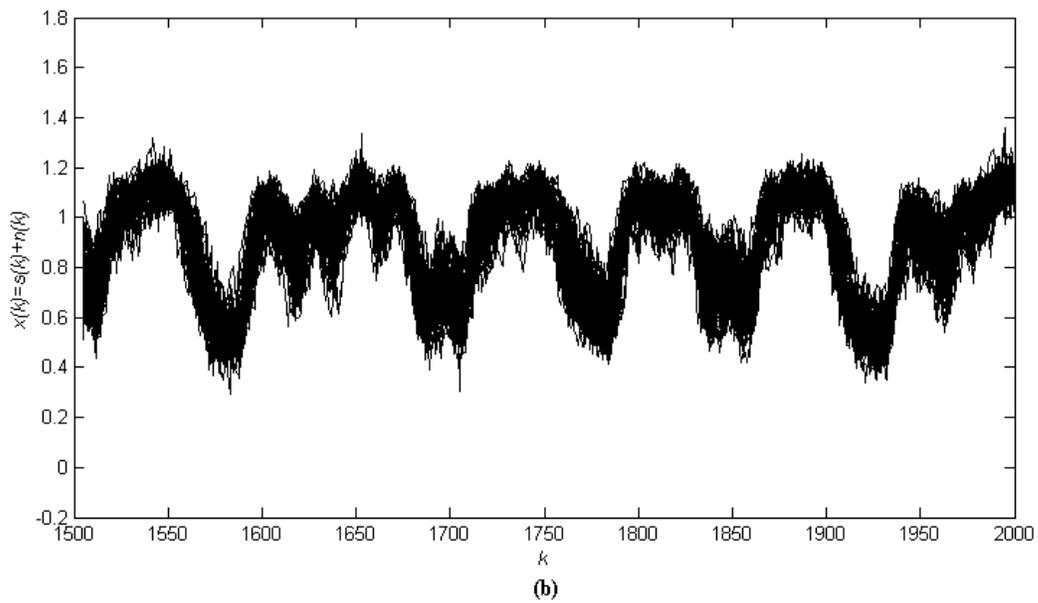
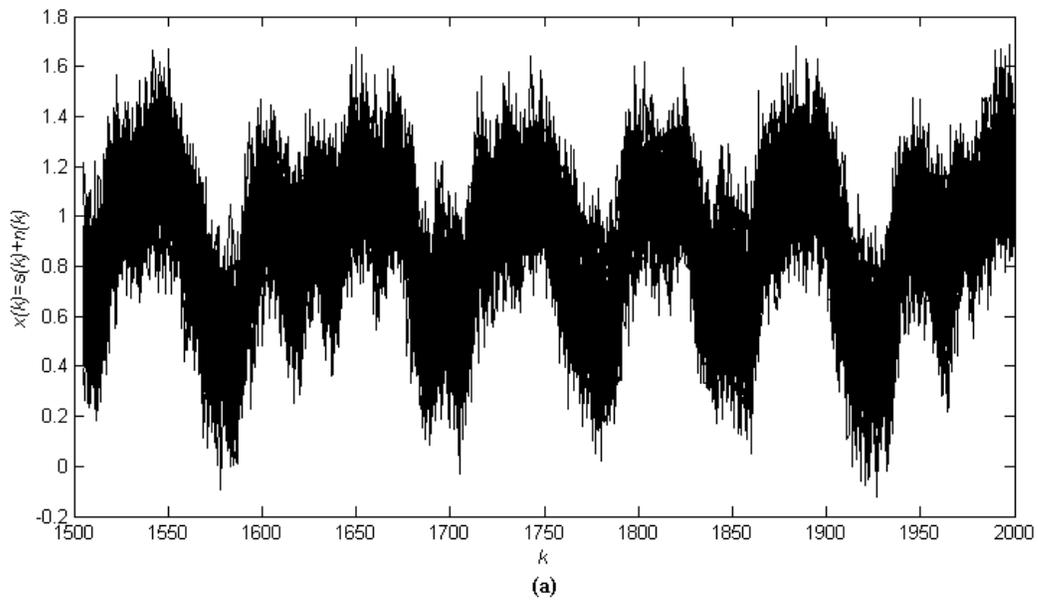


Figura 4.12: Salidas de los SLDT1 usando el método de diseño de un paso y con fuzificación: (a) *singleton*, (b) no *singleton*. En ambos casos hay 50 simulaciones y los datos del 1504 al 2000 están corrompidos por un ruido adicional uniformemente distribuido con $SNR = 0\text{dB}$

obtenidos con esa opción.

Cada SLD fue ajustado usando un parámetro de aprendizaje de $\alpha = 0.3$. El entrenamiento y la prueba del sistema se realizaron durante 6 épocas. Después de cada época se usaron los datos de prueba para evaluar el desempeño de cada SLD, calculando el $RMSE_s(GD)$ con (4.4) y el $RMSE_{ns}(GD)$ con (4.5). Todo este proceso se repitió 50 veces usando 50 conjuntos independientes de 1000 datos, por lo que al final se tenían 50 valores de $RMSE_s(GD)$ y 50 de $RMSE_{ns}(GD)$. Los promedios y las desviaciones estándar de $RMSE_s(GD)$ y $RMSE_{ns}(GD)$ se muestran gráficamente en la figura 4.13. Observe que el SLDT1 no *singleton* siempre supera en desempeño al SLDT1 *singleton*, aunque no por mucho, si se examina cuidadosamente la escala de las gráficas.

Comparando el $RMSE$ final del diseño de gradiente descendente con el $RMSE$ del diseño de un paso, (sección 4.3.1), se puede ver que:

- El SLD *singleton* de 16 reglas diseñado con el método de gradiente descendente supera al desempeño del SLD *singleton* de 500 reglas diseñado con el método de un paso; i.e., $\overline{RMSE}_s(GD) = 0.1566$ y $\overline{RMSE}_s(UP) = 0.1954$.
- El SLD no *singleton* de 16 reglas diseñado con el método de gradiente descendente no supera al desempeño del SLD no *singleton* de 500 reglas diseñado con el método de un paso; i.e., $\overline{RMSE}_{ns}(GD) = 0.1553$ y $\overline{RMSE}_{ns}(UP) = 0.1371$.
- El diseño con menor error considerado hasta ahora es el del SLD no *singleton* de 500 reglas diseñado con el método de un paso; i.e., $RMSE_{ns}(UP) = 0.1371$.

La mejora en el desempeño usando el SLDT1 no *singleton* diseñado con el método de gradiente descendente no es tan considerable debido a que no se ha tomado en cuenta toda la incertidumbre como se debería; i.e., los datos de entrenamiento son ruidosos, pero no hay manera de tomar en cuenta esto en las funciones de membresía de los antecedentes de un SLDT1. Lo más que se puede hacer usando un SLDT1 no *singleton* es tomar en cuenta el ruido en las mediciones de entrada que excitan las reglas del sistema, mediante el prefiltrado que realiza la fuzificación no *singleton*.

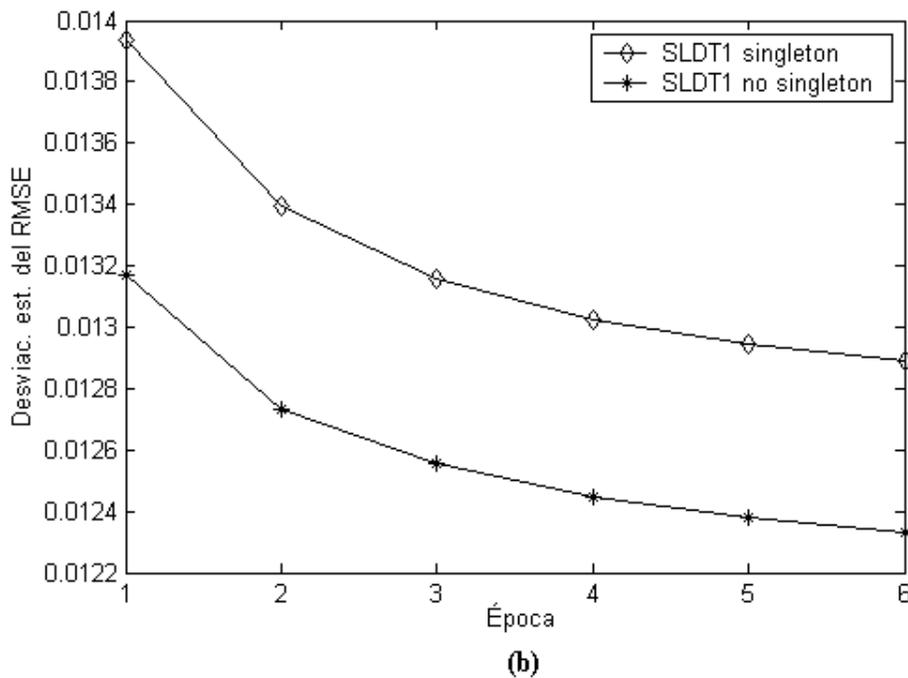
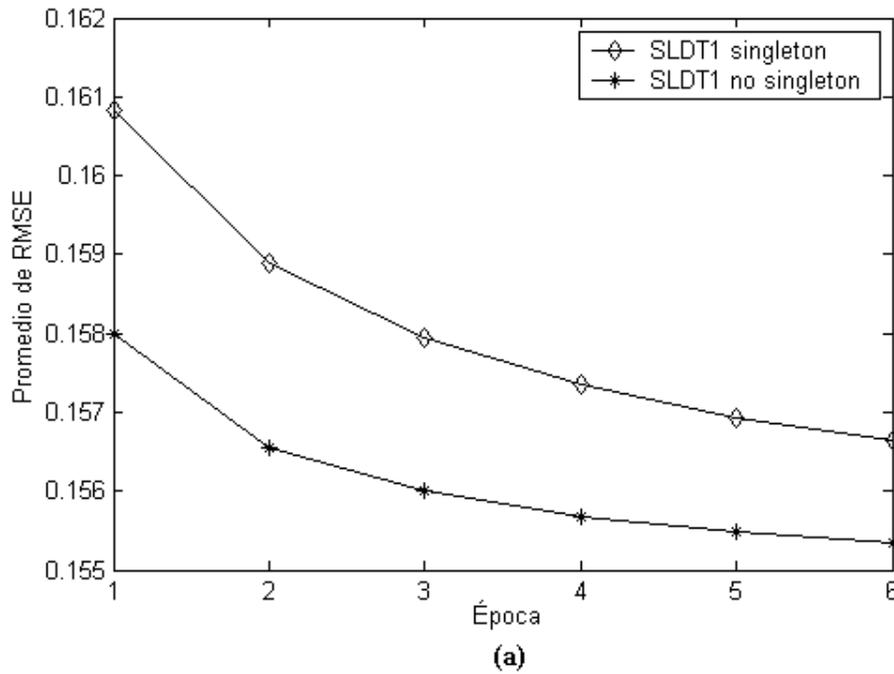


Figura 4.13: Promedios y desviaciones estándar de $RMSE_s(GD)$ y $RMSE_{ns}(GD)$ obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación. (a) Promedios. (b) Desviaciones estándar.

4.4 SLDT2 *singleton*

En esta sección se examinan dos diseños de SLDT2 *singleton* de intervalo que pronostican la serie de tiempo Mackey-Glass. Uno se basa en el método de diseño de un paso descrito en la sección 3.1.1, y el otro se basa en el método de gradiente descendente de la sección 3.1.2. Ambos sistemas utilizaron los mismos criterios de diseño que la sección 4.3; i.e., los mismos conjuntos de entrenamiento y de prueba, el mismo número de antecedentes por regla ($p = 4$), y usaron también funciones de membresía de Gaussianas, sólo que en esta ocasión eran funciones de membresía *primarias* Gaussianas con media incierta definidas por la ecuación (3.19).

El desempeño de cada diseño se evaluó nuevamente usando las siguientes fórmulas:

$$RMSE_{s1} = \sqrt{\frac{1}{496} \sum_{k=1504}^{1999} [s(k+1) - f_{s1}(\mathbf{x}^{(k)})]^2} \quad (4.6)$$

$$RMSE_{ns1} = \sqrt{\frac{1}{496} \sum_{k=1504}^{1999} [s(k+1) - f_{ns1}(\mathbf{x}^{(k)})]^2} \quad (4.7)$$

y

$$RMSE_{s2} = \sqrt{\frac{1}{496} \sum_{k=1504}^{1999} [s(k+1) - f_{s2}(\mathbf{x}^{(k)})]^2} \quad (4.8)$$

Note que las ecuaciones (4.6) y (4.7) son las mismas que (4.4) y (4.5), respectivamente, sólo que fueron reexpresadas para resaltar el hecho de que son *RMSEs* de sistemas de tipo 1, y así poder diferenciarlos de los de tipo 2.

4.4.1 Método de un paso

En este método, como en la sección 3.1.1, se preestablecieron las desviaciones estándar de las funciones de membresía Gaussianas a $\sigma = 0.1$. Sin embargo, los centros de estas funciones de membresía, ahora eran funciones de membresía de intervalo (ver sección B.1.5), por lo que se basaban en un rango de incertidumbre, no en un solo valor. Siendo así, los límites izquierdo y derecho de la media de cada función de membresía primaria Gaussianas (tanto antecedente como consecuente) se establecían en $x(k) - 0.1/4$ y $x(k) + 0.1/4$, respectivamente.

Todo el proceso de generación de 500 reglas (i.e., entrenamiento) y de prueba del sistema, se repitió 50 veces usando 50 conjuntos independientes de 1000 datos, por lo que al final se tuvieron 50 valores de $RMSE_{s1}(UP)$, $RMSE_{ns1}(UP)$ y $RMSE_{s2}(UP)$.

Los promedios y las desviaciones estándar de estos $RMSE$ s se muestran en la tabla 4.4. Observe que:

Tabla 4.4: Promedios y desviaciones estándar de $RMSE_{s1}(UP)$, $RMSE_{ns1}(UP)$ y $RMSE_{s2}(UP)$, obtenidos en 50 simulaciones de diseño.

Sistema	\overline{RMSE}	σ_{RMSE}
SLDT1 <i>singleton</i>	0.1954	0.0082
SLDT2 <i>singleton</i>	0.1784	0.0077
SLDT1 no <i>singleton</i>	0.1371	0.0075

- Hay una mejora en el desempeño (tanto en la media como en la desviación estándar del $RMSE$) del SLDT2 *singleton* de intervalo sobre su contraparte de tipo 1.
- El decremento en el promedio del $RMSE$ en los sistemas con fuzificador *singleton*, de 0.1954 a 0.1784, representa una reducción del 8.7%.
- El decremento en la desviación estándar del $RMSE$ en los sistemas con fuzificador *singleton*, de 0.0082 a 0.0077, representa una reducción del 6%.
- El SLDT1 no *singleton* sigue siendo el de mejor desempeño; i.e., $RMSE_{ns}(UP) = 0.1371$.

