

Universidad de Huelva

Departamento de Ingeniería Electrónica, de Sistemas
Informáticos y Automática



**Metodología para mantener la interpretabilidad en el
modelado utilizando sistemas borrosos**

**Memoria para optar al grado de doctor
presentada por:**

Miguel Ángel Vélez Vélez

Fecha de lectura: 2 de diciembre de 2011

Bajo la dirección del doctor:

Omar Sánchez Pérez

Huelva, 2012

ISBN: 978-84-15147-70-1

D.L.: H 15- 2012



Universidad
de Huelva

Departamento de Ingeniería Electrónica, de Sistemas
Informáticos y Automática

Escuela Técnica Superior de Ingeniería

Metodología para mantener la
interpretabilidad en el modelado utilizando
sistemas borrosos

Tesis Doctoral de
Miguel Ángel Vélez Vélez

Dirigida por
Omar Sánchez Pérez

UNIVERSIDAD DE HUELVA

2011

A mis padres, por creer siempre en mí y confiar en mis posibilidades. A mi mujer, por apoyarme en estos largos años de lucha para conseguir terminar este trabajo. A mi hija, por darme tantas razones para seguir adelante.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a las personas que han colaborado en la realización de este trabajo.

En primer lugar, al Dr. Omar Sánchez Pérez por su labor de dirección de esta tesis, por ofrecerme sus ideas y colaboración a lo largo de todos estos años.

Al Dr. Manuel Emilio Gegúndez Árias, por ofrecer sus conocimientos y aportar luz en algunos aspectos importante de la tesis.

A D. José Luis Fernández Alonso, por ofrecerme en todo momento su ayuda desinteresada en la revisión y traducción de los textos científicos.

Me gustaría reconocer, además, la valiosa e inestimable ayuda ofrecida por el Dr. José Manuel Andujar Márquez, sin cuyo esfuerzo, su tiempo y su dedicación jamás hubiése conseguido terminar este trabajo.

A Thuarek, reconocido miembro de la comunidad de Hispania, por aportarme tantos buenos momentos en estos años.

Quisiera agradecer también a todos los compañeros, amigos y familiares el apoyo y el ánimo constante que he recibido.

Índice general

1. PLANTEAMIENTO GENERAL DE LA TESIS	11
1.1. RESUMEN DE LA TESIS.....	12
1.2. INNOVACIONES QUE APORTA LA TESIS	15
1.3. RENDIMIENTO CIENTÍFICO DE LA TESIS	17
2. INTRODUCCIÓN A LA LÓGICA BORROSA.....	21
2.1. NATURALEZA E HISTORIA DE LA LÓGICA BORROSA	21
2.2. APLICACIONES.....	22
2.3. SOFT COMPUTING.....	27
2.3.1. <i>Redes Neuronales</i>	27
2.3.2. <i>Algoritmos genéticos</i>	32
2.3.3. <i>¿Por qué lógica borrosa?</i>	35
2.4. CONCLUSIONES	36
3. CONCEPTOS BÁSICOS DE SISTEMAS BORROSOS.....	37
3.1. CONJUNTOS BORROSOS	37
3.2. FUNCIONES DE PERTENENCIA	42
3.3. OPERACIONES SOBRE CONJUNTOS BORROSOS.....	49
3.4. VARIABLES LINGÜÍSTICAS	56
3.5. DISEÑO DE SISTEMAS BORROSOS	61
3.5.1. <i>Razonamiento borroso</i>	62
3.5.2. <i>Sistema de inferencia borroso</i>	64
3.5.3. <i>Tipos de sistemas borrosos</i>	68
3.5.4. <i>Metodología para el diseño de sistemas borrosos</i>	75
3.6. CONCLUSIONES	82
4. ADAPTACIÓN PARAMÉTRICA EN SISTEMAS BORROSOS.....	85
4.1. INTRODUCCIÓN.....	85
4.2. ADAPTACIÓN DE UNA BASE DE REGLAS BORROSAS.....	87
4.2.1. <i>La adaptación de parámetros del antecedente de las reglas</i>	90

4.2.2.	<i>Función de pertenencia de Gauss</i>	91
4.2.3.	<i>Función de pertenencia triangular</i>	94
4.2.4.	<i>La adaptación de parámetros del consecuente de las reglas</i>	96
4.3.	CONCLUSIONES	101
5.	LA RAZÓN DE SOLAPE	103
5.1.	INTRODUCCIÓN.....	103
5.2.	OBTENCIÓN DEL PARÁMETRO ψ	107
5.3.	AJUSTE DE PARÁMETROS BASÁNDOSE EN LA RS	107
5.4.	ELECCIÓN DEL PARÁMETRO RS^d	109
5.5.	AJUSTE PARAMÉTRICO DE LOS CONSECUENTES	110
5.6.	ALGORITMO DE ADAPTACIÓN CON AJUSTE BASADO EN LA RS.....	114
5.7.	CONCLUSIONES	118
6.	EL ÁREA DE SOLAPE	121
6.1.	UN NUEVO ÍNDICE EN EL AJUSTE PARAMÉTRICO DE FP.....	121
6.2.	AJUSTE DE PARÁMETROS BASÁNDOSE EN EL AS	126
6.3.	ELECCIÓN DEL PARÁMETRO AS^d	127
6.4.	ALGORITMO DE ADAPTACIÓN CON AJUSTE BASADO EN EL AS.....	127
6.5.	CONCLUSIONES	129
7.	APLICACIONES	131
7.1.	ECUACIÓN DIFERENCIAL DE SEGUNDO ORDEN.....	131
7.2.	HORNO DE GAS	142
7.3.	HELICÓPTERO DE AEROMODELISMO	155
7.3.1.	<i>Aspectos técnicos previos</i>	155
7.3.2.	<i>Modelo completo</i>	157
7.3.3.	<i>Estimación del modelo</i>	160
7.4.	CONCLUSIONES	169
8.	CONCLUSIONES Y DESARROLLOS FUTUROS	171
8.1.	CONCLUSIONES	171

8.2. DESARROLLOS FUTUROS.....	172
APÉNDICE I.....	175
APÉNDICE II.....	177
APÉNDICE III.....	179
BIBLIOGRAFÍA	183

1. Planteamiento general de la Tesis

El objetivo fundamental de esta Tesis es mejorar la interpretabilidad de los modelos adaptativos borrosos, de forma que sin perder mucha precisión en la estimación del modelo podamos mejorar la interpretabilidad del mismo. Se propone, por tanto, una nueva metodología que permita ajustar las funciones de pertenencia del modelo borroso, de forma tal que éstas puedan ser evaluadas por un experto, y sus reglas sean más interpretables y dotadas de semántica.

Este trabajo se ha desarrollado dentro del Grupo de Investigación de Control y Robótica (TEP192) y del Grupo de Investigación Sistemas Inteligentes y Minería de Datos (TIC198). Estos dos grupos, bajo la dirección del Dr. José Manuel Andújar Márquez el primero y del Dr. Antonio Peregrín Rubio el segundo, están realizando numerosos trabajos en los campos de la lógica borrosa, el control no lineal, Soft Computing y Minería de datos, tecnologías de la Web y recuperación de información.

En esta Tesis se trabaja principalmente con sistemas borrosos de tipo Takagi-Sugeno-Kang (TSK), ya que éstos permiten la representación de cualquier sistema no lineal mediante un número reducido de reglas. Aunque se estudian otros sistemas borrosos, tan sólo se hace con carácter introductorio, ya que el trabajo está basado fundamentalmente en sistemas borrosos de tipo TSK. La Tesis se centra en el uso de funciones de pertenencia gaussianas, las cuales poseen la ventaja de poseer un soporte no finito, lo que garantiza que siempre exista un grado de pertenencia definido en el universo de discurso. Esta ventaja plantea a su vez un inconveniente adicional que se ha solventado de forma novedosa en este trabajo, ya que al no tener un soporte finito, se extienden en el universo de discurso teóricamente hasta el infinito.

Este primer capítulo se estructura en tres secciones. La primera presenta un resumen de la Tesis, donde se describen cada uno de los capítulos que la conforman para dar una idea general de su contenido. En la segunda se describen las aportaciones novedosas que este trabajo aporta al estado del arte. En la tercera y última sección se muestra el rendimiento científico de la Tesis, el cual supone una validación del trabajo realizado por parte de la comunidad científica tanto nacional como internacional, así como una muestra clara del grado de innovación de la misma.

1.1. Resumen de la Tesis

La memoria está organizada en ocho capítulos y tres apéndices, cuyo contenido se describe brevemente a continuación:

En el **Capítulo 1**, *Planteamiento general de la Tesis*, como su propio nombre indica, se realiza una descripción de la estructura de la Tesis y se comentan brevemente cada uno de los capítulos que la componen. Además, se recogen las innovaciones principales de la Tesis y se efectúa un análisis del rendimiento científico que este trabajo ha producido hasta la fecha.

El **Capítulo 2**, *Introducción a la lógica borrosa*, realiza un pequeño recorrido en la evolución de la lógica borrosa y en sus principales aplicaciones. Se mencionan los principales avances en la aplicación de las técnicas basadas en sistemas borrosos y se recopilan algunos de los grupos de investigación y entidades más representativos que trabajan dentro del campo de la lógica borrosa a nivel nacional y andaluz. Por último, se resumen las principales técnicas de *Soft Computing* (las redes neuronales, la computación evolutiva y la lógica borrosa) y se justifica el uso de la lógica borrosa como la técnica adecuada para utilizar la metodología aplicada en este trabajo.