La mejora en el desempeño usando el SLD tipo 2 *singleton* sobre el de tipo 1 se debe a que las reglas con conjuntos difusos tipo 2 ya toman en cuenta el ruido en los datos de entrenamiento; sin embargo, no toman en cuenta el ruido de los datos de prueba, por lo que se ve superado por el SLDT1 no *singleton*.

4.4.2 Método de gradiente descendente

En esta sección se presenta el diseño de un SLDT2 *singleton* de intervalo con el método de gradiente descendente y se comparan sus resultados con los obtenidos en los SLDT1s no *singleton* y *singleton*, diseñados también con el método de gradiente descendente. Para el entrenamiento del SLDT2 *singleton* se usó un diseño totalmente independiente (ver sección 3.2), en el cual los valores iniciales de los parámetros del sistema no dependen de diseños anteriores.

Al igual que en las secciones 4.2.2 y 4.3.2, se usaron sólo dos conjuntos difusos para cada uno de los cuatro antecedentes; por lo tanto, se tuvo un total de 16 reglas. Cada regla está caracterizada por 12 parámetros en las funciones de membresía de los antecedentes (los límites izquierdo y derecho de la media, y la desviación estándar por cada una de las cuatro funciones de membresía Gaussianas) y por dos parámetros en el consecuente (los puntos izquierdo y derecho del centroide en el conjunto difuso tipo 2 del consecuente). Por lo tanto, el total de parámetros ajustados en este diseño es de 14 parámetros por regla \times 16 reglas = 224 parámetros.

Los intervalos de incertidumbre para las medias de los dos conjuntos difusos de cada uno de los cuatro antecedentes se inicializaron a

$$[m_x - 2\sigma_x - 0.25\sigma_n, m_x - 2\sigma_x + 0.25\sigma_n]$$

y

$$[m_x + 2\sigma_x - 0.25\sigma_n, m_x + 2\sigma_x + 0.25\sigma_n]$$

respectivamente, donde m_x y σ_x son el promedio y la desviación estándar de los 504 datos de entrenamiento $x(1001), x(1002), \dots, x(1504)$; y σ_n es la desviación estándar del ruido adicional calculado en la sección 4.3.1, (en este caso $\sigma_n = \sigma_s$, ya que $SNR = 0\text{dB}$). Por su parte, los puntos izquierdo y derecho del centroide en el conjunto difuso tipo 2 del consecuente se inicializaron a

$$y_l^i = \bar{y}^i - \sigma_n \quad \text{y} \quad y_r^i = \bar{y}^i + \sigma_n$$

respectivamente, donde \bar{y}^i ($i = 1, \dots, 16$) se eligió como un número aleatorio del intervalo $[0,1]$.

Para el ajuste de cada SLD se usó un parámetro de aprendizaje de $\alpha = 0.3$. El entrenamiento y la prueba del sistema se realizaron durante 6 épocas. Después de cada época se usaron los datos de prueba para evaluar el desempeño de cada SLD, mediante el cálculo de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$ y $RMSE_{s2}(GD)$ usando (4.6)–(4.8), respectivamente. Todo este proceso se repitió 50 veces usando 50 conjuntos independientes de 1000 datos, por lo que al final se tuvieron 50 valores de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$ y $RMSE_{s2}(GD)$. Los promedios y las desviaciones estándar de estos $RMSE$ s se muestran gráficamente en la figura 4.14 para cada una de las 6 épocas. Observe que:

- Las dos curvas superiores de los SLDT1s son las mismas que las de la figura 4.13.

- Hay una mejora substancial en el desempeño (tanto en la media como en la desviación estándar del $RMSE$) del SLDT2 *singleton* de intervalo sobre su contraparte de tipo 1.

La razón de esta mejora substancial en el desempeño es que ahora sí se está incorporando la incertidumbre de los datos de entrenamiento en las reglas del sistema. Note que aunque la reducción en los promedios finales de $RMSE$, de 0.1553 a 0.1486, sólo representa una reducción de 4.3%, si se tratara de una serie de tiempo financiera, se haría bastante dinero con un 4.3% de mejora en el desempeño del pronóstico. Además, la reducción en la desviación estándar del $RMSE$, de 0.0123 a 0.0107, representa una reducción de más del 13%. Por otro lado, estas mejoras pueden ser aun más substanciales si se utilizan otros diseños de entrenamiento³, como el parcialmente independiente o el dependiente, donde se obtuvieron mejoras de hasta 6.82% en el error y de 23% en la desviación estándar del $RMSE$), ver sección 4.5.3.

A pesar de que el SLDT2 *singleton* de intervalo incorpora en sus reglas la incertidumbre existente en los datos de entrenamiento, éste no toma en cuenta la incertidumbre en las mediciones de entrada, pues usa un fuzificador *singleton*. Más adelante, se diseña un SLDT2 no *singleton* con el método de gradiente descendente que permite tomar en cuenta toda la incertidumbre presente en el problema de pronóstico.

Comparando el sistema diseñado con el método de gradiente descendente contra el de un paso, (sección 4.4.1), se puede ver que:

- La mejora en el promedio del $RMSE$, usando un fuzificador *singleton*, es menor en el SLDT2 de gradiente descendente, debido a que el diseño de un paso con sus 500 reglas de tipo 2 puede absorber más error que las 500 reglas de tipo 1, proporcionalmente.
- La mejora en la desviación estándar del $RMSE$, usando fuzificador *singleton*, es mayor en el SLDT2 de gradiente descendente, ya que en el diseño de un paso los valores del $RMSE$ no varían tanto (por el gran número de reglas que tiene), y en consecuencia, su reducción es menor proporcionalmente.
- Ambos diseños de tipo 2 mejoran su contraparte *singleton* de tipo 1; inclusive, el de gradiente descendente de tipo 2 mejora al no *singleton* de tipo 1.
- El diseño con menor error hasta ahora es el SLDT1 no *singleton* con método directo.

³Recuerde que en las secciones 4.3.2, 4.4.2 y 4.5.2 se usa un diseño de entrenamiento totalmente independiente.

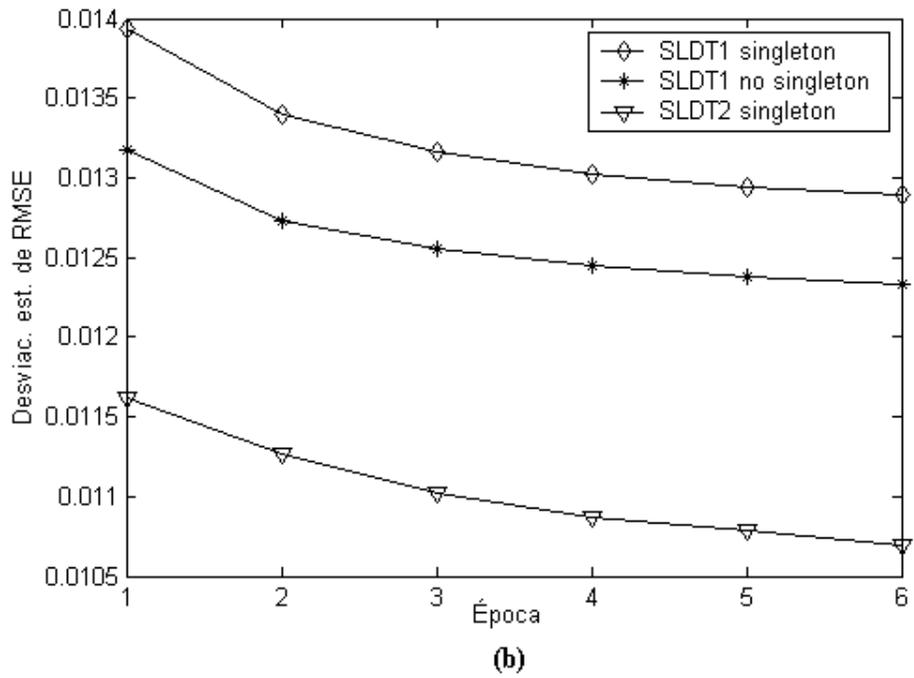
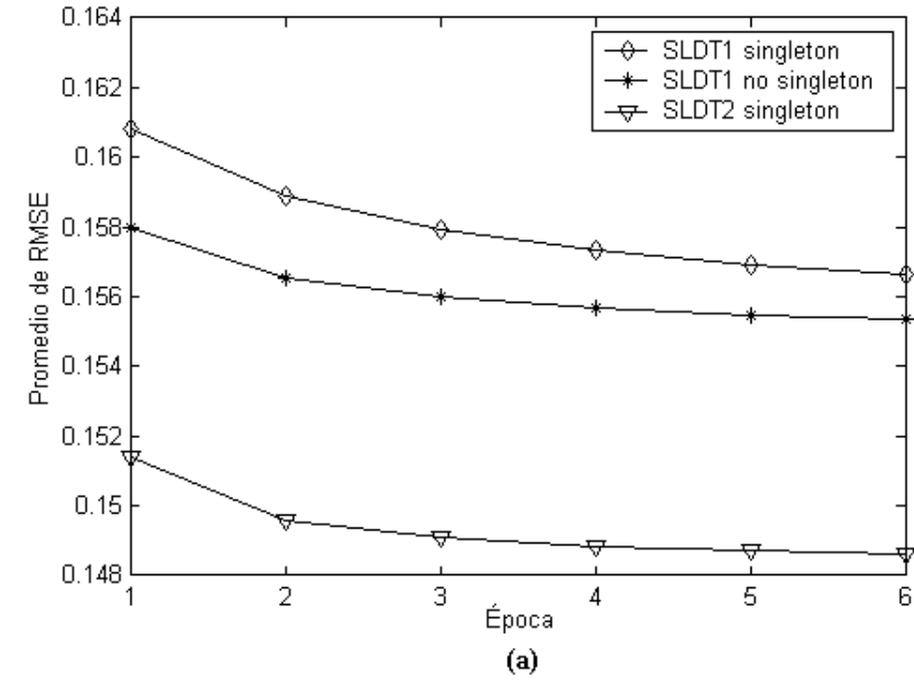


Figura 4.14: Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$ y $RMSE_{s2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación. (a) Promedios. (b) Desviaciones estándar.

4.5 SLDT2 no *singleton*

En esta sección se examinan dos diseños de SLDT2 no *singleton* de intervalo que pronostican la serie de tiempo Mackey-Glass. Uno se basa en el método de diseño de un paso descrito en la sección 3.2.1, y el otro se basa en el método de gradiente descendente de la sección 3.2.2. Ambos sistemas utilizaron los mismos criterios de diseño que la sección 4.4; i.e., los mismos conjuntos de entrenamiento y de prueba, el mismo número de antecedentes por regla ($p = 4$), y usaron también funciones de membresía primarias Gaussianas para los conjuntos difusos de las reglas. Además, se modelaron las cuatro mediciones de entrada ruidosas usando funciones de membresía Gaussianas como en la sección 4.3.

El desempeño de cada SLDT2 no *singleton* se evaluó usando la siguiente expresión:

$$RMSE_{ns2} = \sqrt{\frac{1}{496} \sum_{k=1504}^{1999} [s(k+1) - f_{ns2}(\mathbf{x}^{(k)})]^2} \quad (4.9)$$

4.5.1 Método de un paso

En este método, como en la sección 4.5.1, los límites izquierdo y derecho de la media de cada función de membresía primaria Gaussiana (tanto antecedente como consecuente) se establecían en $x(k) - 0.1/4$ y $x(k) + 0.1/4$, respectivamente; y la desviación estándar de todas ellas se preestableció como $\sigma = 0.1$.

Todo el proceso de generación de 500 reglas (i.e., entrenamiento) y de prueba del sistema, se repitió 50 veces usando 50 conjuntos independientes de 1000 datos, por lo que al final se tuvieron 50 valores de $RMSE_{s1}(UP)$, $RMSE_{ns1}(UP)$, $RMSE_{s2}(UP)$ y $RMSE_{ns2}(UP)$. Los promedios y las desviaciones estándar de estos $RMSEs$ se muestran en la tabla 4.5.

De la tabla 4.5, se puede ver que:

- El SLDT2 no *singleton* supera al desempeño del SLDT2 *singleton*, debido a que se toma en cuenta el ruido de las entradas.
- El desempeño del SLDT2 no *singleton* es prácticamente igual que el de su contraparte de tipo 1. La razón de este hecho se explica en el párrafo siguiente.

Los sistemas diseñados con el método de un paso, generalmente, cuentan con una cantidad muy grande de reglas, lo cual sugiere usar una desviación estándar

Tabla 4.5: Promedios y desviaciones estándar de $RMSE_{s1}(UP)$, $RMSE_{ns1}(UP)$, $RMSE_{s2}(UP)$ y $RMSE_{ns2}(UP)$ obtenidos en 50 simulaciones de diseño.

Sistema	\overline{RMSE}	σ_{RMSE}
SLDT1 <i>singleton</i>	0.1954	0.0082
SLDT2 <i>singleton</i>	0.1784	0.0077
SLDT1 no <i>singleton</i>	0.1371	0.0075
SLDT2 no <i>singleton</i>	0.1370	0.0075

pequeña en las funciones de membresía de los conjuntos difusos (antecedentes y consecuentes), y así generar reglas un tanto específicas (aprovechando que se cuenta con muchos datos). El uso de la desviación estándar pequeña implica que se debe establecer un intervalo aún más pequeño para separar las medias de estas funciones de membresía (medias que corresponden a un mismo conjunto difuso tipo 2). Por estas razones, el método de un paso no permitió modelar bien el ruido del conjunto de entrenamiento con los conjuntos difusos tipo 2, por lo que se dejó de usar para los demás experimentos.

4.5.2 Método de gradiente descendente

En esta sección se presenta el diseño de un SLDT2 no *singleton* de intervalo con el método de gradiente descendente y se comparan sus resultados con los obtenidos en las secciones anteriores, diseñados también con el método de gradiente descendente. Para el entrenamiento del SLDT2 no *singleton* se usó un diseño totalmente independiente (ver sección 3.2), en el cual todos los parámetros del sistema son ajustados y sus valores iniciales no dependen de diseños anteriores.