El **Capítulo 3**, *Conceptos básicos de sistemas borrosos*, estudia los fundamentos de los sistemas borrosos. Este estudio se realiza con la intención de que el texto sea autocontenido y no se requiera acudir a otros textos para comprender los mecanismos que se utilizarán en los capítulos siguientes, sin embargo, abundan las referencias que permiten ampliar dichos contenidos. Se comienza con la definición de conjunto borroso, su representación y propiedades, pasando posteriormente a explicar las funciones de pertenencia como mecanismo de definición de los mismos. Se detallan las principales operaciones sobre conjuntos borrosos y se define el concepto de variable lingüística.

Se explica posteriormente el proceso en el diseño de un modelo borroso, el razonamiento borroso y el sistema de inferencia borroso. Se detallan los principales tipos de sistemas borrosos (Mamdani, SAM, TSK, Tsukamoto,...) y se resume la metodología para el diseño de sistemas borrosos.

El **Capítulo 4**, *Adaptación paramétrica en sistemas borrosos*, trata sobre los mecanismos de adaptación de parámetros en los sistemas adaptativos borrosos. Estos

mecanismos de adaptación buscan preferentemente la precisión en la búsqueda del modelo, sin incorporar métodos en el ajuste de los parámetros, lo cual reduce su interpretabilidad. Se estudia uno de los métodos más populares de adaptación paramétrica, el método presentado por Wang [202], utilizando un desborrosificador centro promedio, con método de inferencia producto. Se presenta la adaptación de los parámetros del antecedente de las reglas, centrándonos en funciones de pertenencia de Gauss y funciones de pertenencia triangulares. Posteriormente, la adaptación de los parámetros para el consecuente de las reglas es analizada, tanto si el consecuente es Singleton, TSK o un consecuente no lineal.

En el **Capítulo 5**, *La Razón de Solape*, se propone el primer índice para medir el grado de solape que se produce entre dos funciones de pertenencia adyacentes. Esta medida nos permitirá realizar el ajuste correspondiente en los parámetros implicados en la estimación del modelo. En un principio se detallan las 4 reglas establecidas por Marsh [129] para considerar que las funciones de pertenencia están correctamente distribuidas en el universo de discurso. Se definen los conceptos de *Razón de Solape*, *Factor de Ancho* y *Razón de Solape Deseada*.

La elección del Factor de ancho y de la Razón de Solape Deseada depende del criterio del diseñador y tiene una gran importancia en la correcta estimación del modelo, por lo que se aportan las consideraciones que pueden facilitar en gran medida la elección de los valores adecuados para estos parámetros.

Se incorpora la Razón de Solape en el procedimiento de adaptación de los parámetros, ajustando los mismos para que el solape entre las funciones de pertenencia sea el adecuado. Este procedimiento de ajuste se realiza tanto en la parte precedente de las reglas para los centros y anchos de las funciones de pertenencia, como en los consecuentes.

En el caso de los consecuentes, si éstos son escalares o de tipo TSK, la aplicación de la Razón de Solape se sustituye por un método acorde a su representación. Aparecen así dos nuevos conceptos, *Razón de Solape de los Consecuentes* y *Razón de Solape de los Consecuentes Deseada*, los cuales son explicados en profundidad.

Por último, se describe detalladamente el algoritmo empleado para realizar la estimación de un modelo aplicando la Razón de Solape como índice en el ajuste de los parámetros.

El **Capítulo 6**, *El Área de Solape*, propone un nuevo índice para medir el verdadero solape existente entre dos funciones de pertenencia adyacentes. En este caso se mide el área de solape real entre dos funciones de pertenencia. El nuevo índice depende de dos factores, de la función con un área menor y de la posición que ésta ocupe con respecto a la otra. En función de estos dos factores nos surgen 4 posibles casos por lo que se describe la expresión del Área de Solape para cada uno de ellos.

Se detalla el procedimiento de ajuste de los parámetros utilizando el Área de Solape, centrándonos en funciones gaussianas, y se define un nuevo concepto, el de *Área de Solape Deseada*. Este nuevo parámetro es crítico en el adecuado ajuste de las funciones por lo que se proponen los valores más óptimos para que el diseñador pueda aplicarlo correctamente.

El capítulo termina con la descripción del nuevo algoritmo en el que se aplica como índice de ajuste de los parámetros el Área de Solape.

El **Capítulo 7**, *Aplicaciones*, aborda la estimación de diferentes modelos a partir de las técnicas presentadas en los capítulos anteriores con el objetivo de mostrar la mejora de la interpretabilidad en los resultados finales. En primer lugar se trabaja con una planta descrita por una ecuación diferencial de segundo orden. Se presentan los resultados obtenidos tanto aplicando la Razón de Solape como el Área de Solape y se comparan los resultados con el modelo obtenido sin utilizar ningún tipo de ajuste a los parámetros. Se pasa a continuación a estudiar el modelo serie-temporal de Box y Jenkins [28] que se obtiene a partir de un horno de gas. La mejora en la interpretabilidad de los resultados puede observarse con claridad en este ejemplo, tanto aplicando la Razón como el Área de Solape. Por último, se ha elegido el modelo matemático de un helicóptero de aeromodelismo presentado por Kim-Tilbury [98]. Se ha optado por mostrar sólo algunos de los resultados obtenidos debido a la complejidad del modelo, el cuál presenta cuatro variables de entrada y doce variables de salida.

Por último, el **Capítulo 8**, *Conclusiones y desarrollos futuros*, incide en las conclusiones que se derivan de la Tesis. A continuación, se esbozan los temas que,

sustentados en desarrollos y propuestas de la Tesis, dejan el camino abierto para ser resueltos en trabajos futuros.

El **Apéndice I**, *Definición del Área de Solape*, se describen las cuatro expresiones del Área de Solape para cada uno de los casos.

En el **Apéndice II**, *Definición del Área de Solape para funciones de pertenencia gaussianas*, se detallan las expresiones del Área de Solape cuando las funciones de pertenencia son gaussianas para cada uno de los cuatro casos.

El **Apéndice III**, *Adaptación de cada parámetro implicado en el ajuste para funciones de pertenencia gaussianas*, describe para cada uno de los cuatro casos las expresiones para el ajuste de los centros y anchos en el caso de que las funciones de pertenencia sean gaussianas.

1.2. Innovaciones que aporta la Tesis

Los aspectos innovadores que pretende incorporar este trabajo en el campo del modelado de sistemas borrosos se encuentran en los capítulos 5, 6 y 7. En esta sección se pretende resumir cuáles son cada una de estas innovaciones en cada uno de los capítulos.

El **Capítulo 5**, *La Razón de Solape*, comienza con la definición de este concepto como medida del grado de solape entre dos funciones de pertenencia adyacentes. La definición inicial dada por Marsh [129] es ampliada en este trabajo para funciones gaussianas. El principal problema que se planteaba era calcular el soporte de una función gaussiana cuando el grado de pertenencia de éstas se hace cero teóricamente en el infinito. Se propone así el parámetro *Factor de Ancho* como medida que el diseñador puede utilizar para determinar cuando una función de pertenencia alcanza ese grado de pertenencia cero. El valor del Factor de Ancho es claramente 0.5 para funciones de pertenencia triangulares, pero el valor más adecuado para las funciones gaussianas debe ser analizado. Tras estudiar los valores que hacen que una función de pertenencia gaussiana alcance grados de pertenencia suficientemente cercanos a cero, se propone que este Factor de Ancho se encuentre en el intervalo [2, 2.5].

Se propone a continuación, como método de ajuste de los parámetros en el método de adaptación, el uso de la Razón de Solape, de forma que en cada época del

mecanismo de adaptación se ajusten los centros y anchos de las funciones de pertenencia teniendo en cuenta el solape que se produce entre cada dos funciones de pertenencia adyacentes.

Para aplicar correctamente el ajuste es necesario definir un nuevo concepto, la *Razón de Solape Deseada*. Este parámetro se refiere al valor de Razón de Solape que el diseñador considera adecuado entre dos funciones de pertenencia adyacentes. Para ayudar en la elección de este parámetro se presenta una gráfica que permite determinar el valor de la Razón de Solape Deseada una vez fijados los grados de pertenencia que el diseñador considera oportunos en la intersección de las dos funciones adyacentes y en el que la función se hace prácticamente cero.

El ajuste paramétrico no sólo debe ser aplicado a la parte precedente, sino que los consecuentes pueden ser distribuidos también adecuadamente en el universo de discurso utilizando la Razón de Solape como mecanismo de adaptación. Se propone una metodología totalmente novedosa para el caso de que los consecuentes sean un escalar (singleton) o de tipo TSK, ya que en ambos casos, el cálculo de la Razón de Solape entre dos consecuentes adyacentes no es posible, pues no se produce solape entre ellos.

Por último, se propone el algoritmo que permite la obtención de un modelo borroso a partir de datos de entrada-salida, adaptando los parámetros de las funciones de pertenencia y ajustándolos para que mantengan un adecuado solape en el universo de discurso.