Para cada uno de los cuatro antecedentes se usaron sólo dos conjuntos difusos, como en las secciones 4.2.2, 4.3.2 y 4.4.2; por lo tanto, se tuvo un total de 16 reglas. Los parámetros iniciales de cada regla se inicializaron de la misma manera que en la sección 4.4.2, y la desviación estándar de las funciones de membresía de las entradas se modelaron como en la sección 4.3.2. El total de parámetros ajustados en este diseño es de 14 parámetros por regla \times 16 reglas + 1 parámetro común a las entradas = 225 parámetros.

Para el ajuste de cada SLD se usó un parámetro de aprendizaje de $\alpha = 0.3$. El entrenamiento y la prueba del sistema se realizaron durante 6 épocas. Después de cada época se usaron los datos de prueba para evaluar el desempeño del sistema

mediante el cálculo de $RMSE_{ns2}(GD)$ usando (4.9). Todo este proceso se repitió 50 veces usando 50 conjuntos independientes de 1000 datos, por lo que al final se tuvieron 50 valores de $RMSE_{ns2}(GD)$. Los promedios y las desviaciones estándar de los $RMSE$ s obtenidos con este diseño y con los anteriores, se muestran gráficamente en la figura 4.15 para cada una de las 6 épocas. Observe que:

- Las tres curvas superiores de los SLDT1s y del SLDT2 no *singleton* de intervalo son las mismas que las de la figura 4.15.
- Hay una pequeña mejora en el desempeño del SLDT2 no *singleton* de intervalo sobre su contraparte *singleton*, tanto en el promedio como en la desviación estándar del RMSE.
- Los valores mínimos del promedio y de la desviación estándar del RMSE del SLDT2 no *singleton* de intervalo se alcanzan mucho antes que su contraparte *singleton*.

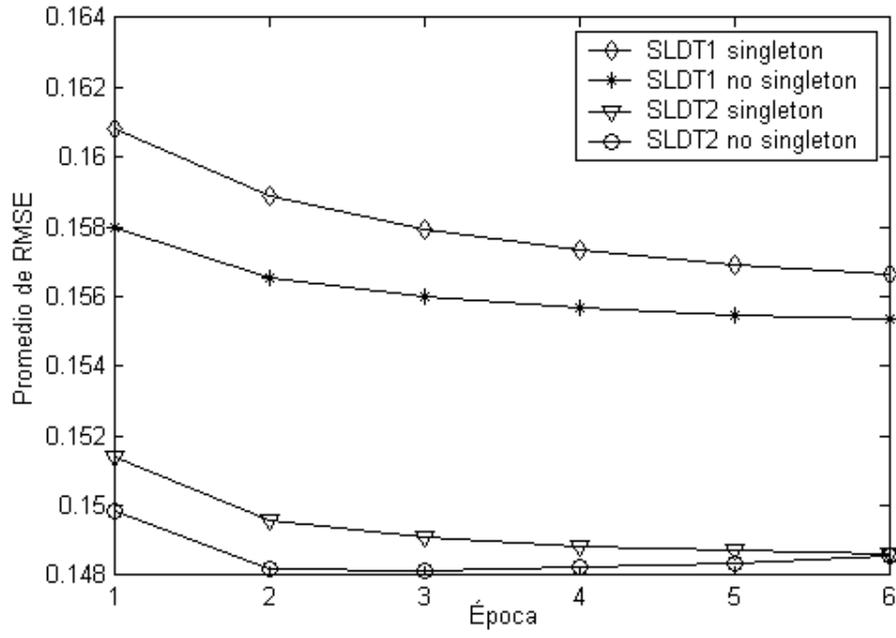
La última observación sugiere el uso del SLDT2 no *singleton* de intervalo sobre los otros diseños, cuando se trabaja en entornos variables en tiempo real. Note que el SLDT2 no *singleton* toma en cuenta toda la incertidumbre presente, tanto en las reglas por los datos de entrenamiento ruidosos como en las entradas por el ruido en las mediciones.

Comparando el sistema diseñado con el método de gradiente descendente contra el de un paso, (sección 4.4.1), se puede ver que:

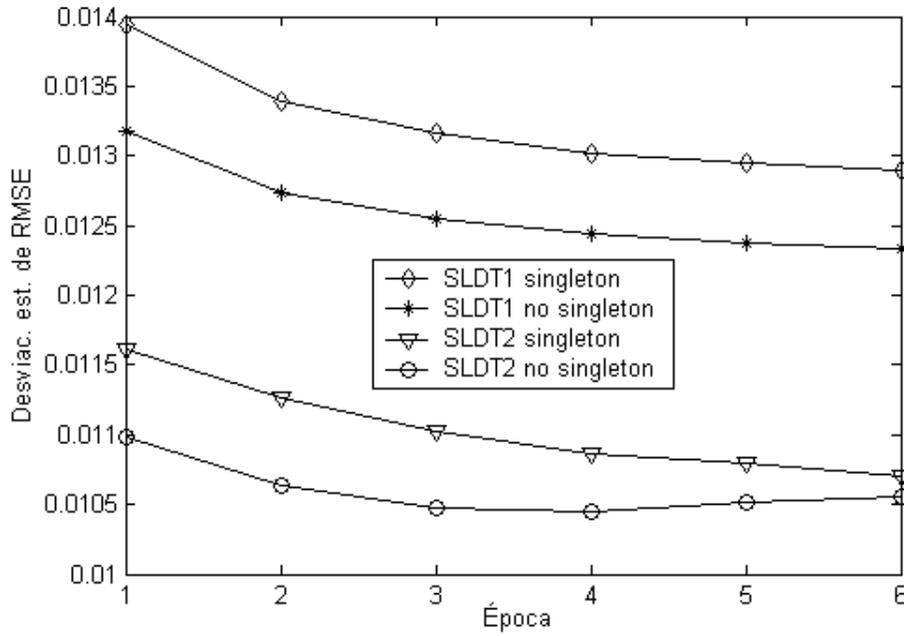
- Todos los diseños de tipo 2 mejoran su contraparte de tipo 1, aunque en el caso del diseño de un paso con fuzificador no *singleton* los desempeños son prácticamente iguales.
- Todos los diseños de tipo 2 con fuzificador no *singleton* mejoran su contraparte de *singleton*, aunque en el caso del diseño de gradiente descendente no por mucho.
- Los sistemas con menor error fueron los SLDs tipo 1 y tipo 2 no *singleton*, diseñados con el método de un paso.

La última observación no significa que los diseños de un paso sean los mejores, pues en caso de que fuera una aplicación de tiempo real, estos sistemas no serían los más adecuados debido a su lentitud al obtener un valor de salida; ahí los mejores serían los de diseño de gradiente descendente⁴.

⁴De hecho en algunos experimentos (variando algunos parámetros por prueba y error), el diseño de gradiente descendente obtuvo errores más cercanos al del $RMSE_{ns2}(UP) = 0.1370$, por ejemplo $RMSE_{ns2}(BP) = 0.1391$.



(a)



(b)

Figura 4.15: Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$, $RMSE_{s2}(GD)$ y $RMSE_{ns2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación utilizando un diseño de entrenamiento totalmente independiente. (a) Promedios. (b) Desviaciones estándar.

4.5.3 Otros diseños de entrenamiento

Las secciones 4.3.2, 4.4.2 y 4.5.2 usaron un diseño de entrenamiento totalmente independiente, en el cual los valores iniciales de los parámetros del sistema no dependen de diseños anteriores. En esta sección se utilizan los diseños de entrenamiento parcialmente independiente y dependiente.

Diseño de entrenamiento parcialmente independiente

En este diseño, al igual que en el totalmente independiente, se hace un ajuste de todos los parámetros del sistema, sólo que aquí se utilizan datos de sistemas anteriormente diseñados como *semillas* para inicializar los parámetros.

Específicamente, los valores iniciales de los parámetros en cada sistema fueron los siguientes:

- En el SLDT1 no *singleton*,
 - los centros y las desviaciones estándar iniciales de las Gaussianas se tomaron de los parámetros finales del SLDT1 *singleton*.
 - la desviación estándar σ_X de las Gaussianas de entrada se inicializó con el valor de la desviación estándar del ruido σ_n .
- En el SLDT2 *singleton*, todos los parámetros se inicializaron de la misma manera que en el diseño de entrenamiento totalmente independiente (ver sección 4.4.2), sólo que aquí el valor de σ_n se tomó de la desviación estándar final σ_X del SLDT1 no *singleton*.
- En el SLDT2 no *singleton*,
 - los parámetros iniciales de las reglas son los mismos que los parámetros iniciales del SLDT2 *singleton*, ya que ambos diseños comparten la misma estructura de las reglas,
 - la desviación estándar σ_n de las Gaussianas de entrada se inicializó con el valor de la desviación estándar σ_X de las Gaussianas de entrada del SLDT1 no *singleton*.

Los promedios y las desviaciones estándar de los *RMSEs* obtenidos se muestran gráficamente en la figura 4.16 para las 6 épocas de entrenamiento, donde se puede observar nuevamente la brecha entre el desempeño de los sistemas tipo 1 y los de tipo 2.

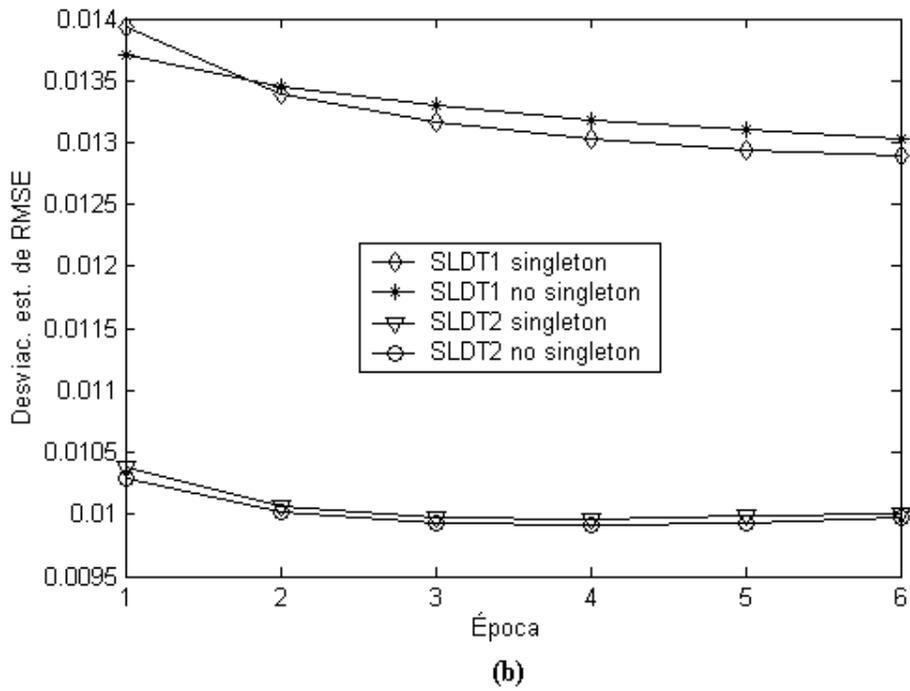
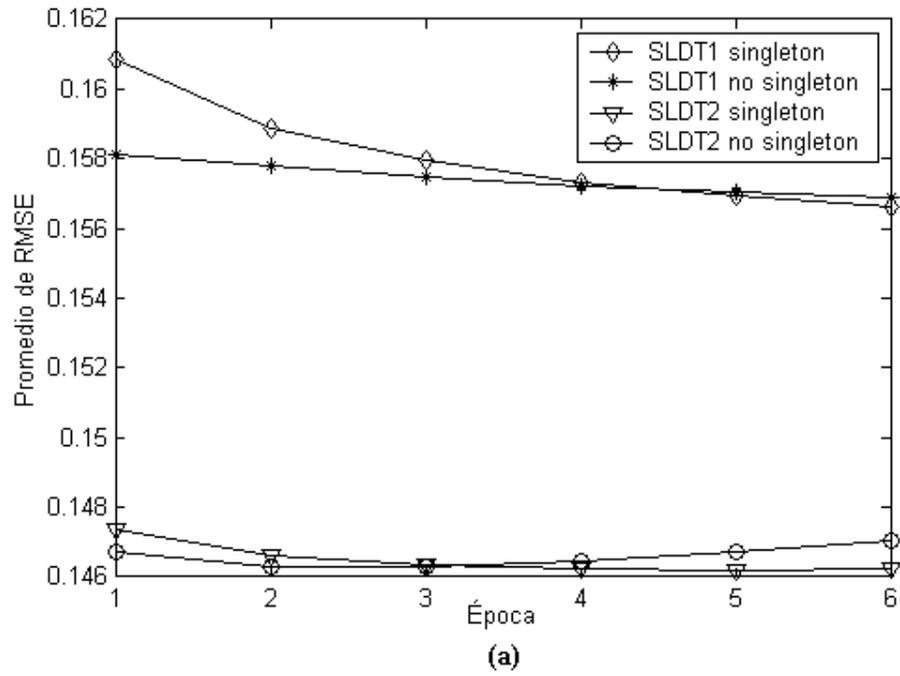


Figura 4.16: Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$, $RMSE_{s2}(GD)$ y $RMSE_{ns2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación utilizando un diseño de entrenamiento parcialmente independiente. (a) Promedios. (b) Desviaciones estándar.

Diseño de entrenamiento dependiente

En este diseño, al igual que en el parcialmente independiente, se utilizan datos de sistemas anteriormente diseñados para inicializar los parámetros, pero aquí se hace ajuste únicamente en los parámetros nuevos; los parámetros comunes mantienen los valores del sistema anterior, como se muestra a continuación:

- En el SLDT1 no *singleton*,
 - los centros y las desviaciones estándar de las Gaussianas se establecieron con los parámetros finales del SLDT1 *singleton*,
 - la desviación estándar σ_X de las Gaussianas de entrada se inicializó con el valor de la desviación estándar del ruido σ_n . El único parámetro ajustado fue σ_X .
- En el SLDT2 *singleton*,
 - las desviaciones estándar de los antecedentes se inicializaron con los valores finales de las desviaciones estándar del SLDT1 *singleton*,
 - los intervalos de incertidumbre para las medias de cada antecedente se inicializaron usando los centros de los antecedentes del SLDT1 *singleton* $\pm 0.25 \times$ la desviación estándar final σ_X de las Gaussianas de entrada del SLDT1 no *singleton*.
 - los puntos izquierdo y derecho de los centroides de los consecuentes se inicializaron usando los centroides finales del SLDT1 *singleton* \pm la desviación estándar final de las Gaussianas de entrada σ_X del SLDT1 no *singleton*. Todos los parámetros fueron ajustados durante 6 épocas.
- En el SLDT2 no *singleton*,
 - todos los parámetros de las reglas fueron tomados de los parámetros finales del SLDT2 *singleton*,
 - la desviación estándar σ_n de las Gaussianas de entrada se inicializó con el valor de σ_X del SLDT1 no *singleton*. El único parámetro ajustado fue σ_n .

Los promedios y las desviaciones estándar de los *RMSEs* obtenidos se muestran gráficamente en la figura 4.17 para las 6 épocas de entrenamiento. En este diseño de entrenamiento, como en el anterior, se observa que bien podría cortarse el entrenamiento desde las primeras épocas, y se tendría un desempeño aceptable; con esto se reduciría el tiempo de entrenamiento de los sistemas sin sacrificar del todo su desempeño.

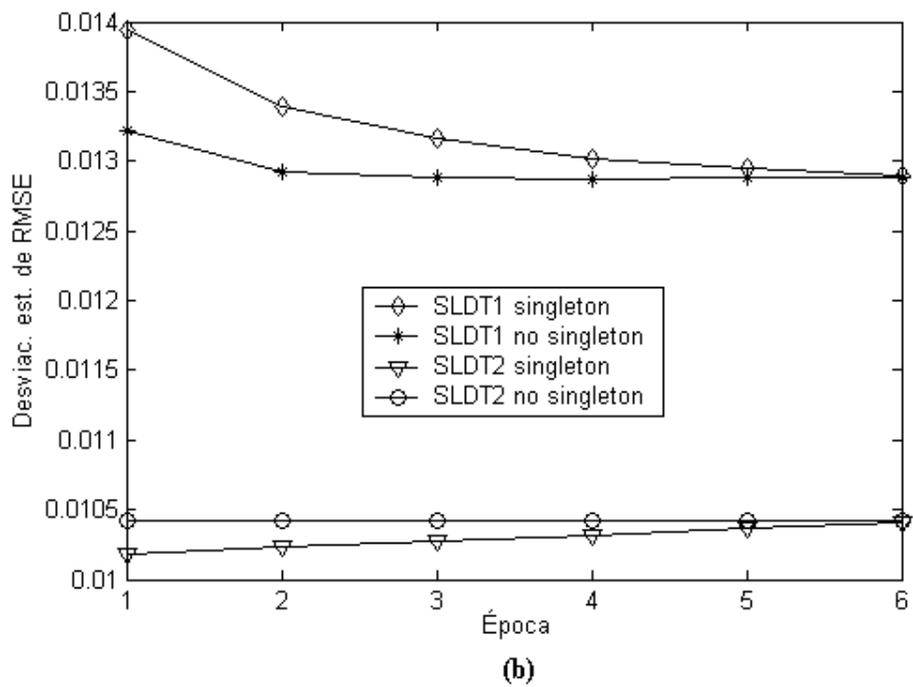
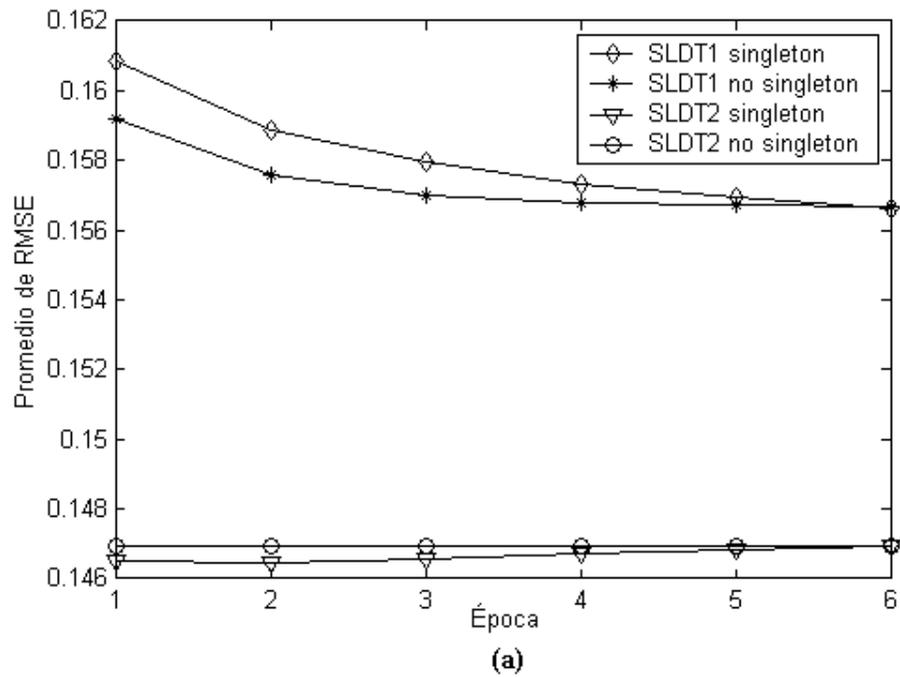


Figura 4.17: Promedios y desviaciones estándar de $RMSE_{s1}(GD)$, $RMSE_{ns1}(GD)$, $RMSE_{s2}(GD)$ y $RMSE_{ns2}(GD)$, obtenidos en 50 simulaciones de diseño. El ajuste se realizó por seis épocas en cada simulación utilizando un diseño de entrenamiento dependiente. (a) Promedios. (b) Desviaciones estándar.

4.6 Pronóstico de serie alterna

En las secciones anteriores, se utilizaron los datos de la serie Mackey-Glass $s(k)$ con condiciones iniciales sin ruido que generaban la gráfica de la figura 4.2. Esta serie era corrompida por un ruido adicional $n(k)$, dando como resultado la serie $x(k) = s(k) + n(k)$ de la figura 4.8. Los sistemas diseñados con el método de gradiente descendente, entrenados con los datos ruidosos $x(k)$, se utilizaron posteriormente para el pronóstico de la serie Mackey-Glass, pero ahora con condiciones iniciales de la serie con ruido, ver figura 4.18 (a). Para cuestiones prácticas, de ahora en adelante se hará referencia a esta serie como la *serie alterna* $s_2(k)$, y a su versión con ruido adicional como la *serie alterna con ruido* $x_2(k)$, ver figura 4.18 (b). Note que en ambos incisos de la figura 4.18 se grafican 1000 datos, i.e., $k = 1001, 1002, \dots, 2000$; pero en realidad los que se utilizaron para hacer el pronóstico son los del conjunto de prueba, i.e., $k = 1505, 1506, \dots, 2000$; pues los sistemas ya habían sido entrenados con los datos de la serie $x(k)$.

Los desempeños obtenidos al utilizar los sistemas para el hacer el pronóstico de la serie alterna con ruido se muestran en la tabla 4.6, donde también se incluyen los errores resultantes al pronosticar la serie con la que fueron entrenados dichos sistemas.

Tabla 4.6: Promedios y desviaciones estándar de los $RMSE$ s obtenidos al hacer 50 simulaciones de pronóstico de la serie original y de la serie alterna, ambas con ruido adicional.

Sistema	\overline{RMSE}	σ_{RMSE}	\overline{RMSE}_2	σ_{RMSE_2}
SLDT1 <i>singleton</i>	0.1566	0.0129	0.1566	0.0112
SLDT1 no <i>singleton</i>	0.1553	0.0123	0.1553	0.0107
SLDT2 <i>singleton</i>	0.1486	0.0107	0.1486	0.0091
SLDT2 no <i>singleton</i>	0.1486	0.0106	0.1486	0.0089

De la tabla 4.6, se puede ver que el comportamiento de los sistemas es muy similar al mostrado en la sección 4.5.2; i.e., los promedios de los $RMSE$ s de la sexta época coinciden con los que aquí se obtuvieron; inclusive las desviaciones estándar resultaron menores. Esto indica que los sistemas fueron poco sensibles al ruido en las condiciones iniciales de la serie.

En la tabla 4.7 se resumen las figuras y tablas de los experimentos más importantes realizados en este capítulo.

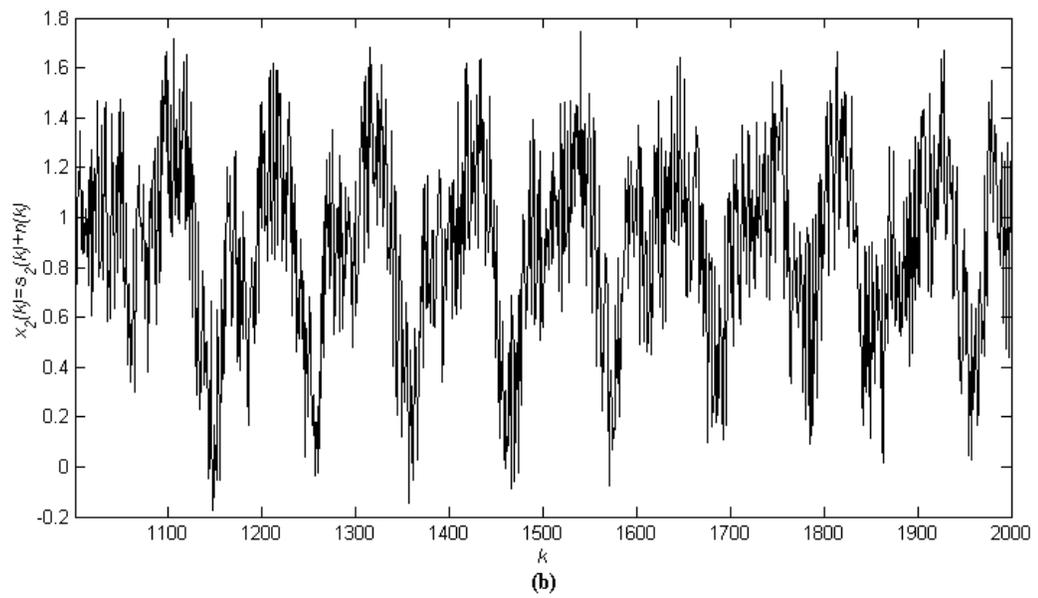
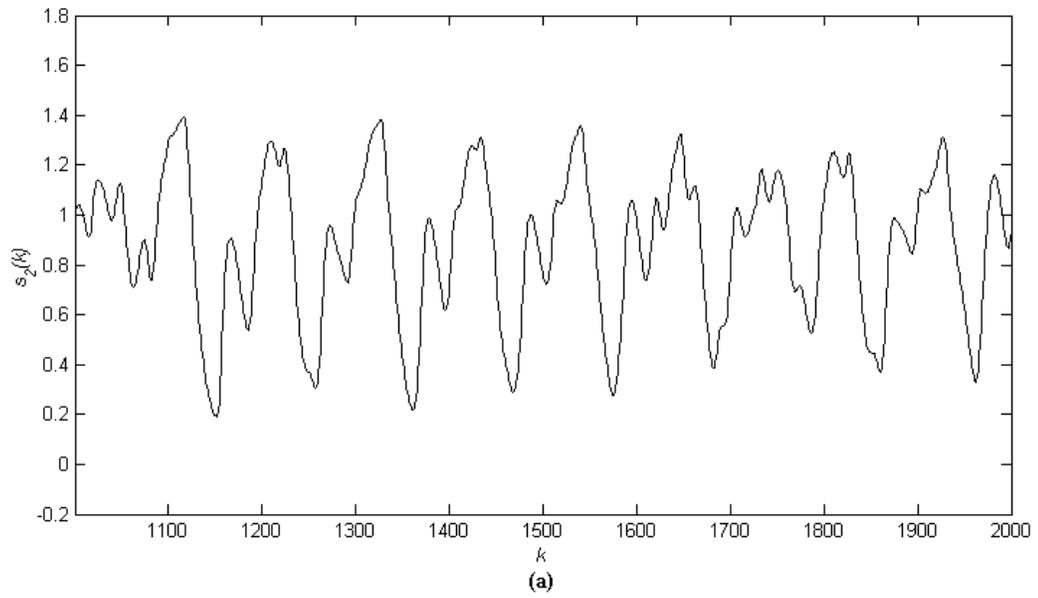


Figura 4.18: Serie de tiempo Mackey-Glass con condiciones iniciales con ruido. (a) Datos sin ruido adicional, $s_2(1001), s_2(1002), \dots, s_2(2000)$. (b) Datos con ruido adicional, $x_2(1001), x_2(1002), \dots, x_2(2000)$

Tabla 4.7: Resumen de experimentos realizados.

Serie	SLD		Fuzificador		Mét. diseño		Dis. entren.		Entren.		Pronóstico		Referencia			
	Orig.	Alt.	T1	T2	Singl.	No Singl.	UP	GD	TI	PI	D	s/r		c/r	s/r	c/r
✓	✓	✓	✓	✓	✓		✓					✓		✓		Fig. 4.4
✓	✓	✓	✓	✓	✓		✓					✓		✓		Fig. 4.7
✓	✓	✓	✓	✓	✓		✓					✓		✓		Fig. 4.9
✓	✓	✓	✓	✓	✓	✓	✓					✓		✓		Fig. 4.10
✓	✓	✓	✓	✓	✓	✓	✓					✓		✓		Fig. 4.12
✓	✓	✓	✓	✓	✓	✓	✓					✓		✓		Fig. 4.13
✓	✓	✓	✓	✓	✓		✓					✓		✓		Tab. 4.4
✓	✓	✓	✓	✓	✓		✓					✓		✓		Fig. 4.14
✓	✓	✓	✓	✓	✓		✓					✓		✓		Tab. 4.5
✓	✓	✓	✓	✓	✓	✓	✓					✓		✓		Fig. 4.15
✓	✓	✓	✓	✓	✓	✓	✓					✓		✓		Fig. 4.16
✓	✓	✓	✓	✓	✓	✓	✓					✓		✓		Fig. 4.17
✓	✓	✓	✓	✓	✓		✓					✓		✓		Tab. 4.6

SLD Sistema de lógica difusa

T1 Tipo 1

T2 Tipo 2

UP Método de diseño de un paso

GD Método de diseño de gradiente descendente

TI Diseño de entrenamiento totalmente independiente

PI Diseño de entrenamiento parcialmente independiente

D Diseño de entrenamiento dependiente

s/r Sin ruido

c/r Con ruido

Capítulo 5

Conclusiones y Trabajo Futuro

El presente trabajo describe una metodología para hacer el pronóstico de la serie de tiempo caótica Mackey-Glass. Esta metodología se utilizó para hacer el diseño de varios de Sistemas de Lógica Difusa, mismos que fueron evaluados en su desempeño mediante el cálculo del $RMSE$ al hacer el pronóstico con los conjuntos de prueba.