El **Capítulo 6**, *El Área de Solape*, aporta un nuevo índice para medir el solape entre dos funciones de pertenencia adyacentes. Con el propósito de buscar una medida que se ajustara más al verdadero solape existente entre dos funciones se propone calcular el verdadero área de solape entre ambas para convertirlo en un nuevo parámetro que nos permita medir con mayor exactitud el solape. El nuevo índice depende del área de la menor de las funciones involucradas y de la posición que ésta ocupe en el universo de discurso. Esto nos genera 4 posibles casos. El trabajo se centra en estudiar la aplicación de este índice en funciones de pertenencia gaussianas, proponiendo el ajuste de los centros y anchos de las funciones en cada época del mecanismo de adaptación.

Para un correcto uso del ajuste mediante Área de Solape se requiere la introducción de un nuevo parámetro, el *Área de Solape Deseada*. Para ayudar al diseñador en la

elección del valor más adecuado para este parámetro se ha realizado un estudio que permite averiguar que si el Área de Solape Deseada se encuentra en el intervalo [0.2, 0.3], se obtienen los mejores resultados.

Se propone, además, un nuevo algoritmo que incorpora el ajuste de los parámetros involucrados en la adaptación utilizando el Área de Solape, para obtener un modelo borroso partiendo de datos de entrada-salida.

El **Capítulo 7, Aplicaciones**, aborda la estimación de modelos mediante sistemas borrosos en base a las técnicas presentadas en los capítulos anteriores. Se presentan tres ejemplos de aplicación, los dos primeros son modelos sencillos de dos entradas y una salida (MISO) y el último es un modelo mucho más complejo con 4 entradas y 12 salidas (MIMO).

En todos los casos, una vez realizada la estimación de la función, se comprueba mediante una comparativa, como la distribución de las funciones de pertenencia en el universo de discurso mejora considerablemente al aplicar las técnicas de ajuste propuestas en este trabajo, lo que conlleva a una mayor interpretabilidad del modelo. Se visualiza la distribución de las funciones para las distintas variables de entrada y para los consecuentes, comprobándose como la pérdida de precisión es asumible para poder ganar en interpretabilidad del modelo.

En conclusión, esta Tesis aporta una metodología completa de modelado, novedosa en muchos aspectos, que aporta interpretabilidad al mismo aunque sea perdiendo cierta precisión, y cuyo desarrollo se estudia paso a paso a lo largo de toda la memoria. Además, se aportan los algoritmos necesarios para su implementación práctica. Un aspecto importante de la metodología desarrollada es que permite su ampliación mediante la elaboración de nuevas técnicas que mejoren el ajuste de los parámetros y la visualización de los resultados obtenidos. Por ejemplo, cabría la posibilidad de permitir solapes máximos y mínimos para impedir excesivo solapamiento o solapamiento nulo, así como un etiquetado automático de la salida para una mayor interpretabilidad.

1.3. Rendimiento científico de la Tesis

Esta sección está dedicada a enmarcar el entorno científico en el que se ha desarrollado la Tesis y destacar los resultados de la investigación realizada; para ello se

detallarán las principales publicaciones realizadas. También, por supuesto, dentro del entorno científico de la Tesis, se hará hincapié en los hitos relevantes de su andadura. En definitiva, el objetivo principal de esta sección es mostrar que la memoria que se presenta constituye la recopilación ordenada de una producción científica realizada durante 10 años.

Producción científica por capítulos

La Tesis comenzó su andadura propiamente dicha, con el estudio del modelado mediante sistemas borrosos. El primer paso fue conseguir el algoritmo que permitía la estimación de una función sin aplicar ningún tipo de ajuste a los parámetros utilizando la metodología descrita por Wang [202]. Era importante la visualización normalizada de los resultados, permitiendo que se mostraran como quedaba la distribución de las variables de entrada y del consecuente tras la estimación.

El siguiente paso fue incorporar al algoritmo el ajuste de los parámetros utilizando la Razón de Solape, permitiendo que el diseñador pudiera ajustar el grado de solape con los parámetros Factor de Ancho y Razón de Solape Deseada. El estudio se centró en funciones de pertenencia gaussianas, por su probada capacidad para representar modelos más precisos. La comparativa entre los resultados obtenidos con este nuevo algoritmo y el primero desarrollado nos permitió comprobar como mejoraba considerablemente la interpretabilidad del modelo sin una pérdida significativa de la precisión. La técnica se aplicó exclusivamente a la parte precedente de las reglas y los resultados fueron presentados en las publicaciones:

Vélez, M.A. and Sánchez, O. *La razón de solape como mecanismo de adaptación paramétrica basada en restricciones para la obtención de modelos borrosos*. XXII Jornadas de Automática. Barcelona, 2001.

Vélez, M.A., Sánchez, O. and Ollero, A. *Neuro-Fuzzy modelling with parameters adjustment based on overlap ratio*. 10th Mediterranean Conference on Control and Automation. Lisboa, 2002.

Una vez comprobada la eficacia de la metodología utilizada en la parte precedente de las reglas, el siguiente paso fue aplicarla a los consecuentes. Se desarrolla un nuevo algoritmo que incorpora el ajuste de los consecuentes cuando estos son escalares o de tipo TSK. La aplicación de la técnica del solape cuando los consecuentes son un escalar

o de tipo TSK planteaba un problema, que se solucionó con la novedosa metodología presentada en la publicación:

Vélez, M.A., Andújar, J.M. and Sánchez, O. *Distribución uniforme de los consecuentes en el universo de discurso basado en la razón de solape*. Seminario Annual de Automática, Electrónica Industrial e Instrumentación. Alcalá de Henares, 2002.

Las técnicas de ajuste de los parámetros utilizando la Razón de Solape fueron motivo de colaboración en el trabajo:

Sánchez, O., Romero, S., Moreno, F.J. and Vélez, M.A. *Mathematical formulation of a type of hierarchical neurofuzzy system*. Lecture Notes in Computer Science. Vol. 4253, pp. 406-414, 2006.

Una vez terminado el trabajo tanto para la parte precedente como para los consecuentes se decidió emprender el estudio de un nuevo índice para medir el solape de las funciones, un índice que midiera con más exactitud el auténtico solape de dos funciones de pertenencia adyacentes. Se implementa un nuevo algoritmo que permita incorporar al proceso de ajuste el uso del Área de Solape en la parte precedente de las reglas, manteniendo la Razón de Solape como índice para el ajuste de los consecuentes.

El resumen completo de la Razón de Solape y el desarrollo del Área de Solape como nuevo índice para medir el ajuste de las funciones, así como los resultados obtenidos aplicándolo al modelo del horno de gas de Box y Jenkins [28] y al modelo de un helicóptero a escala de Kim y Tilbury [98] se presentó en la publicación más importante de la Tesis:

Vélez, M.A., Sánchez, O., Romero, S. and Andújar, J.M. *A new methodology to improve interpretability in neuro-fuzzy TSK models*. Applied Soft Computing. Vol. 10, pp. 578-591, 2010.

Índices de calidad: 21 de Computer Science, Artificial Intelligence en 2010 (2,415)

10 de Computer Science, Interdisciplinary Applications en 2010 (2,415)

2. Introducción a la lógica borrosa

2.1. Naturaleza e historia de la lógica borrosa

El comienzo de la lógica parece estar en Aristóteles (384-322 A.C). Sus principales contribuciones a la fundación de la lógica son dos axiomas llamados Ley de Contradicción y Ley de Exclusión Media. La Ley de Contradicción dice que una cosa no puede pertenecer a una clase y no pertenecer a esa clase al mismo tiempo. Es decir, no puede llover y no llover al mismo tiempo. La Ley de la Exclusión Media dice que una cosa debe pertenecer a una clase o no pertenecer a esa clase. En otras palabras, debe estar lloviendo o no estar lloviendo. Sin embargo, estos dos axiomas no dejan hueco para conceptos como llueve suavemente o fuerte lluvia.

Los dos axiomas de Aristóteles presentan la lógica con sólo dos colores, blanco y negro. Blanco representa verdad absoluta y negro representa falso completo. Sin embargo sólo con una gama extensa de grises, o grados de verdad, es posible capturar la profundidad y subjetividad del razonamiento humano.

La lógica de Aristóteles intentaba reducir todo argumento racional a la deducción de una secuencia de silogismos: “Todos los hombres son mortales. Sócrates es un hombre. Sócrates es mortal”.

Dos mil años después, en 1847, Boole desarrolló un método algebraico para el análisis de los silogismos de Aristóteles. El método es conocido hoy como Álgebra de Boole. Este método no sólo revelaba lo inadecuado del razonamiento silogístico, sino que se convertía en un punto de inicio para varios campos en matemáticas y ciencias de la computación.

En 1920 Jan Lukasiewicz publica un artículo describiendo una lógica trivaluada [123], cierto (1), falso (0) y posible (1/2) rompiendo la barrera de la lógica usual que sólo permitía dos posibles valores de verdad. Con esta expansión Lukasiewicz abrió la puerta a infinitas posibilidades, ya que sus tres grados de verdad podrían expandirse a un mayor número de niveles, dando lugar a la lógica multivaluada.