Los sistemas utilizados en los experimentos fueron SLDs tipo 1 y tipo 2, con fuzificadores tanto *singleton* como no *singleton*; diseñados bajo el método de gradiente descendente y bajo el método de un paso (ver resultados en la tabla 4.5). Para el método de gradiente descendente, se utilizaron 3 diseños de entrenamiento: el totalmente independiente (figura 4.15), el parcialmente independiente (figura 4.16) y el dependiente (figura 4.17). Los dos últimos presentaron un buen desempeño desde sus primeras épocas de entrenamiento, lo cual sugiere su uso en aplicaciones de tiempo real. Los parámetros ajustados en el método de gradiente descendente se resumen en la tabla 5.1.

Una vez diseñados los sistemas con la serie de tiempo Mackey-Glass, se realizaron experimentos de pronóstico con esos sistemas utilizando la misma serie pero con ruido en sus condiciones iniciales y con ruido adicional a la serie resultante.

De los resultados obtenidos en los experimentos, se puede resumir que:

- Los SLDT1 *singleton* hicieron muy buenos pronósticos cuando no se tiene ruido en absoluto en el ambiente.
- Los SLDT1 no *singleton* absorbieron de cierta manera el ruido presente en los datos del conjunto de prueba, gracias a su fuzificador no *singleton* (en este caso Gaussiano); sin embargo, no hubo manera de modelar el ruido de los datos de entrenamiento, aun si se conocía de antemano el valor de su variación.

Tabla 5.1: Valores iniciales de los parámetros en cuatro SLDs diseñados con el método de gradiente descendente. Cada antecedente está descrito por dos conjuntos difusos ($i = 1, \dots, M$ y $k = 1, \dots, p$).

Sistema	Entrada	Para cada antecedente	Consecuente
SLDT1 <i>singleton</i>	N/A	media: $m_x - 2\sigma_x$ o $m_x + 2\sigma_x$; desv.: $\sigma_{F_k^i}$	$\bar{y}^i \in [0, 1]$
SLDT1 no <i>singleton</i>	$\sigma_X = \sigma_n$	media: $m_x - 2\sigma_x$ o $m_x + 2\sigma_x$; desv.: $\sigma_{F_k^i}$	$\bar{y}^i \in [0, 1]$
SLDT2 <i>singleton</i>	N/A	$[m_x - 2\sigma_x - 0.25\sigma_n, m_x - 2\sigma_x + 0.25\sigma_n]$ o $[m_x + 2\sigma_x - 0.25\sigma_n, m_x + 2\sigma_x + 0.25\sigma_n]$; $\sigma_k^i = 2\sigma_x$	$\bar{y}_l^i = \bar{y}^i - \sigma_n$ $\bar{y}_r^i = \bar{y}^i + \sigma_n$
SLDT2 no <i>singleton</i>	$\sigma_X = \sigma_n$	$[m_x - 2\sigma_x - 0.25\sigma_n, m_x - 2\sigma_x + 0.25\sigma_n]$ o $[m_x + 2\sigma_x - 0.25\sigma_n, m_x + 2\sigma_x + 0.25\sigma_n]$; $\sigma_k^i = 2\sigma_x$	$\bar{y}_l^i = \bar{y}^i - \sigma_n$ $\bar{y}_r^i = \bar{y}^i + \sigma_n$

- Los SLDT2 *singleton* de intervalo pudieron modelar por anticipado el ruido existente en los datos del conjunto de entrenamiento, mediante el uso de funciones de membresía tipo 2, las cuales tienen intervalos de incertidumbre que permiten incluir dicho ruido en las reglas del sistema.
- Los SLDT2 no *singleton* de intervalo pudieron modelar el ruido presente tanto en los datos de entrenamiento (por medio de las funciones de membresía de intervalo tipo 2 en las reglas) como en los datos de prueba (mediante su fuzificador no *singleton*, en este caso Gaussiano).

El sistema con el menor error fue el diseñado con el método de un paso; sin embargo este método típicamente genera una gran cantidad de reglas, por lo que se vuelve extremadamente lento al hacer un solo pronóstico. Este inconveniente podría aminorarse si se utiliza una extensión para SLDT2s del método de *Table Look-Up*, en el que el número de reglas generadas es menor, debido a que evita la redundancia de las reglas y sólo elige las que tengan mayor peso.

Por su parte, el método de gradiente descendente generó un número considerablemente menor de reglas, gracias a que se utilizaron sólo dos conjuntos difusos por cada antecedente (dando un total de $2^4 = 16$ reglas, contra 500 reglas del método de un paso), y se obtuvieron también muy buenos resultados; inclusive, cuando se realizaron los experimentos con la serie alterna, se comprobó que los sistemas entrenados eran robustos al ruido en las condiciones iniciales y al ruido adicional en los conjuntos de prueba (ver resultados en la tabla 4.6).

La presente tesis deja una base para futuros trabajos en pronósticos de series de tiempo, donde la incertidumbre se haga presente por medio del ruido en los datos del conjunto de entrenamiento o de prueba; pudiéndose mejorar los diseños aquí presentados.

Una mejora de los sistemas diseñados con el método de un paso, por ejemplo, es reducir el número de reglas generadas, utilizando el esquema *Table Look-up* de Wang [32] adaptándolo para trabajar con conjuntos difusos tipo 2. Este esquema permite definir de antemano el número de reglas que se obtendrán, predefiniendo el número de conjuntos difusos correspondientes a cada una de las n variables; generándose así una tabla n -dimensional, cuyas celdas se irán llenando con el conjunto difuso consecuente que obtenga el mayor valor de membresía correspondiente al par de entrada-salida evaluado. Al final, se obtendrán las reglas directamente de esta tabla y su número estará limitado por ; evitándose así un sistema con demasiadas reglas cuando el conjunto de entrenamiento es muy grande.

En los sistemas diseñados con el método de gradiente descendente, si se obtienen muchas reglas (por utilizar muchos conjuntos difusos por cada antecedente), se puede reducir esta cantidad aplicando el método SVD-QR (*Singular-Value-QR Decomposition*) de Mendel [17]. Este método permite identificar las FBFs (*Fuzzy Basis Functions*) que más contribuyen al SLD y cuántas son necesarias para representarlo. Las FBFs que menos contribuyen pueden ser descartadas y se procede a diseñar un SLD ahora reducido.

En el caso que se trabaje con un ruido adicional no estacionario, se puede utilizar un fuzificador no *singleton* que modele las entradas mediante conjuntos difusos tipo 2 (recuerde que aquí las entradas se modelaron únicamente con conjuntos difusos tipo 1 por ser un ruido adicional estacionario), de esta manera se puede absorber el ruido en los datos del conjunto de prueba a la hora de hacer el pronóstico.

Apéndice A

Pronóstico de Series de Tiempo

Sea $s(k)$ (donde $k = 1, 2, \dots, N$) una serie de tiempo, tal como la temperatura diaria en una ciudad, los valores medidos de $s(k)$ se denotan como $x(k)$, donde $x(k) = s(k) + n(k)$, y $n(k)$ denota errores de medición (ruido). El problema de pronosticar una serie de tiempo (i.e., predicción) es:

Dada una ventana de p mediciones pasadas de $s(k)$, llamadas $x(k - p + 1), x(k - p + 2), \dots, x(k)$, determine un estimado de un valor futuro de s , $\hat{s}(k + l)$, donde p y l son enteros positivos predeterminados.

Note que si las mediciones son libres de ruido (i.e., perfectas), entonces $x(k - p + 1), x(k - p + 2), \dots, x(k)$ en esta afirmación se reemplazan por $s(k - p + 1), s(k - p + 2), \dots, s(k)$.

Cuando $l = 1$, se obtiene un pronosticador de una etapa de l ; cuando $l = 2$, se obtiene un pronosticador de dos etapas de l ; y, en general, para valores arbitrarios de l , se obtiene un pronosticador de l etapas. En este trabajo se trabajará únicamente con el caso en el que $l = 1$.

Suponga que se da una colección de N datos, $x(1), x(2), \dots, x(N)$. Entonces, como es común hacerlo cuando se usan redes neuronales para pronosticar una serie de tiempo, se particionarán los datos en dos conjuntos: un *conjunto de entrenamiento*, con D datos, $x(1), x(2), \dots, x(D)$, y un *conjunto de prueba* con $N - D$ datos, $x(D + 1), x(D + 2), \dots, x(N)$. Debido a que se usa una ventana de p datos para pronosticar el siguiente dato, hay a lo más $D - p$ pares de entrenamiento, $x^{(1)}, x^{(2)}, \dots, x^{(D-p)}$, donde

$$\begin{aligned}
\mathbf{x}^{(1)} &= [x(1), x(2), \dots, x(p), x(p+1)]^T \\
\mathbf{x}^{(2)} &= [x(2), x(3), \dots, x(p+1), x(p+2)]^T \\
&\vdots \\
\mathbf{x}^{(D-p)} &= [x(D-p), x(D-p+1), \dots, x(D-1), x(D)]^T
\end{aligned} \tag{A.1}$$

En (A.1), los primeros p elementos de $\mathbf{x}^{(t)}$ son las entradas al pronosticador y el último elemento de $\mathbf{x}^{(t)}$ es la salida deseada del pronosticador, i.e.,

$$\mathbf{x}^{(t)} = [p \times 1 \text{ entrada, salida deseada}]^T = [x_1^{(t)}, x_2^{(t)}, \dots, x_p^{(t)}, x_{p+1}^{(t)}]^T \tag{A.2}$$

donde $t = 1, 2, \dots, D - p$. Los datos de entrenamiento son usados en un sistema de lógica difusa (SLD) pronosticador para establecer sus reglas.

Debido a que un valor predicho de s dependerá de p valores pasados de x , habrá p antecedentes en cada regla. Sean estos p antecedentes denotados por x_1, x_2, \dots, x_p . La característica interesante del pronóstico de series de tiempo es que, a pesar de que cada regla tiene p antecedentes, estos antecedentes están todos asociados con la *misma* variable, e.g., la temperatura de la ciudad, y también lo está el consecuente.

Hay al menos tres maneras de extraer reglas de los datos de entrenamiento numéricos:

1. Dejar que los datos establezcan los centros de los conjuntos difusos que aparecen en los antecedentes y los consecuentes de las reglas. El método de un paso corresponde a este esquema, ver sección 2.1.1.
2. Preespecificar conjuntos difusos para los antecedentes y consecuentes, y entonces asociar los datos con los conjuntos difusos. El método de *Table Look-Up* corresponde a este esquema [32], [31].
3. Establecer la arquitectura de un SLD y usar los datos para optimizar sus parámetros. El método de gradiente descendente corresponde a este esquema, ver sección 2.1.2.

Los métodos que se utilizaron en la experimentación fueron el primero y al tercero.

Apéndice B

Conceptos y Operaciones de Conjuntos Difusos Tipo 2

La LD tipo 2 no ha sido tan ampliamente explotada como la LD convencional (tipo 1), cuyas operaciones y conceptos son bien conocidos en el ámbito de la Inteligencia Artificial. Por esta razón, se ha agregado este apéndice al documento, de manera que sirva como referencia rápida de los conceptos y operaciones básicas que se manejan en esta técnica.

B.1 Definiciones y conceptos básicos

En esta sección se explican los conceptos más comunes relacionados con los conjuntos difusos tipo 2.

B.1.1 El concepto de conjunto difuso tipo 2

Cuando no se puede determinar la membresía de un elemento en un conjunto como 0 ó 1, se usan conjuntos difusos de tipo 1. Similarmente, cuando las circunstancias son tan difusas que se tienen problemas para determinar el grado de membresía aun como un número nítido en $[0,1]$, se usan conjuntos difusos de tipo 2¹.

No significa que se deben tener circunstancias extraordinariamente difusas para usar conjuntos tipo 2. Cuando algo es incierto (e.g., una medición), se tienen problemas para determinar su valor exacto, y en este caso usar conjuntos tipo 1 tiene más sentido que usar conjuntos nítidos. Pero, aun en los conjuntos tipo 1, se deben especificar exactamente las funciones de membresía. Si no se puede determinar el valor exacto de una cantidad incierta, ¿cómo se va a determinar su membresía exacta en un conjunto difuso? Esto aplica a los conjuntos de tipo 2, porque aun cuando la

¹Concepto introducido por Zadeh en 1975 [35]

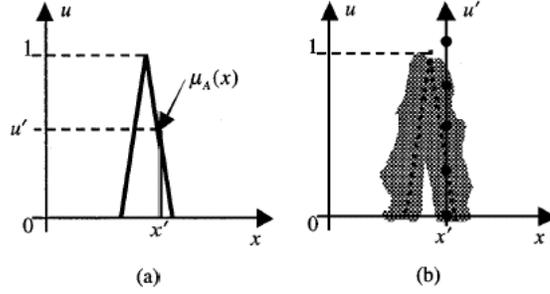


Figura B.1: (a) Función de membresía tipo 1. (b) Función de membresía tipo 1 con trazo difuso, incluyendo discretización en $x = x'$.

membresía es difusa, se puede especificar de manera exacta con este tipo de conjuntos. Si se continúa con el análisis anterior, se puede decir que sería necesario conjuntos difusos de tipo ∞ para representar completamente la incertidumbre. Esto no se puede realizar en la práctica, así que se tienen que utilizar conjuntos de tipo finito, típicamente hasta de tipo 2. Uno podría usar conjuntos de mayor tipo, pero mientras el tipo de conjunto crece, la complejidad del SLD se incrementa rápidamente. Razón por la cual en este trabajo sólo se usan los de tipo 1 y 2.

B.1.2 Definición de conjunto difuso tipo 2

La figura B.1 (a) muestra una función de membresía tipo 1. Si se hace difuso el trazo del triángulo hacia la izquierda o derecha y no necesariamente en cantidades iguales, se obtiene la figura B.1 (b). Entonces, en un valor específico de x , digamos x' , ya no habrá un solo valor para la función de membresía u' , sino que tomará valores donde la línea vertical intersekte el trazo difuso. Estos valores no necesariamente serán ponderados igualmente; por lo tanto, se les puede asignar una distribución de amplitud a todos estos puntos. Al hacer esto para todas las $x \in X$, se crea una función de membresía tridimensional —una función de membresía de tipo 2— que caracteriza a un conjunto difuso de tipo 2.