El siguiente paso, y realmente el más significativo, en el desarrollo de la lógica borrosa, se produjo en 1965 con el trabajo de Lotfi Zadeh [213] denominado *Fuzzy Sets*.

En su artículo, define las nociones matemáticas de inclusión, unión, intersección, complemento, relación, convexidad, etc. para los conjuntos borrosos y proporciona un marco teórico para representar el conocimiento expresado de forma lingüística, ofreciendo procedimientos de inferencia capaces de manejar la incertidumbre. Los conjuntos borrosos constituyen la base para manejar información carente de precisión a través de la extensión de los conceptos basados en la lógica convencional (lógica de Boole) a la lógica borrosa.

2.2. Aplicaciones

A pesar de su corta historia, la lógica borrosa tiene un crecimiento muy rápido, ya que consigue resolver problemas relacionados con la incertidumbre de la información o del conocimiento, proporcionando un método formal para la expresión del conocimiento en forma entendible y comprensible por los humanos. Gracias a esto, se puede asegurar un amplio campo de aplicaciones con un alto grado de interés.

A partir de la publicación, en 1973, de la teoría básica de los controladores borrosos de Zadeh [214], otros investigadores comenzaron a aplicar la lógica borrosa a diversos procesos, como por ejemplo, los profesores Mamdani y Assilian (1974), que experimentaron con éxito un controlador borroso en una máquina de vapor [128]. En 1977, King y Mamdani publican *The application of fuzzy control systems to industrial processes* [99], en el que describen cómo emplear la lógica borrosa como método de control para sistemas industriales complejos. También se puede resaltar la aplicación, en 1980 por la empresa F. L. Smidth & Co., de esta técnica al control de hornos rotativos en una cementera en Dinamarca.

Uno de los países donde más éxito han tenido los sistemas borrosos ha sido en Japón. En 1983, Fuji Electric aplica lógica borrosa para el control de inyección química para plantas depuradoras de agua, lo que les permitió adquirir los conocimientos necesarios para desarrollar el primer controlador borroso de propósito general FRUITAX (Fuzzy RULE Information processing Tool of Advanced control), lanzado en 1985, y que estaba constituido básicamente por un procesador de 16 bits junto con un sistema de inferencia borrosa basado en FORTRAN. Al año siguiente se produjo otro de los hitos más importantes en la historia de la lógica borrosa, ya que nació el primer chip VLSI para la realización de inferencias borrosas desarrollado por Togai y Watanabe

[186]. Los chips VLSI incrementaron en gran medida las posibilidades de elección de sistemas basados en reglas para aplicaciones en tiempo real.

En 1987, Yasunobu y sus colaboradores de la empresa Hitachi crearon una aplicación de control borroso para el sistema de frenado del metro de Sendai [209], Matsushita Electric (conocida fuera de Japón como Panasonic) el control borroso en unidades de suministro de agua caliente para uso doméstico. En 1989, el Ministerio de Comercio Internacional e Industria de Japón crea LIFE (Laboratory for International Fuzzy Engineering Research), el cual agrupaba a más de 50 compañías y poseía un presupuesto de 5.000 millones de yenes (unos 50 millones de dólares) para seis años [89].

La industria del automóvil es otro de los sectores donde se ha aplicado la lógica borrosa con mucho éxito [199][200][201]. Se han aplicado controladores borrosos en sistemas de suspensión activa [41][158][185], en sistemas de frenado [97][111][113][114][125][134] y en el control electrónico de motores [196]. Con la acuciante necesidad de eliminar la dependencia del petróleo, casi todos los fabricantes tienen abiertas líneas de investigación sobre vehículos híbridos. En este campo novedoso y complejo también está triunfando la lógica borrosa como un excelente método de control, y son muy numerosos los avances que se están consiguiendo gracias al control borroso [22][63][112][178][179][206].

Entre otras se pueden mencionar aplicaciones en el área médica, para el diagnóstico, desde campos como la acupuntura, hasta el análisis de los ritmos cardíacos o de la arterioestenosis coronaria. En la rama de la toma de decisiones se ha utilizado para la búsqueda de caminos críticos en la ejecución de proyectos y asesoramiento a la inversión. Para el control de sistemas en tiempo real destaca el control de un helicóptero por órdenes de voz y el control con derrapaje controlado de un modelo de coche de carreras. En el sector de la fabricación de electrodomésticos se han diseñado aplicaciones para lavadoras que evalúan la carga y ajustan por sí mismas el detergente necesario, la temperatura del agua y el tipo de ciclo de lavado (Matsushita, Hitachi, Siemens), tostadoras de pan, controles para calefacción y aire acondicionado, televisiones (Sony) que ajustan automáticamente el contraste, brillo y las tonalidades de color. Para el control de maquinaria, además del ya mencionado control de frenado de

metro de Sendai, se han utilizado para el control de una máquina de perforación de túneles, para el control de ascensores (Mitsubishi-Elec., Hitachi, Fuji Tech) y para grúas en el manejo y elevación de contenedores. Se conocen también aplicaciones para el procesado de imágenes y para el reconocimiento de caracteres, así como para verificadores de ortografía (proporcionando palabras probables para reemplazar una mal escrita) y en correctores de voz (sugiriendo palabras para sustituir una mal dicha).

En España la investigación sobre la lógica borrosa la impulsó el Profesor Enric Trillas, catedrático de Ciencias de la Computación e Inteligencia Artificial en la Universidad Politécnica de Madrid, quien ha sido galardonado con los tres premios más prestigiosos que se conceden en el ámbito de la lógica borrosa: el *Pioneer Award* de la European Society For Fuzzy Logic and Technologies (EUSFLAT), el *Fellow* de la International Fuzzy System Association (IFSA) y el *Fuzzy Systems Pioneer Award* otorgado por la IEEE Computational Intelligence Society. El profesor Trillas ha contribuido notablemente a convertir a España en uno de los núcleos de referencia en Europa en la actividad basada en la lógica borrosa al participar como autor y co-autor en más de doscientos artículos [6][7][8][38][39][143][157][188][190] y varios libros [189][191][192] [193].

En muchas universidades españolas, como por ejemplo, en la Universidad Complutense de Madrid, el grupo *Ingeniería de Sistemas, Control, Automática y Robótica (ISCAR)*, desarrolla numerosos trabajos y proyectos de investigación relacionados con temas de control, optimización, robótica y simulación [60][171][172][173][174][175][176][177]. También en Madrid, en la Universidad Politécnica, el grupo de investigación *Control Inteligente* (<http://www.disam.upm.es/control>) centra su investigación en robótica móvil y control de procesos [2][3][4][131][132][133], para lo que hacen uso de la lógica borrosa como herramienta. La Universidad de Valencia da cobijo a varios grupos de investigación, como el *Grupo de Control Predictivo y Optimización Heurística (CPOH)* (<http://ctl-predictivo.upv.es>), cuya investigación se centra en temas relacionados con el control predictivo basado en modelos y la optimización aplicada al control de procesos [167][168][169], y el grupo de *Control de Sistemas Complejos (GCSC)* (<http://www.isa.upv.es/investigacion/sistemas-complejos>), que centra sus estudios en el modelado y control de sistemas dinámicos que presentan incertidumbre, no linealidades

y distintos modos de comportamiento [24][25][26] [165]. El grupo de investigación de *Inteligencia Artificial e Ingeniería del Conocimiento (AIKE)* (<http://perseo.inf.um.es/~aike>) de la Universidad de Murcia, desarrolla actividades relacionadas con la aplicación de técnicas de ingeniería y gestión del conocimiento a dominios diversos como la medicina, la agricultura, la evaluación de recursos medioambientales o el diagnóstico y mantenimiento de equipos de plantas industriales, empleando la lógica borrosa como herramienta en muchas de sus investigaciones [21][31][64][65][66][76][92] [93][155]. En la Universidad Politécnica de Cataluña, el grupo de *Robótica Inteligente y Sistemas* (<http://grins.upc.es>), trabaja en el estudio, evaluación y desarrollo de sistemas robóticos [1][20][142].

No se pueden dejar de mencionar el *Instituto de Automática e Informática Industrial* (<http://www.ai2.upv.es>), de la Universidad de Valencia, ya que posee varias líneas de investigación sobre control de procesos [5][58][164][166] en las que aportan características inteligentes y predictivas a la consecución de soluciones tecnológicas empleando, entre otras herramientas, la lógica borrosa. La división de *Automatización y Control de Procesos* de la fundación CARTIF de Valladolid (<http://www.cartif.es>), que utiliza la lógica borrosa como herramienta para la consecución de sus objetivos de investigación aplicada, desarrollo tecnológico e innovación de procesos, productos y sistemas [94][50][161][162][163] y el *Instituto de Investigación en Inteligencia Artificial* (<http://www.iiia.csic.es>), perteneciente al Consejo Superior de Investigaciones Científicas que investiga sobre los fundamentos matemáticos de la lógica borrosa, el razonamiento aproximado y la elaboración de modelos conceptuales eficientes [9][10][27][61][62][68][73][74][187].