Un conjunto difuso tipo 2, denotado por \tilde{A} , es caracterizado por una función de membresía tipo 2 [18], $\mu_{\tilde{A}}(x, u)$, donde $x \in X$ y $u \in J_x \subseteq [0, 1]$, i.e.,

$$\tilde{A} = \{(x, u), \mu_{\tilde{A}}(x, u) \mid \forall x \in X, \forall u \in J_x \subseteq [0, 1]\} \quad (\text{B.1})$$

en la cual $0 \leq \mu_{\tilde{A}}(x, u) \leq 1$. \tilde{A} también puede ser expresada como

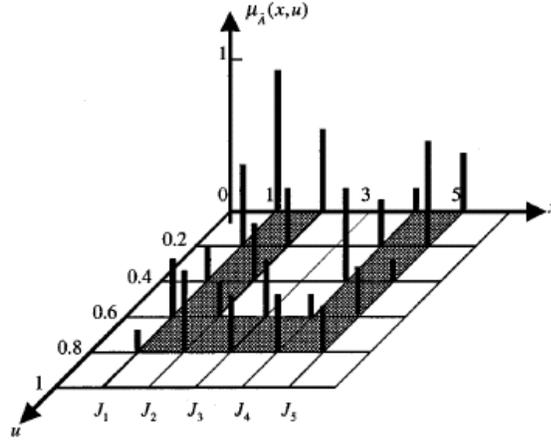


Figura B.2: Ejemplo de un conjunto difuso tipo 2.

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x, u)/(x, u) \quad J_x \subseteq [0, 1] \quad (\text{B.2})$$

donde $\int \int$ denota la unión sobre todas las x y u admisibles. Para universos de discurso discretos \int es reemplazada por \sum .

En la definición de la ecuación (B.1), la restricción de que $\forall u \in J_x \subseteq [0, 1]$ es consistente con la restricción en el tipo 1 de que $0 \leq \mu_{\tilde{A}}(x) \leq 1$. De esta manera, si el trazo confuso desaparece, la variable $u = \mu_{\tilde{A}}(x)$. La restricción adicional de que $0 \leq \mu_{\tilde{A}}(x, u) \leq 1$ es consistente con el hecho de que las amplitudes de una función de membresía deberían estar en $[0, 1]$.

La figura B.2 muestra $\mu_{\tilde{A}}(x, u)$, para x y u discretas. En particular $X = \{1, 2, 3, 4, 5\}$ y $U = \{0, 0.2, 0.4, 0.6, 0.8, 1\}$. Observe, que $J_1 = \{0, 0.2, 0.4, 0.6, 0.8\}$, $J_3 = \{0.6, 0.8\}$, y $J_2 = J_4 = J_5$, y, sólo se han incluido valores en J_1, \dots, J_5 para los cuales $\mu_{\tilde{A}}(x, u) \neq 0$. Cada una de las espigas en la figura B.2 representa a $\mu_{\tilde{A}}(x, u)$ en un par específico de (x, u) .

B.1.3 Función de membresía secundaria

Para cada valor de x , digamos $x = x'$, el plano 2D cuyos ejes son u y $\mu_{\tilde{A}}(x, u)$, se le llama un *corte vertical* de $\mu_{\tilde{A}}(x, u)$. Una función de membresía secundaria es un corte vertical de $\mu_{\tilde{A}}(x, u)$. Esto es $\mu_{\tilde{A}}(x = x', u)$ para $\forall u \in J_{x'} \subseteq [0, 1]$, i.e.

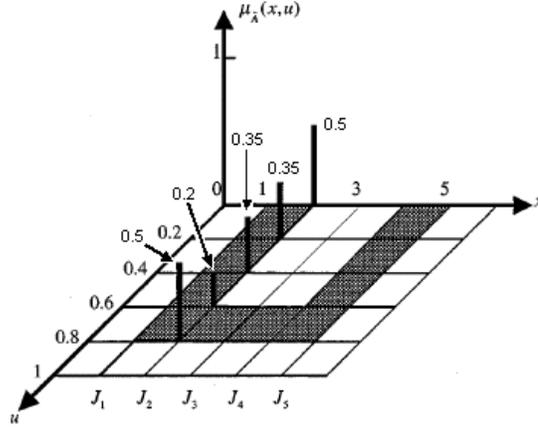


Figura B.3: Ejemplo de un corte vertical para la función de membresía tipo 2 mostrada en la figura B.2.

$$\mu_{\tilde{A}}(x = x', u) \equiv \mu_{\tilde{A}}(x') = \int_{u \in J_{x'}} f_{x'}(u)/u \quad J_{x'} \subseteq [0, 1] \quad (\text{B.3})$$

En la que $0 \leq f_{x'}(u) \leq 1$. Debido a que $\forall x' \in X$, se quita la notación prima en $\mu_{\tilde{A}}(x')$, y se refiere a $\mu_{\tilde{A}}(x)$ como una función de membresía secundaria, y es un conjunto difuso tipo 1, al cual se le conoce también como *conjunto secundario*.

La función de membresía que se muestra en la figura B.2 tiene cinco cortes verticales asociados con ella. El corte vertical en $x = 2$ se muestra en la figura B.3, donde su función de membresía es $\mu_{\tilde{A}}(2) = 0.5/0 + 0.35/0.2 + 0.35/0.4 + 0.2/0.6 + 0.5/0.8$.

Basado en el concepto de conjuntos secundarios, se puede interpretar un conjunto difuso tipo 2 como una unión de todos los conjuntos secundarios, i.e., usando (B.3), se puede reexpresar \tilde{A} a manera de cortes verticales, como

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid \forall x \in X\} \quad (\text{B.4})$$

o como

$$\tilde{A} = \int_{x \in X} \mu_{\tilde{A}}(x)/x = \int_{x \in X} \left[\int_{u \in J_x} f_x(u)/u \right] / x \quad J_x \subseteq [0, 1] \quad (\text{B.5})$$

El *dominio* de una función de membresía secundaria es llamado la **membresía primaria** de x . En (B.5), J_x es la membresía primaria de x , donde $J_x \subseteq [0, 1]$. En el ejemplo anterior, $J_2 = 0, 0.2, 0.4, 0.6, 0.8$. La *amplitud* de una función de membresía

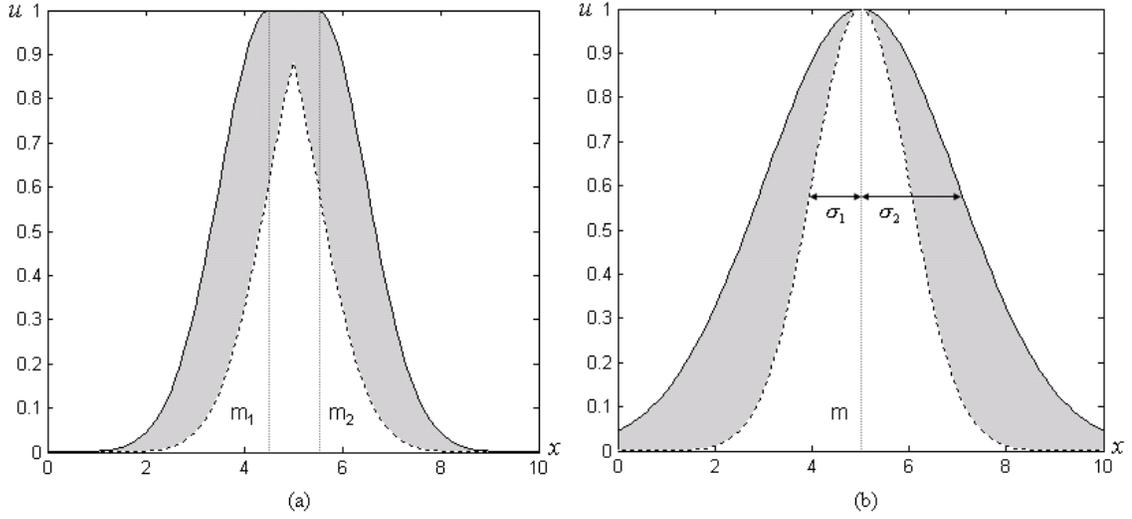


Figura B.4: Las áreas sombreadas representan las FOUs, las líneas sólidas denotan las FMs superiores y las punteadas denotan las inferiores. (a) FM Gaussiana con media incierta. (b) FM Gaussiana con desviación estándar incierta.

secundaria es llamada **grado secundario**. En (B.5), $f_x(u)$ es un grado secundario. En el ejemplo de la figura B.3, la función de membresía secundaria en $x = 2$ es $0.5/0 + 0.35/0.2 + 0.35/0.4 + 0.2/0.6 + 0.5/0.8$. Sus valores de membresía primaria en $x = 2$ son $u = 0, 0.2, 0.4, 0.6, 0.8$, y sus grados secundarios asociados son $0.5, 0.35, 0.35, 0.2, 0.5$, respectivamente.

B.1.4 Huella de incertidumbre (FOU)

La incertidumbre en las membresía primarias de un conjunto difuso tipo 2, \tilde{A} , consiste en una región limitada que se le llama *huella de incertidumbre* [18], FOU (del inglés *Footprint Of Uncertainty*). La FOU es la unión de todas las membresías primarias (i.e., el dominio de las secundarias):

$$FOU(\tilde{A}) = \bigcup_{x \in X} J_x \quad (\text{B.6})$$

La región sombreada de la figura B.1 es la FOU. Otros ejemplos de FOUs se dan en la figura B.4.

La FOU puede ser descrita en términos de sus funciones de membresía superior e

inferior [14]. Una **función de membresía superior** y una **función de membresía inferior** son dos funciones de membresía tipo 1 que son los límites de la FOU de un conjunto difuso tipo 2 \tilde{A} . La función de membresía superior está asociada con el límite superior de $FOU(\tilde{A})$, y se denota como $\bar{\mu}_{\tilde{A}}(x), \forall x \in X$.

$$\bar{\mu}_{\tilde{A}}(x) \equiv \overline{FOU(\tilde{A})} \quad \forall x \in X \quad (\text{B.7})$$

La función de membresía inferior está asociada con el límite inferior de $FOU(\tilde{A})$, y se denota como $\underline{\mu}_{\tilde{A}}(x), \forall x \in X$, i.e.,

$$\underline{\mu}_{\tilde{A}}(x) \equiv \underline{FOU(\tilde{A})} \quad \forall x \in X \quad (\text{B.8})$$

Se puede reexpresar (B.5) en términos de funciones de membresía superior e inferior de la siguiente manera:

$$\begin{aligned} \tilde{A} = \mu_{\tilde{A}}(x, u) &= \int_{x \in X} \mu_{\tilde{A}}(x)/x = \int_{x \in X} \left[\int_{u \in J_x} f_x(u)/u \right] / x \\ &= \int_{x \in X} \left[\int_{u \in [\bar{\mu}_{\tilde{A}}(x), \underline{\mu}_{\tilde{A}}(x)]} f_x(u)/u \right] / x \end{aligned} \quad (\text{B.9})$$

Se puede ver de esta ecuación, que la función de membresía secundaria $\mu_{\tilde{A}}(x)$ puede ser expresada en términos de funciones de membresía superior e inferior como:

$$\mu_{\tilde{A}}(x) = \int_{u \in [\bar{\mu}_{\tilde{A}}(x), \underline{\mu}_{\tilde{A}}(x)]} f_x(u)/u \quad (\text{B.10})$$

El término *huella de incertidumbre* es muy útil, porque no sólo enfoca la atención en la incertidumbre inherente en una función de membresía tipo 2 específica, cuya forma es una consecuencia directa de la naturaleza de esta incertidumbre, sino que también provee una descripción verbal conveniente del dominio entero de soporte de todos los grados secundarios de una función de membresía de tipo 2. Además, permite visualizar un conjunto difuso gráficamente en dos dimensiones en vez de tres, lo que hace que sea más fácil de dibujarlos.

B.1.5 Conjunto difuso tipo 2 de intervalo

La FOU sombreada implica que hay una distribución que se posa encima de ella —la nueva tercera dimensión de conjuntos difusos—. La forma que tome dicha distribución depende de la elección específica de los grados secundarios. Cuando todos

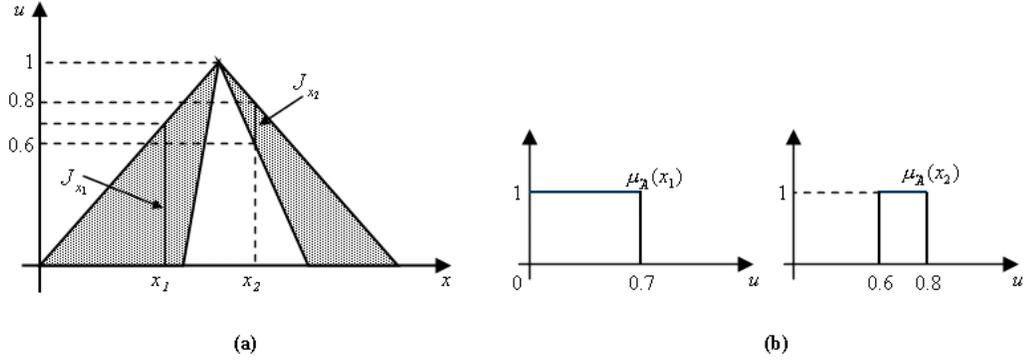


Figura B.5: La región sombreada en (a) es la FOU para un conjunto difuso tipo 2. (a) Las membresías primarias, J_{x_1} y J_{x_2} , y sus funciones de membresía secundarias asociadas $\mu_{\tilde{A}}(x_1)$ y $\mu_{\tilde{A}}(x_2)$ se muestran en los puntos x_1 y x_2 . (b) Las funciones de membresía secundarias son conjuntos de intervalo.

estos grados secundarios son iguales a uno, el conjunto difuso tipo 2 resultante es llamado *conjunto difuso tipo 2 de intervalo* [14], y reflejan una incertidumbre uniforme en las membresías primarias de x . Tales conjuntos son los conjuntos tipo 2 más utilizados a la fecha.

En la figura B.5, la FOU es sombreada uniformemente para indicar que es un conjunto difuso tipo 2 de intervalo; por lo tanto, una FOU uniformemente sombreada también representa completamente el conjunto difuso tipo 2 de intervalo $\mu_{\tilde{A}}(x, u)$. También se muestran las membresías primarias y la función de membresía secundaria en dos puntos.

B.1.6 Conjuntos incluidos

Un conjunto difuso tipo 2, \tilde{A} , puede ser considerado como una colección de conjuntos difusos tipo 2, \tilde{A}_e , a los que se les llama *conjuntos difusos tipo 2 incluidos* en \tilde{A} .