En la comunidad andaluza existen muchos grupos de investigación que trabajan con lógica borrosa, algunos tratando de ampliar y mejorar la teoría en torno a este campo y otros empleando la lógica borrosa como herramienta para la resolución de problemas complejos. Por citar algunos ejemplos, los grupos *Ingeniería Automática y Robótica* [15][16][77] y *Robótica, Visión y Control* (<http://grvc.us.es>) [18][46][47][48][75][130][149][150][151] de la Universidad de Sevilla, investigan sobre lógica borrosa y, fundamentalmente, sobre sus aplicaciones al campo de la robótica móvil. En Granada el grupo de investigación *Soft computing y sistemas de información inteligentes* (<http://sci2s.ugr.es>) posee múltiples líneas de investigación

sobre lógica borrosa y sus aplicaciones [42][45][51][80][81][82][83][84][85][86][87][88]. Entre estas líneas de investigación destacan sus trabajos sobre sistemas genético-borrosos, toma de decisiones mediante lingüística y Soft Computing. El grupo de investigación de *Inteligencia computacional* (<http://ic.ugr.es>) es otro grupo de investigación, con sede en la Universidad de Granada, que tiene como principal objetivo la investigación para el desarrollo de sistemas computacionales que estén en la vanguardia de la inteligencia artificial. Para ello, sus investigadores emplean y desarrollan, entre otras, técnicas basadas en lógica borrosa [32][33][34][35][36][37][40]. El grupo de investigación *ARAI, Razonamiento Aproximado e Inteligencia Artificial*, también de la Universidad de Granada, desarrolló, entre otros, sistemas de aprendizaje, de conocimiento, de soporte a la toma de decisiones y de reconocimiento de patrones basados en lógica borrosa [52][53][54][55][56][57]. En la Universidad de Málaga, el grupo de Ingeniería de Sistemas y Automática (<http://www.isa.uma.es>) también emplea la lógica borrosa en aplicaciones de control [15][72][120][149][160] y robótica [70][121][151].

Por último, mencionar que entre las líneas de investigación del grupo Control y Robótica de la Universidad de Huelva, se encuentran la modelización y simulación de procesos industriales, el aprendizaje automático y el control inteligente, empleando como una de las herramientas básicas de trabajo la lógica borrosa [11][12][13][14][17][49][78].

El nuevo reto de la lógica borrosa parece estar en Internet. Según algunos de los más prestigiosos investigadores en Internet, parece que el futuro para abordar la ingente cantidad de datos, recuperar información, controlar y gestionar la red, pasa por el uso de tecnologías borrosas. Además, parece que esta intuición coincide con la nueva senda que, según el profesor Zadeh, debe seguir la lógica borrosa. Prueba de ello fue la celebración del primer encuentro sobre lógica borrosa e Internet (FLINT 2001) celebrado en la Universidad de Berkeley y organizado por el propio profesor Zadeh.

La idea principal es la tendencia hacia el *Computing with words* (computación con palabras), usando esencialmente técnicas de *Soft Computing* (que engloba básicamente a la lógica borrosa, las redes neuronales y la computación evolutiva - algoritmos genéticos).

En resumen, la utilización de la lógica borrosa es aconsejable para procesos muy complejos, cuando se carece de un modelo matemático simple o para procesos altamente no lineales o si el procesamiento del conocimiento experto puede ser desempeñado.

Sin embargo, es mejor evitar su uso si el control convencional rinde un resultado satisfactorio, o cuando existe un modelo matemático fácilmente soluble y adecuado o también cuando el problema no tiene solución.

2.3. Soft Computing

2.3.1. Redes Neuronales

Las Redes de Neuronas Artificiales (RNA) o en inglés Artificial Neural Networks (ANN) son un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales. Se trata de un sistema de interconexión de neuronas en una red que colabora para producir un estímulo de salida. El objetivo es conseguir que las máquinas den respuestas similares a las que es capaz de dar el cerebro.

Una red neuronal se compone de unidades llamadas neuronas. Cada neurona recibe una serie de entradas a través de interconexiones y emite una salida. Esta salida viene dada por tres funciones:

Una **función de propagación** (función de excitación): por lo general consiste en la sumatoria de cada entrada multiplicada por el peso de su interconexión (valor neto). Si el peso es positivo, la conexión se denomina *excitadora*; si es negativo, se denomina *inhibitoria*.

Una **función de activación**: modifica a la anterior. Puede no existir, siendo en este caso la salida la misma función de propagación.

Una **función de transferencia**: Aplica el valor devuelto por la función de activación. Se utiliza para acotar la salida de la neurona. Las más utilizadas son la sigmoide (valores en el intervalo $[0,1]$) y la hiperbólica-tangente (valores en el intervalo $[-1,1]$).

Las RNA parten de un conjunto de datos de entrada suficientemente significativo y su objetivo es conseguir que la red definida a través de un modelo matemático aprenda automáticamente las propiedades deseadas. El proceso por el que los parámetros de la red se adecuan a la resolución de cada problema se denomina entrenamiento.

Por ejemplo, en una red que se va a aplicar al diagnóstico de imágenes médicas, durante la fase de entrenamiento el sistema recibe imágenes de tejidos que se sabe son cancerígenos y tejidos que se sabe son sanos, así como las respectivas clasificaciones de dichas imágenes. Si el entrenamiento es el adecuado, una vez concluido, el sistema podrá recibir imágenes de tejidos no clasificados y obtener su clasificación sano/no sano con un buen grado de seguridad.

Las RNA son bastante apropiadas para aplicaciones en las que no se dispone a priori de un modelo identificable que pueda ser programado, pero se dispone de un conjunto básico de ejemplos de entrada (previamente clasificados o no). Además, son altamente robustas tanto al ruido como a la disfunción de elementos concretos y son fácilmente paralelizables. Como ejemplo, se suelen utilizar en problemas de clasificación y reconocimiento de patrones de voz, imágenes, señales, etc. Se utilizan, además, para encontrar patrones de fraude económico, hacer predicciones de mercado financiero, hacer predicciones de tiempo atmosférico, etc.

También se pueden utilizar cuando no existen modelos matemáticos precisos o algoritmos con complejidad razonable. Por ejemplo, la red de Kohonen ha sido aplicada con éxito al clásico problema del viajante (problema para el que no se conoce solución algorítmica de complejidad polinómica).

Funciones de Base Radial

Dentro de los modelos neuronales multicapas, para realizar ajustes de funciones no lineales tenemos las Funciones de Base Radial (FBR), Radial Basis Function (RBF) en inglés. A diferencia de los Modelos Perceptron Multicapa (MPM), las FBR construyen sus modelos con funciones de activación que son diferentes tanto en la capa oculta como en la de salida. Las neuronas de la capa oculta se activan mediante funciones radiales de carácter no lineal y la capa de salida mediante funciones lineales.

Una red FBR está conformada por tres capas:

La **capa de entrada** que sirve para los ejemplos o patrones de entrenamiento y prueba.

La **capa oculta** completamente interconectada entre todos sus nodos con la capa de entrada y activada a través de la función radial (Gaussiana).

La **capa de salida**, también completamente interconectada a la capa oculta y activada a través de una función lineal continua.

De este modo, la salida z de una red FBR, en general, está influenciada por una transformación no lineal originada en la capa oculta a través de la función radial y una lineal en la capa de salida a través de la función lineal continua.

Los nodos ocultos contienen una función de base radial, la cual tiene como parámetros centros y anchos (ver Figura 2.1).

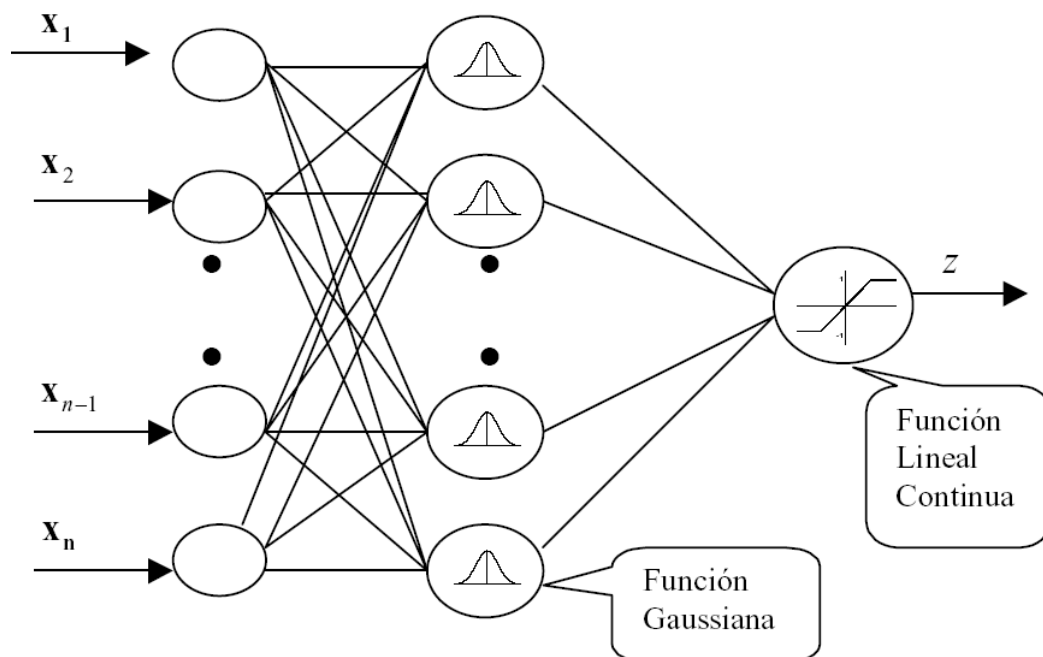


Figura 2.1 Esquema básico de una Red de Base Radial

Existe un centro para cada función radial involucrada en la capa oculta. Regularmente, definen un vector de la misma dimensión del vector de entrada y hay normalmente un centro diferente por cada nodo de la capa oculta.

Por otro lado, el ancho es el término empleado para identificar la amplitud de la campana de gauss originada por la función radial. Es decir, la desviación estándar de la función radial. Algunos autores [122] consideran este ancho como un valor constante para cada una de las funciones radiales consideradas en la capa oculta y de este modo, así contribuiría a simplificar los pasos de construcción del modelo de entrenamiento de la red.

El primer cálculo efectuado en la capa oculta es hallar en un nodo de la capa oculta la distancia radial (distancia euclídea) d entre el vector de entrada x , con n observaciones, a ese nodo en particular y el centro de gravedad c de ese mismo nodo. Es decir:

$$d = \|x - c\| = \sqrt{(x_1 - c_1)^2 + (x_2 - c_2)^2 + \dots + (x_n - c_n)^2} \quad (2.1)$$

Este valor d es un componente de la entrada para activar la función radial $G(\bullet)$. Este valor establece la principal diferencia con las redes MPM, quienes incluyen el producto interno en sus capas ocultas de las entradas por sus respectivos pesos.

Entre la capa oculta y la capa de salida se derivan un conjunto de pesos w que se verían afectados de acuerdo al algoritmo de aprendizaje. En este caso particular, sería la combinación lineal entre los pesos y la resultante de cada función radial para determinar la salida z .

$$\text{Por lo tanto, sería, } z = \sum_{i=1}^p w_i G(\bullet) \quad (2.2)$$

donde $G(\bullet)$ es la salida de la capa oculta y se corresponde con la función radial aplicada a la distancia euclídea en cada una de las unidades ocultas (ver Figura 2.2).

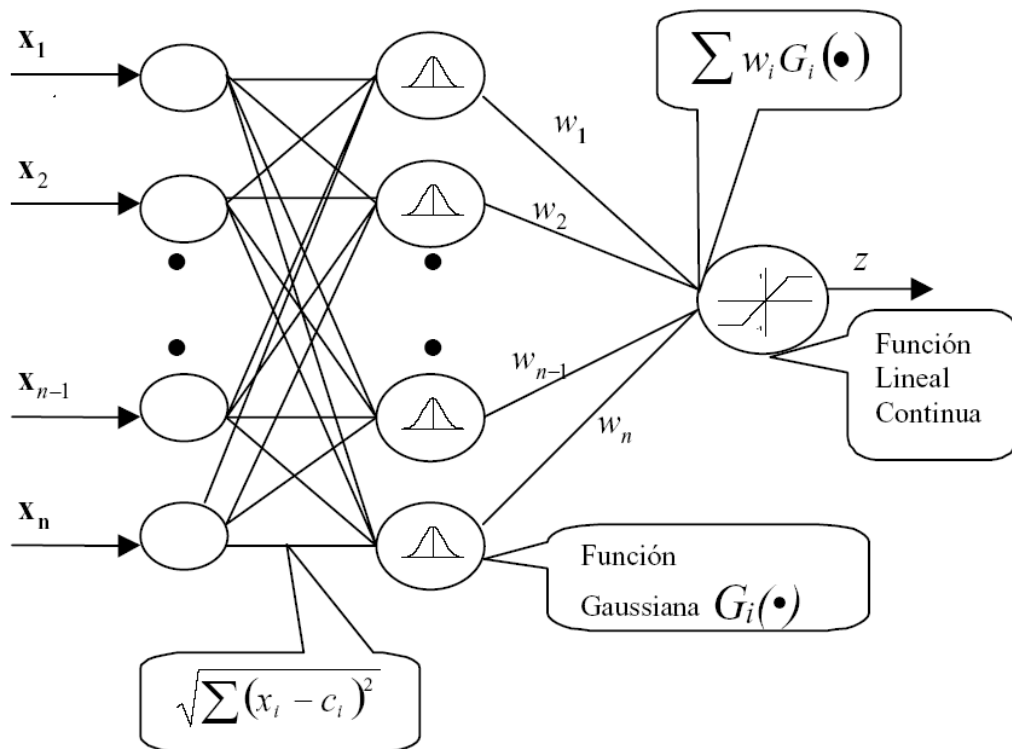


Figura 2.2 Red de Base Radial aplicando distancia euclídea y pesos

En [181] se propone una Función en Base Radial Gaussiana Normalizada; la salida del sistema viene dada por¹:

$$z = \frac{\sum_{i=1}^p w_i G_i(X)}{\sum_{j=1}^p G_j(X)} \quad (2.3)$$

Nótese como (2.3) es semejante a (4.4); por ello, en cualquier aplicación donde se utiliza la función en base radial, puede plantearse un modelo borroso equivalente.

¹ En Shao , Lee y Jones (1993) la salida es equivalente a la proyección consecuente del método TSK que se verá posteriormente; no obstante, considerar la salida del TSK como término independiente da lugar a la misma expresión que aquí se expone.

2.3.2. Algoritmos genéticos

Un Algoritmo Genético (AG) es una técnica de programación que imita a la evolución biológica como estrategia para resolver problemas. Dado un problema específico a resolver, la entrada del AG es un conjunto de soluciones potenciales a ese problema, codificadas de alguna manera, y una métrica llamada función de aptitud que permite evaluar cuantitativamente a cada candidata. Estas candidatas pueden ser soluciones que ya se sabe que funcionan, con el objetivo de que el AG las mejore, pero se suelen generar aleatoriamente.

Luego, el AG evalúa cada candidata de acuerdo con la función de aptitud. En un conjunto de candidatas generadas aleatoriamente, la mayoría no funcionarán en absoluto, y serán eliminadas. Sin embargo, por puro azar, unas pocas pueden ser prometedoras, es decir, pueden mostrar actividad, aunque sólo sea actividad débil e imperfecta, hacia la solución del problema.

Estas candidatas prometedoras se conservan y se les permite reproducirse. Se realizan múltiples copias de ellas, pero las copias no son perfectas; se introducen cambios aleatorios durante el proceso de copia. Luego, esta descendencia digital prosigue con la siguiente generación, formando un nuevo conjunto de soluciones candidatas, y son sometidas a una ronda de evaluación de aptitud. Las candidatas que han empeorado o no han mejorado con los cambios en su código son eliminadas de nuevo; pero, de nuevo, por puro azar, las variaciones aleatorias introducidas en la población pueden haber mejorado a algunos individuos, convirtiéndolos en mejores soluciones del problema, más completas o más eficientes. De nuevo, se seleccionan y copian estos individuos vencedores hacia la siguiente generación con cambios aleatorios, y el proceso se repite. Las expectativas son que la aptitud media de la población se incrementará en cada ronda y, por tanto, repitiendo este proceso cientos o miles de rondas, pueden descubrirse soluciones razonables del problema.

Representación: Antes de que un algoritmo genético pueda aplicarse a un problema, se necesita un método para codificar las soluciones potenciales del mismo, de forma que una computadora pueda procesarlas. Un enfoque común es codificar las soluciones como cadenas binarias: secuencias de 1s y 0s, donde el dígito de cada posición representa el valor de algún aspecto de la solución. Otro método similar

consiste en codificar las soluciones como cadenas de enteros o números decimales, donde cada posición, de nuevo, representa algún aspecto particular de la solución. Este método permite una mayor precisión y complejidad que el método comparativamente restringido de utilizar sólo números binarios.

Selección: Un algoritmo genético puede utilizar muchas técnicas diferentes para seleccionar a los individuos que deben copiarse hacia la siguiente generación, pero abajo se listan algunos de los más comunes. Algunos de estos métodos son mutuamente exclusivos, pero otros pueden utilizarse en combinación, algo que se hace a menudo.

- *Selección elitista:* se garantiza la selección de los miembros más aptos de cada generación. (La mayoría de los AGs no utilizan elitismo puro, sino que usan una forma modificada por la que el individuo mejor, o algunos de los mejores, son copiados hacia la siguiente generación en caso de que no surja nada mejor).
- *Selección proporcional a la aptitud:* los individuos más aptos tienen más probabilidad de ser seleccionados, pero no la certeza.
- *Selección por rueda de ruleta:* una forma de selección proporcional a la aptitud en la que la probabilidad de que un individuo sea seleccionado es proporcional a la diferencia entre su aptitud y la de sus competidores. (Conceptualmente, esto puede representarse como un juego de ruleta -cada individuo obtiene una sección de la ruleta, pero los más aptos obtienen secciones mayores que las de los menos aptos. Luego la ruleta se hace girar, y en cada vez se elige al individuo que "posea" la sección en la que se pare la ruleta).
- *Selección escalada:* al incrementarse la aptitud media de la población, la fuerza de la presión selectiva también aumenta y la función de aptitud se hace más discriminadora. Este método puede ser útil para seleccionar más tarde, cuando todos los individuos tengan una aptitud relativamente alta y sólo les distinguen pequeñas diferencias en la aptitud.
- *Selección por torneo:* se eligen subgrupos de individuos de la población, y los miembros de cada subgrupo compiten entre ellos. Sólo se elige a un individuo de cada subgrupo para la reproducción.