Para universos de discurso discretos X y U , un conjunto tipo 2 incluido \tilde{A}_e [18] tiene N elementos, donde \tilde{A}_e contiene exactamente un elemento de $J_{x_1}, J_{x_2}, \dots, J_{x_N}$, llamados u_1, u_2, \dots, u_N , cada uno con su grado secundario asociado, llamado $f_{x_1}(u_1), f_{x_2}(u_2), \dots, f_{x_N}(u_N)$, i.e.,

$$\tilde{A}_e = \sum_{i=1}^N [f_{x_i}(u_i)/u_i] / x_i \quad u_i \in J_{x_i} \subseteq U = [0, 1] \quad (\text{B.11})$$

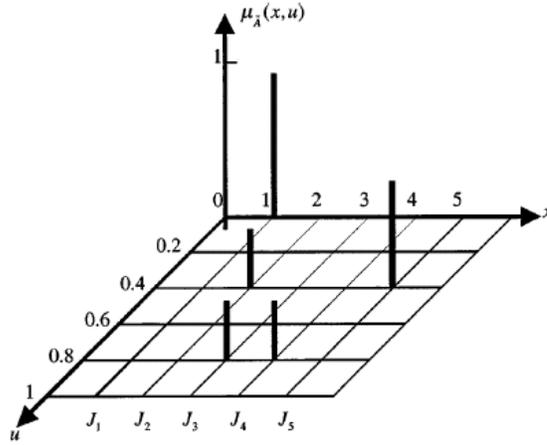


Figura B.6: Ejemplo de un conjunto incluido tipo 2 asociado con la función de membresía de la figura B.2.

El conjunto \tilde{A}_e está incluido en \tilde{A} , y el total² de \tilde{A}_e s es $\prod_{i=1}^N M_i$

Para universos de discurso discretos X y U , un conjunto tipo 1 incluido A_e tiene N elementos, donde \tilde{A}_e tiene N elementos, uno de cada $J_{x_1}, J_{x_2}, \dots, J_{x_N}$, llamados u_1, u_2, \dots, u_N , i.e.,

$$A_e = \sum_{i=1}^N u_i/x_i \quad u_i \in J_{x_i} \subseteq U = [0, 1] \quad (\text{B.12})$$

El conjunto A_e es la unión de todas las membresías primarias (i.e., el dominio) del conjunto \tilde{A}_e en ecuación B.11, y hay un total de $\prod_{i=1}^N M_i A_e$.

La figura B.6 muestra uno de los $1250 = (5 \times 5 \times 2 \times 5 \times 5)$ posibles conjuntos tipo 2 incluidos para la función de membresía que se muestra en la figura B.2. Observe que el conjunto tipo 1 incluido que es asociado con este conjunto tipo 2 incluido es $A_e = 0/1 + 0.4/2 + 0.8/3 + 0.8/4 + 0.4/5$.

Un conjunto difuso tipo 1 puede también ser expresado como un conjunto difuso tipo 2. Su representación de tipo 2 es $(1/\mu_F(x))/x$, o para simplificar $1/\mu_F(x)$, $\forall x \in X$. La notación $1/\mu_F(x)$ significa que la función de membresía secundaria tiene sólo un valor en su dominio, llamado la membresía primaria $\mu_F(x)$, en la cual el grado secundario es igual a 1.

²Para conjuntos difusos tipo 2 continuos, hay un incontable número de conjuntos difusos tipo 2 incluidos, y este concepto no es muy útil.

B.2 Operaciones en conjuntos difusos tipo 2 generales

Considere dos conjuntos difusos tipo 2 \tilde{A} y \tilde{B} , i.e.,

$$\tilde{A} = \int_X \mu_{\tilde{A}}/x = \int_X \left[\int_{J_x^u} f_x(u)/u \right] / x \quad J_x^u \subseteq [0, 1] \quad (\text{B.13})$$

$$\tilde{B} = \int_X \mu_{\tilde{B}}/x = \int_X \left[\int_{J_x^w} g_x(w)/w \right] / x \quad J_x^w \subseteq [0, 1] \quad (\text{B.14})$$

Observe que en (B.13) y (B.14), u y w son variables ficticias usadas para diferenciar las funciones de membresía de x en \tilde{A} y \tilde{B} , respectivamente. En esta sección se establecen las operaciones para estos conjuntos difusos tipo 2 generales.

B.2.1 Unión de conjuntos difusos tipo 2

La unión de \tilde{A} y \tilde{B} es otro conjunto difuso tipo 2, de la misma manera que la unión de dos conjuntos difusos tipo 1 A y B es otro conjunto difuso tipo 1; y se puede expresar de la forma siguiente:

$$\tilde{A} \cup \tilde{B} \Leftrightarrow \mu_{\tilde{A} \cup \tilde{B}}(x, v) = \int_{x \in X} \mu_{\tilde{A} \cup \tilde{B}}(x)/x = \int_{x \in X} \left[\int_{v \in J_x^v \subseteq [0, 1]} h_x(v)/v \right] / x \quad (\text{B.15})$$

donde $h_x(v)$ representa una función ficticia para realizar —por el Principio de Extensión de Zadeh [34]— una conorma T^3 en el dominio $v = u \vee w$; y para realizar también una norma T de las funciones de membresía secundarias, $\mu_{\tilde{A}}(x)$ y $\mu_{\tilde{B}}(x)$, las cuales son conjuntos de tipo uno. Por lo tanto, se puede expresar la función de membresía secundaria de (B.15) como un conjunto difuso tipo 1 de la siguiente manera:

$$\mu_{\tilde{A} \cup \tilde{B}}(x) = \int_{v \in J_x^v \subseteq [0, 1]} h_x(v)/v = \int_{u \in J_x^u} \int_{w \in J_x^w} f_x(u) \star g_x(w) / (u \vee w) \quad x \in X \quad (\text{B.16})$$

En esta ecuación \vee indica máximo, \star indica mínimo o producto, y \iint indica unión sobre $J_x^u \times J_x^w$. Otra forma de expresar (B.16) es en términos de las funciones de membresía secundarias de \tilde{A} y \tilde{B} , $\mu_{\tilde{A}}(x)$ y $\mu_{\tilde{B}}(x)$, como

$$\mu_{\tilde{A} \cup \tilde{B}}(x) = \int_{u \in J_x^u} \int_{w \in J_x^w} f_x(u) \star g_x(w) / (u \vee w) \equiv \mu_{\tilde{A}} \sqcup \mu_{\tilde{B}} \quad x \in X \quad (\text{B.17})$$

³También llamada norma S

donde \sqcup indica la operación llamada *join* [20].

Lo que (B.17) expresa es que para realizar la operación *join* entre dos funciones de membresía secundarias, $\mu_{\tilde{A}}$ y $\mu_{\tilde{B}}$, se debe realizar $u \vee w$ entre cada par posible de membresías primarias u y w ; y que se debe obtener el grado secundario de $\mu_{\tilde{A} \cup \tilde{B}}(x)$ a medida que se calcula la norma T entre los grados secundarios correspondientes a $\mu_{\tilde{A}}$ y $\mu_{\tilde{B}}$, $f_x(u)$ y $g_x(w)$, respectivamente. De acuerdo a (B.15), esto se debe hacer para $\forall x \in X$ para así obtener $\mu_{\tilde{A} \cup \tilde{B}}(x, v)$.

Si más de una combinación de u y w da el mismo punto $u \vee w$, entonces en la operación *join* se mantiene el de mayor grado de membresía. Suponga, por ejemplo, que $u_1 \vee w_1 = \theta$ y $u_2 \vee w_2 = \theta$; entonces en el cálculo de (B.17), se tendrían los términos

$$f_x(u_1) \star g_x(w_1) / \theta + f_x(u_2) \star g_x(w_2) / \theta$$

donde $+$ denota unión. Combinar estos dos términos en el θ común es una operación de tipo 1, en la cual se usa una conorma T para la unión. En este trabajo se utiliza el máximo como conorma T.

B.2.2 Intersección de conjuntos difusos tipo 2

La intersección de \tilde{A} y \tilde{B} es otro conjunto difuso tipo 2, de la misma manera que la intersección de dos conjuntos difusos tipo 1 A y B es otro conjunto difuso tipo 1; y se puede expresar de la forma siguiente:

$$\tilde{A} \cap \tilde{B} \Leftrightarrow \mu_{\tilde{A} \cap \tilde{B}}(x, v) = \int_{x \in X} \mu_{\tilde{A} \cap \tilde{B}}(x) / x \quad (\text{B.18})$$

El desarrollo de $\mu_{\tilde{A} \cap \tilde{B}}(x)$ es similar al de $\mu_{\tilde{A} \cup \tilde{B}}(x)$, sólo que ahora el dominio de la función ficticia sería obtenido por medio de $v = u \wedge w$, como se muestra a continuación:

$$\mu_{\tilde{A} \cap \tilde{B}}(x) = \int_{u \in J_x^u} \int_{w \in J_x^w} f_x(u) \star g_x(w) / (u \wedge w) \equiv \mu_{\tilde{A}} \sqcap \mu_{\tilde{B}} \quad x \in X \quad (\text{B.19})$$

donde \wedge y \star indican mínimo o producto, y \sqcap indica la operación llamada *meet* [20].

Lo que (B.19) expresa es que para realizar la operación *meet* entre dos funciones de membresía secundarias, $\mu_{\tilde{A}}$ y $\mu_{\tilde{B}}$, se debe realizar $u \wedge w$ entre cada par posible de membresías primarias u y w ; y que se debe obtener el grado secundario de $\mu_{\tilde{A} \cap \tilde{B}}(x)$ a medida que se calcula la norma T entre los grados secundarios correspondientes a $\mu_{\tilde{A}}$ y $\mu_{\tilde{B}}$, $f_x(u)$ y $g_x(w)$, respectivamente. De acuerdo a (B.18), esto se debe hacer para

$\forall x \in X$ para así obtener $\mu_{\tilde{A} \cap \tilde{B}}(x, v)$.

Si más de una combinación de u y w da el mismo punto $u \wedge w$, entonces en la operación *meet* (al igual que en la operación *join*) se mantiene el de mayor grado de membresía.

B.2.3 Complemento de un conjunto difuso tipo 2

El complemento de \tilde{A} es otro conjunto difuso tipo 2, de la misma manera que el complemento de un conjunto difuso tipo 1 es otro conjunto difuso tipo 1; y se puede expresar de la forma siguiente:

$$\tilde{\tilde{A}} \Leftrightarrow \mu_{\tilde{\tilde{A}}}(x, v) = \int_{u \in J_x^u} \mu_{\tilde{A}}(x)/x \quad (\text{B.20})$$

En esta ecuación $\mu_{\tilde{A}}(x)$ es una función de membresía secundaria; i.e., en cada valor de x , $\mu_{\tilde{A}}(x)$ es una función (a diferencia de la situación de tipo 1, donde en cada valor de x , $\mu_{\tilde{A}}(x)$ es un valor). Usando nuevamente el Principio de Extensión, se tiene que

$$\mu_{\tilde{\tilde{A}}}(x) = \int_{u \in J_x^u} f_x(u)/(1-u) \equiv \neg \mu_{\tilde{A}}(x) \quad x \in X \quad (\text{B.21})$$

donde \neg denota la operación llamada *negation* [20].

Lo que (B.21) expresa es que para realizar la operación *negation* de la función de membresía secundaria $\mu_{\tilde{A}}(x)$, se debe calcular $1-u$ para $\forall u \in J_x^u$; y el grado secundario de $\mu_{\tilde{\tilde{A}}}(x)$ en $1-u$ es el grado secundario correspondiente a $\mu_{\tilde{A}}(x)$, $f_x(u)$. De acuerdo a (B.20), esto se debe hacer para $\forall x \in X$ para así obtener $\mu_{\tilde{\tilde{A}}}(x, v)$.

A continuación se da un ejemplo⁴ de cómo calcular (B.17), (B.19) y (B.21) para dos conjuntos difusos tipo 2 \tilde{A} y \tilde{B} , y para un elemento particular x para el cual las funciones de membresía secundaria en estos dos conjuntos son $\mu_{\tilde{A}}(x) = 0.5/0 + 0.7/0.1$ y $\mu_{\tilde{B}}(x) = 0.3/0.4 + 0.9/0.8$. De (B.17), usando mínimo como norma T y máximo como conorma T, se tiene:

$$\begin{aligned} \mu_{\tilde{A} \cup \tilde{B}}(x) &= \mu_{\tilde{A}} \sqcup \mu_{\tilde{B}} = (0.5/0 + 0.7/0.1) \sqcup (0.3/0.4 + 0.9/0.8) \\ &= \frac{0.5 \wedge 0.3}{0 \vee 0.4} + \frac{0.5 \wedge 0.9}{0 \vee 0.8} + \frac{0.7 \wedge 0.3}{0.1 \vee 0.4} + \frac{0.7 \wedge 0.9}{0.1 \vee 0.8} \\ &= 0.3/0.4 + 0.5/0.8 + 0.3/0.4 + 0.7/0.8 \\ &= \max\{0.3, 0.3\}/0.4 + \max\{0.5, 0.7\}/0.8 \\ &= 0.3/0.4 + 0.7/0.8 \end{aligned}$$

⁴Ejemplo presentado en [18]

De (B.19), usando la norma T mínimo y la conorma T máximo, se tiene:

$$\begin{aligned}
\mu_{\tilde{A} \cap \tilde{B}}(x) &= \mu_{\tilde{A}} \sqcap \mu_{\tilde{B}} = (0.5/0 + 0.7/0.1) \sqcap (0.3/0.4 + 0.9/0.8) \\
&= \frac{0.5 \wedge 0.3}{0 \wedge 0.4} + \frac{0.5 \wedge 0.9}{0 \wedge 0.8} + \frac{0.7 \wedge 0.3}{0.1 \wedge 0.4} + \frac{0.7 \wedge 0.9}{0.1 \wedge 0.8} \\
&= 0.3/0 + 0.5/0 + 0.3/0.1 + 0.7/0.1 \\
&= \max\{0.3, 0.5\}/0 + \max\{0.3, 0.7\}/0.1 \\
&= 0.5/0 + 0.7/0.1
\end{aligned}$$

Finalmente, de (B.21), se tiene:

$$\begin{aligned}
\mu_{\tilde{A}}(x) &= \neg \mu_{\tilde{A}}(x) = 0.5/(1 - 0) + 0.7/(1 - 0.1) \\
&= 0.5/1 + 0.7/0.9
\end{aligned}$$

B.3 Operaciones en conjuntos difusos tipo 2 de intervalo

Esta sección se enfoca en las operaciones para conjuntos difusos tipo 2 porque son los que más se van a utilizar en este trabajo. De (B.17), (B.19) y (B.21), es claro que para calcular la unión, intersección o complemento de conjuntos difusos tipo 2, se necesitan calcular las operaciones *join*, *meet* y *negation* de funciones de membresía secundarias, las cuales son conjuntos difusos tipo 1; por lo que se hará referencia a los grados de membresía de un conjunto tipo 2 de intervalo como *conjuntos tipo 1 de intervalo* [14].