- *Selección por rango:* a cada individuo de la población se le asigna un rango numérico basado en su aptitud, y la selección se basa en este ranking, en lugar de las diferencias absolutas en aptitud. La ventaja de este método es que puede evitar que individuos muy aptos ganen dominancia al principio a expensas de los menos aptos, lo que reduciría la diversidad genética de la población y podría obstaculizar la búsqueda de una solución aceptable.
- *Selección generacional:* la descendencia de los individuos seleccionados en cada generación se convierte en toda la siguiente generación. No se conservan individuos entre las generaciones.
- *Selección por estado estacionario:* la descendencia de los individuos seleccionados en cada generación vuelven al acervo genético preexistente, reemplazando a algunos de los miembros menos aptos de la siguiente generación. Se conservan algunos individuos entre generaciones.
- *Selección jerárquica:* los individuos atraviesan múltiples rondas de selección en cada generación. Las evaluaciones de los primeros niveles son más rápidas y menos discriminatorias, mientras que los que sobreviven hasta niveles más altos son evaluados más rigurosamente. La ventaja de este método es que reduce el tiempo total de cálculo al utilizar una evaluación más rápida y menos selectiva para eliminar a la mayoría de los individuos que se muestran poco o nada prometedores, y sometiendo a una evaluación de aptitud más rigurosa y computacionalmente más costosa sólo a los que sobreviven a esta prueba inicial.

Cambios: Una vez que la selección ha elegido a los individuos aptos, éstos deben ser alterados aleatoriamente con la esperanza de mejorar su aptitud para la siguiente generación. Existen dos estrategias básicas para llevar esto a cabo. La primera y más sencilla se llama mutación. Al igual que una mutación en los seres vivos cambia un gen por otro, una mutación en un algoritmo genético también causa pequeñas alteraciones en puntos concretos del código de un individuo.

El segundo método se llama cruzamiento, e implica elegir a dos individuos para que intercambien segmentos de su código, produciendo una *descendencia* artificial cuyos individuos son combinaciones de sus padres. Este proceso pretende simular el

proceso análogo de la recombinación que se da en los cromosomas durante la reproducción sexual. Las formas comunes de cruzamiento incluyen al cruzamiento de un punto, en el que se establece un punto de intercambio en un lugar aleatorio del genoma de los dos individuos, y uno de los individuos contribuye todo su código anterior a ese punto y el otro individuo contribuye todo su código a partir de ese punto para producir una descendencia, y al cruzamiento uniforme, en el que el valor de una posición dada en el genoma de la descendencia corresponde al valor en esa posición del genoma de uno de los padres o al valor en esa posición del genoma del otro padre, elegido con un 50% de probabilidad.

Aunque pueda parecer asombroso y antiintuitivo, los algoritmos genéticos han demostrado ser una estrategia enormemente poderosa y exitosa para resolver problemas, demostrando de manera espectacular el poder de los principios evolutivos. Se han utilizado algoritmos genéticos en una amplia variedad de campos para desarrollar soluciones a problemas tan difíciles o más difíciles que los abordados por los diseñadores humanos [30][90][108]. Además, las soluciones que consiguen son a menudo más eficientes, más elegantes o más complejas que nada que un ingeniero humano produciría.

2.3.3. ¿Por qué lógica borrosa?

Existen varias razones para justificar el uso de la lógica borrosa frente a métodos como redes neuronales o algoritmos genéticos:

1. Los parámetros del sistema adaptativo borroso tienen un claro significado físico, lo que facilita la selección de las condiciones iniciales de los parámetros. Esto no es así en las redes neuronales debido a que se seleccionan de manera aleatoria.
2. Basado en este significado físico que tienen los parámetros se puede verificar si como resultado de la adaptación paramétrica, las posiciones y formas de las funciones de pertenencia son correctas.
3. El diseñador podría actuar incorporándole información lingüística (añadiendo nuevas funciones de pertenencia en el universo de discurso, desplazando alguna de ellas, etc.).

2.4. Conclusiones

A lo largo del capítulo se ha visto como la lógica borrosa se ha ganado el reconocimiento del que goza en la actualidad en base a los excelentes resultados que ha obtenido, tanto en el ámbito académico e investigador como en el industrial. Con los investigadores japoneses al frente desde casi los inicios y la posterior incorporación de investigadores del resto del mundo, la lógica borrosa se ha convertido hoy en día en una tecnología de vanguardia, útil y eficaz en muchas aplicaciones de muy diversos ámbitos. El desarrollo inicial de la lógica borrosa, liderado fundamentalmente por su creador, Lofti A. Zadeh, puede consultarse en las referencias [213] a [219], mientras que el proceso de formalización y adaptación de esta teoría al ámbito industrial y de consumo puede documentarse en las referencias [89][128][186][209] y [199][200][201][206], respectivamente. También se han recogido algunos de los trabajos realizados por investigadores españoles sobre lógica borrosa.

Se han resumido brevemente las técnicas de Soft Computing y se ha justificado el uso de la lógica borrosa como la técnica adecuada para solucionar el problema abordado en esta Tesis.

De entre las referencias incluidas en este capítulo, cabe destacar por su especial relevancia el primer trabajo del profesor Zadeh [213], que dio inicio a la lógica borrosa.

3. Conceptos básicos de sistemas borrosos

La lógica borrosa trabaja con reglas de razonamiento muy cercanas a la forma de pensar de los seres humanos, que es aproximada e intuitiva. La principal característica de la lógica borrosa es que permite definir valores sin concretar un valor preciso, lo que no es posible con la lógica clásica en la que se ha basado hasta ahora el desarrollo de la informática. En la lógica clásica, la pertenencia a una clase o conjunto es binaria, es decir, se pertenece o no se pertenece a él, por tanto, se trabaja sólo con dos valores precisos (1 y 0, sí o no). Así por ejemplo, si se define la clase personas altas, está claro que un hombre que mida 1.90 metros pertenece al conjunto y otro que mida 1.60 metros no pertenece, pero ¿cómo clasificamos a un hombre que mida 1.79 metros? En la respuesta a este tipo de preguntas es donde precisamente la lógica clásica falla, ya que los umbrales no están bien definidos. Sin embargo, en la forma de razonar de los seres humanos, la lógica humana, no hay mayor problema en trabajar con incertidumbre léxica, ya que es inherente al lenguaje humano.

3.1. Conjuntos borrosos

Un conjunto borroso A de un conjunto X puede definirse como un conjunto de pares ordenados, con el primer elemento que pertenezca a X y el segundo elemento perteneciente al intervalo $[0,1]$, con un par ordenado por cada elemento de X . Ello define una aplicación, μ_A , entre elementos del conjunto X y los valores del intervalo $[0,1]$. El valor cero se utiliza para representar la no-pertenencia y el valor uno representa la completa pertenencia. Los valores intermedios representan un *grado de pertenencia* parcial. Al conjunto X se le denomina *universo de discurso* para el conjunto borroso A . El grado de certeza de la afirmación “ x pertenece a A ” depende de la definición del par ordenado $(x, \mu_A(x))$, específicamente del segundo elemento del par ordenado $\mu_A(x)$; de forma tal que mientras más próximo a uno sea este, mayor será el grado de veracidad de la afirmación.

Sea X un conjunto no vacío. Un conjunto borroso A está caracterizado por una función de pertenencia:

$$\mu_A : X \rightarrow [0,1] \quad (3.1)$$

donde la función de pertenencia $\mu_A(x)$ representa un determinado grado de pertenencia del elemento x en el conjunto borroso A para cada $x \in X$.

El conjunto borroso A está completamente determinado por el conjunto de pares:

$$A = \{(x, \mu_A(x)) | x \in X\} \quad (3.2)$$

A la familia de todos los conjuntos borrosos definidos en X se denota por $\mathfrak{F}(X)$.

Un conjunto borroso puede representarse de forma continua o de forma discreta. Si la representación es continua, la forma geométrica de las funciones de pertenencia es totalmente arbitraria, pero en general se suelen utilizar funciones de geometría sencilla y ecuación conocida, como trapecios, triángulos o sigmoides.

Por ejemplo, un conjunto borroso, definido a través de una función que represente la sentencia “cercano a 1”, puede ser definida como:

$$\mu_A(x) = \exp(-\beta(x-1)^2) \quad (3.3)$$

donde β es un número real positivo (ver Figura 3.1).