B.3.1 Operación *join* en conjuntos de intervalo

Sean $F = \sum_{v \in F} 1/v$ y $G = \sum_{w \in G} 1/w$ dos conjuntos tipo 1 de intervalo (llamados frecuentemente conjuntos de intervalo) con dominios $[l_r, r_f]$ y $[l_g, r_g]$, respectivamente. La operación *join* entre F y G puede ser expresada usando (B.17) como

$$F \sqcup G = \int_{u \in J_x^u} \int_{w \in J_x^w} 1 \star 1 / (u \vee w) = \int_{q \in [l_f \vee l_g, r_f \vee r_g]} 1/q \quad (\text{B.22})$$

donde $q = u \vee w$, $u \in J_x^u$ y $w \in J_x^w$.

La operación *join* de n conjuntos de intervalo tipo 1 $F_1, \dots, F_n, \sqcup_{i=1}^n F_i$, teniendo dominios $[l_1, r_1], \dots, [l_n, r_n]$, respectivamente, es un conjunto de intervalo con dominio $[(l_1 \vee l_2 \vee \dots \vee l_n), (r_1 \vee r_2 \vee \dots \vee r_n)]$ donde \vee denota máximo.

B.3.2 Operación *meet* en conjuntos de intervalo

La operación *meet* entre dos conjuntos tipo 1 de intervalo F y G bajo las normas t mínimo o producto (i.e., \star) está dado por

$$F \sqcap G = \int_{u \in J_x^u} \int_{w \in J_x^w} 1 \star 1 / (u \wedge w) = \int_{q \in [l_f \star l_g, r_f \star r_g]} 1/q \quad (\text{B.23})$$

donde $q = u \star w$, $u \in J_x^u$ y $w \in J_x^w$.

La operación *meet* de n conjuntos de intervalo tipo 1 $F_i (i = 1, 2, \dots, n)$ es un conjunto de intervalo con dominio $[(l_1 \star l_2 \star \dots \star l_n), (r_1 \star r_2 \star \dots \star r_n)]$.

B.3.3 Complemento de un conjunto de intervalo

La operación *negation* de un conjunto tipo 1 de intervalo F está dada por:

$$\neg F = \int_{u \in J_x^u} 1/(1 - u) \quad (\text{B.24})$$

Recuerde que estas operaciones para conjuntos tipo 1 de intervalo deben calcularse para $\forall x \in X$ para que así se obtenga el conjunto difuso tipo 2 de intervalo resultante. En otras palabras, cada vez que se hace el cálculo de cualquiera de estas operaciones, se obtiene la función de membresía secundaria (que son conjuntos difusos tipo 1) para un valor específico de $x = x'$ solamente. Por lo tanto, es necesario hacer el cálculo de todas estas membresías secundarias (los conjuntos de intervalo) pero sobre todo el dominio (las membresías primarias), obteniéndose finalmente la descripción completa del conjunto de intervalo tipo 2.

En la figura B.7 se presenta un ejemplo ilustrativo del resultado de aplicar las operaciones de unión, intersección y complemento a dos conjuntos difusos tipo 2 de intervalo \tilde{F} y \tilde{G} . Si se traza una línea vertical en la figura B.7 (a) en cualquier valor de x , por ejemplo en $x = 7.5$, se obtienen los conjuntos de intervalo F y G que se muestran en la figura B.7 (b). Para obtener las operaciones *join*, *meet* y *negation* entre estos conjuntos de intervalo, se aplican las fórmulas de las ecuaciones (B.22), (B.23) y (B.24), respectivamente, como se muestra a continuación:

- Para la operación *join*, tomando máximo como conorma T , se obtiene un conjunto de intervalo con dominio $[(0.04 \vee 0.61), (0.46 \vee 0.80)] = [0.61, 0.80]$; que en este caso $F \sqcup G = G$.
- Para la operación *meet*, tomando mínimo como norma T , se obtiene un conjunto de intervalo con dominio $[(0.04 \star 0.61), (0.46 \star 0.80)] = [0.04, 0.46]$; que en este caso $F \sqcap G = F$.

- Para la operación *negation*, se obtiene un conjunto de intervalo $\neg F$ con dominio $[(1-0.01), (1-0.46)] = [0.54, 0.96]$; como se puede observar en la figura B.7 (f).

Si estas operaciones se realizan para $\forall x \in [0, 15]$, entonces se obtienen finalmente la unión, intersección y complemento de \tilde{F} y \tilde{G} ; dando como resultado las gráficas de las figuras B.7 (c), (d) y (e), respectivamente.

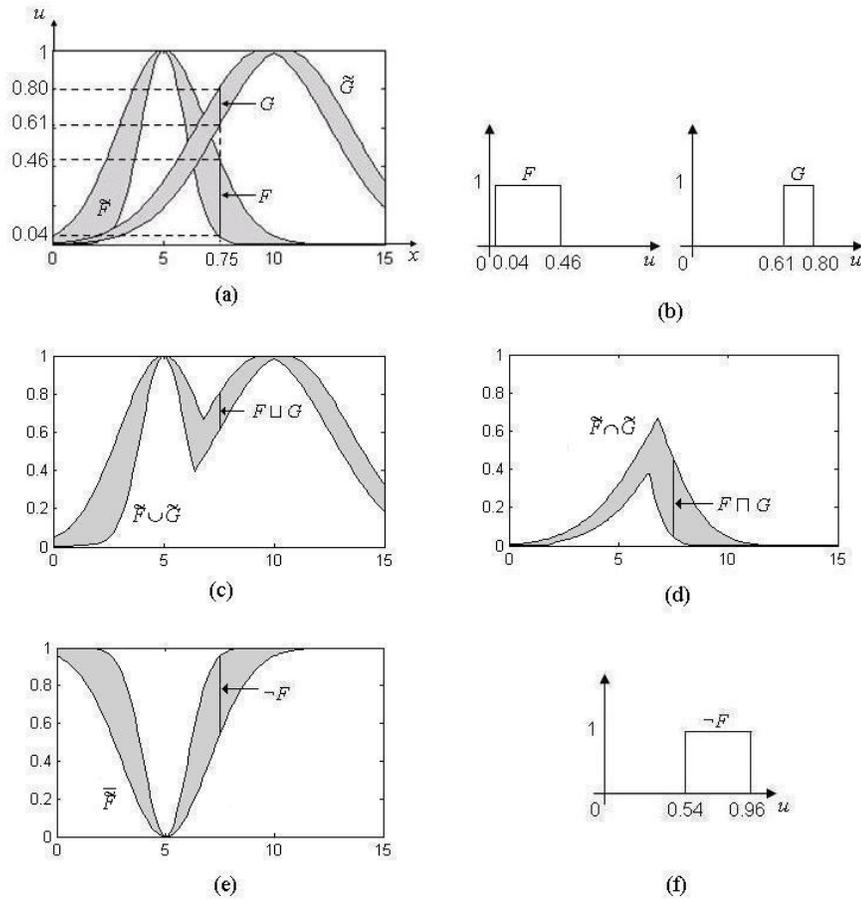


Figura B.7: (a) Conjuntos difusos tipo 2 de intervalo \tilde{F} y \tilde{G} . (b) Conjuntos tipo 1 de intervalo F y G cuando $x = 0.75$. (c) Unión entre \tilde{F} y \tilde{G} . (d) Intersección entre \tilde{F} y \tilde{G} . (e) Complemento de \tilde{F} . (f) Operación *negation* de F .

Bibliografía

- [1] Box, George and Jenkins, Gwilym: *Time Series Analysis: Forecasting, and Control*, Prentice Hall 1994.
- [2] Casdagli, M.: “A Dynamical Systems Approach to Modeling Input-Output Systems.”, *Nonlinear Modeling and Forecasting*, vol. XII of *SFI Studies in the Sciences of Complexity*, (pp. 265–281), Addison–Wesley, New York 1992.
- [3] Crowder III, R. S.: “Predicting the Mackey-Glass Time Series with Cascade Correlation Learning.”, G. E. Hinton D. S. Touretzky and T. J. Sejnowski., eds., *Proceedings of the 1990 Connectionist Models Summer School*, (pp. 117–123), Morgan Kaufman, San Mateo, CA 1990.
- [4] DeLaurentis, Daniel and Mavris, Dimitri N.: “A new model for the aerospace design process based on a control system analogy.”, *AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA Sept. 2000.
- [5] Farmer, J. D.: “Chaotic attractors of an infinite dimensional dynamical system.”, *Physica*, vol. 4–D, (pp. 366–393) 1982.
- [6] Hohenson, J. and Mendel, Jerry M.: “Two-Pass Orthogonal Least-Squares Algorithm to Train and Reduce the Complexity of Fuzzy Logic Systems.”, *J. of Intelligent and Fuzzy Systems*, vol. 4, (pp. 295–308) 1996.
- [7] Jang, J.-S. R.: “ANFIS – Adaptive-Network-Based Fuzzy Inference System.”, Tech. rep., University of California, Berkeley, CA 1993.
- [8] Jang, J.-S. R., Sun, C-T., and Mizutani, E.: *Neuro-Fuzzy and Soft-Computing*, Prentice-Hall, Upper Saddle River, N.J. 1997.
- [9] Jang, J.-S. R. and Sun, Chuen-Tsai: “Neuro-fuzzy modeling and control.”, *Proceedings of the IEEE*, vol. 83, (pp. 378–406) Mar. 1995.
- [10] Karnik, Nilesh N. and Mendel, Jerry M.: “Introduction to Type-2 Fuzzy Logic Systems.”, *IEEE Fuzzy System Proc.*, vol. 2, (pp. 915–920), Anchorage, AK May 1998.

- [11] Karnik, Nilesh N. and Mendel, Jerry M.: “Type-2 Fuzzy Logic Systems.”, *IEEE Trans. Fuzzy Syst.*, vol. 7, (pp. 643–658) Dec. 1999.
- [12] Kyrtsov, Catherine and Terraza, Michel: “Is it possible to study jointly chaotic and ARCH behaviour?”, *Computing and Economics and Finance Intern. Conf.*, Barcelona Jul 2000.
- [13] Liang, Qiliang and Mendel, Jerry M.: “An Introduction to Type-2 TSK Fuzzy Logic Systems.”, *IEEE Fuzzy Syst. Conf. Proc.*, vol. 3, (pp. 1534–1539), Seoul, Korea Aug. 1999.
- [14] Liang, Qiliang and Mendel, Jerry M.: “Interval Type-2 Fuzzy Logic Systems: Theory and Design.”, *IEEE Trans. on Fuzzy Syst.*, vol. 8, (pp. 535–550) Oct. 2000.
- [15] Lillekjendlie, B., Kugiumtzis, D., and Christophersen, N.: “Chaotic time series – Part II: System identification and prediction.”, *Modeling, Identification and Control*, vol. 15, (pp. 225–243) Feb. 1994.
- [16] Mackey, M. C. and Glass, L.: “Oscillation and Chaos in Physiological Control Systems.”, *Science*, vol. 197, (pp. 287–289) 1977.
- [17] Mendel, Jerry M.: *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, Prentice-Hall, Upper Saddle River, N.J. 2001.
- [18] Mendel, Jerry M. and John, Robert I. Bob: “Type-2 Fuzzy Sets Made Simple.”, *IEEE Trans. on Fuzzy Syst.*, vol. 10, (pp. 117–127) Apr. 2002.
- [19] Mendenhall, William, Wackerly, Dennis D., and Scheaffer, Richard L.: *Mathematical Statistics with Applications*, fourth ed., Duxbury Press, Belmont, CA 1990.
- [20] Mizumoto, M. and Tanaka, K.: “Some Properties of Fuzzy Sets of Type-2.”, *Information and Control*, vol. 31, (pp. 312–340) 1976.
- [21] Moody, John: “Fast Learning in Multi-Resolution Hierarchies.”, D. S. Touretzky, ed., *Advances in Neural Information Processing Systems 1*, (pp. 29–39) 1989.
- [22] Nagle, R. Kent and Saff, Edward B.: *Fundamentals of Differential Equations*, third ed., Addison-Wesley Publishing Company 1993.
- [23] Park, Seihwan and Lee-Kwang, H.: “A Designing Method for Type-2 Fuzzy Logic Systems Using Genetic Algorithms.”, *IFSA World Congress and 20th NAFIPS International Conference*, vol. 5, (pp. 2567–2572), Vancouver, Canada Jul. 2001.

- [24] Phillips, L. D.: *Bayesian Statistics for Social Scientists*, Thomas Y. Crowell Company, New York 1973.
- [25] Rasband, S. N.: *Chaotic Dynamics of Non-linear Systems*, Wiley, New York 1990.
- [26] Sanger, T. D.: “A Tree-Structured Adaptive Network for Function Approximation in High-Dimensional Spaces.”, *IEEE Trans. Neural Networks*, vol. 2, (pp. 285–293) Mar. 1991.
- [27] Schiavo, A. Lo and Luciano, A. M.: “Powerful and Flexible Fuzzy Algorithm for Nonlinear Dynamic System Identification.”, *IEEE Trans. on Fuzzy Systems*, vol. 9, (pp. 828–835) Dec. 2001.
- [28] Taniguchi, Michiaki, Haft, Michael, Hollmén, Jaakko, and Tresp, Volker: “Fraud Detection in Communication Networks using Neural and Probabilistic Methods.”, *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 2, (pp. 1241–1244) 1998.
- [29] Vasilakos, A., Ricudis, C., Anagnostakis, K. G., Predrycz, W., Pitsillides, A., and Gao, X. Z.: *Evolutionary-Fuzzy Prediction for Strategic ID-QoS, Routing in Broadband Networks*, Department of computer science, University of Cyprus Sept. 2001.
- [30] Wang, Li-Xin: *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*, Prentice Hall, Englewood Cliffs, N.J. 1994.
- [31] Wang, Li-Xin: *A Course in Fuzzy Systems and Control*, Prentice Hall, Upper Saddle River, N.J. 1997.
- [32] Wang, Li-Xin and Mendel, Jerry M.: “Generating Fuzzy Rules by Learning from Examples.”, *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 22, (pp. 1414–1427) Nov./Dec. 1992.
- [33] Wang, Xin: ““On-Line” Time Series Prediction System – EFuNN-T.”, Tech. rep., University of Otago, Dunedin, New Zealand 2001.
- [34] Zadeh, L. A.: “Fuzzy Sets.”, *Information and Control*, vol. 8, (pp. 338–353), L.A. 1965.
- [35] Zadeh, L. A.: “The Concept of a Linguistic Variable and its Application to Approximate Reasoning.”, *Information Sciences*, vol. 8, (pp. 199–249) 1975.

Centro de Información-Biblioteca



30002006318802