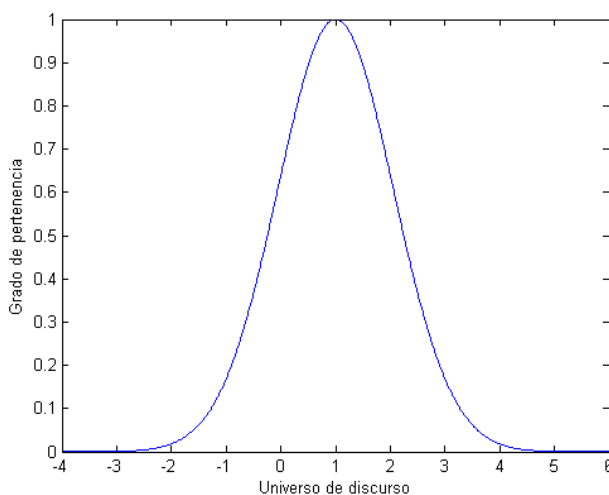


Figura 3.1 Representación de un conjunto borroso continuo para la sentencia “ x es cercano a 1”

En el caso de realizar una representación discreta de ese conjunto borroso, se ha de enumerar el grado de pertenencia para cada elemento del conjunto. Así, dado el conjunto borroso discreto A definido en su correspondiente universo de discurso X , su representación es de la forma:

$$A = \sum_{i=1}^n \mu_A(x_i) / x_i; \quad x_i \in X \quad (3.4)$$

donde el término $\mu_A(x_i) / x_i$ representa un conjunto de pares (grado de pertenencia del elemento x_i en el conjunto A , elemento x_i) y la sumatoria la operación de unión (no la suma aritmética).

En la Figura 3.2 se muestra como ejemplo el conjunto borroso discreto dado por:

$$A = 0,0/-2 + 0,3/-1 + 0,6/0 + 1,0/1 + 0,6/2 + 0,3/3 + 0,0/4 \quad (3.5)$$

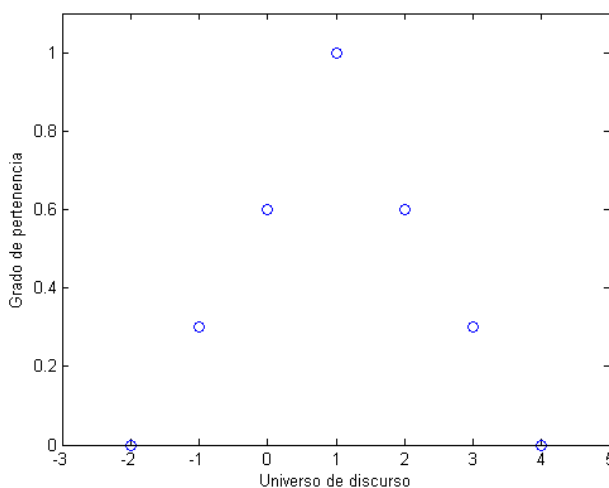


Figura 3.2 Representación de un conjunto borroso discreto para la sentencia “x es cercano a 1”

Sea A un conjunto borroso definido en X . El **soporte** del conjunto borroso A es el conjunto de elementos del universo de discurso X cuyo grado de pertenencia es mayor que cero:

$$sop(A) = \{x \in X \mid \mu_A(x) > 0\} \quad (3.6)$$

Por ejemplo, el soporte del conjunto borroso definido en (3.5) y que se representa en la Figura 3.2 es:

$$\text{sop}(A) = -1, 0, 1, 2, 3 \quad (3.7)$$

Cuando el soporte de un conjunto borroso contiene un único elemento ese conjunto borroso se denomina **singleton o punto borroso**. A la función de pertenencia singleton se le denomina conjunto borroso escalar (ver la Figura 3.3) y juegan un papel de vital importancia cuando se desea obtener modelos borrosos a partir de datos de entrada-salida.

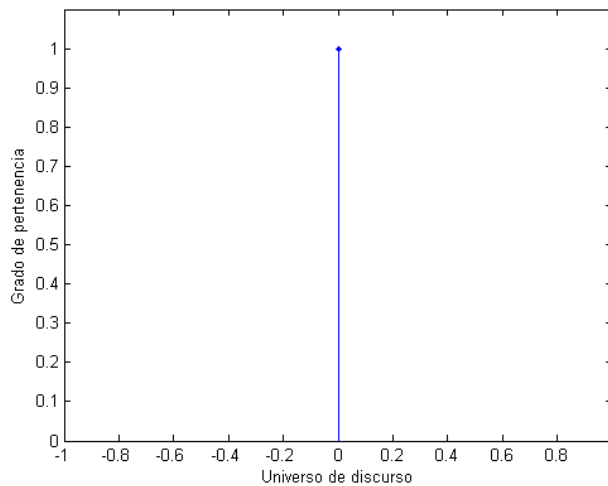


Figura 3.3 Representación de un singleton ($A=0$)

Sea A un conjunto borroso definido en X . Se dice que A es normal si existe un $x \in X$ tal que $\eta_A(x) = 1$. De otra forma A es no-normal. En la Figura 3.4 se muestra la diferencia entre un conjunto borroso normal y uno no normal.

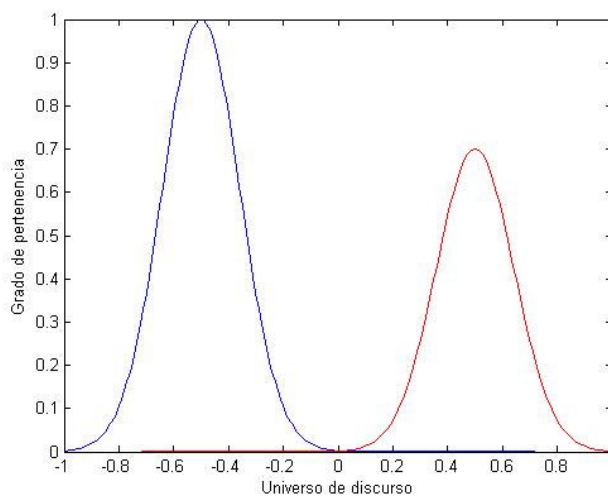


Figura 3.4 Conjunto borroso normal y conjunto borroso no normal

Sea A un conjunto borroso definido en X , se dice que A es un conjunto borroso normalizado si y sólo si $0 \leq \mu_A(x) \leq 1$.

Dado que un grado de pertenencia mayor que uno no tiene mucho sentido lógico, normalmente siempre se emplean conjuntos borrosos normalizados, aunque no se mencione explícitamente.

Sea A un conjunto borroso definido en X , se dice que A es un conjunto borroso convexo si y sólo si para dos puntos cualesquiera x_1 y x_2 , y cualquier $\lambda \in [0,1]$, se verifica la ecuación (3.8). Nótese que esta definición no implica que deba ser una función convexa.

$$\mu_A(\lambda x_1 + (1-\lambda)x_2) \geq \min(\mu_A(x_1), \mu_A(x_2)) \quad (3.8)$$

Esta definición indica que, dados dos puntos del universo de discurso, la función de pertenencia de un conjunto borroso convexo evaluada entre esos puntos tomará valores mayores o iguales que la función de pertenencia evaluada en cada uno de dichos puntos.

Sea α un número entre 0 y 1. Se denomina **α -corte** de un conjunto borroso A , y se denota como A_α al conjunto de elementos cuyo grado de pertenencia en A es mayor o igual que α . De forma matemática, el α -corte del conjunto borroso A en X se define como:

$$A_\alpha = \{x \in X \mid \mu_A(x) \geq \alpha\} \quad (3.9)$$

La Figura 3.5 representa un conjunto borroso con un α -corte=0.5.

Un **número borroso** es un conjunto borroso cuyo universo de discurso es el eje real. Los números borrosos son un tipo de conjunto borroso que poseen un claro significado cuantitativo, en la medida en que clasifican un concepto alrededor de un número o intervalo de números.

Definición de número cuasi-borroso.

Como se expone más adelante, un número borroso (o cuasi-borroso) puede ser representado a través de una **función de pertenencia**.

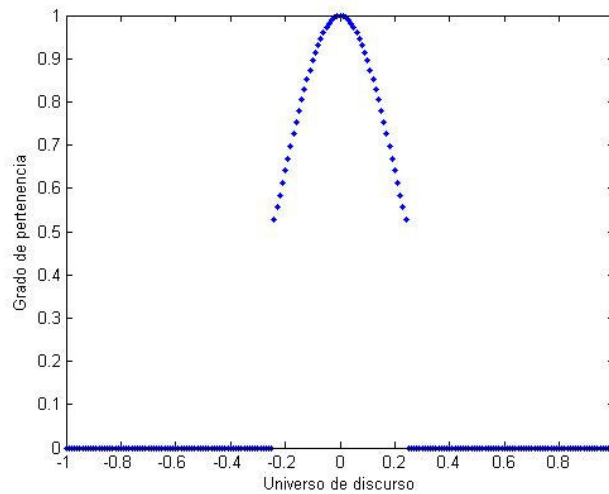


Figura 3.5 Representación de un conjunto borroso
con un α -corte=0.5

3.2. Funciones de pertenencia

La definición de un conjunto borroso de forma cuantitativa, es decir, mediante una expresión matemática, se realiza a través de una función de pertenencia. Un conjunto borroso está completamente caracterizado por su función de pertenencia. Las funciones de pertenencia más utilizadas para representar conjuntos borrosos suelen ser:

Triangular

Esta función de pertenencia se caracteriza por tres parámetros: el centro a , y los anchos izquierdo α y derecho β , de forma que un conjunto borroso A es triangular si:

$$\mu_A(x) = \begin{cases} 1 - (a - x) / \alpha & \text{si } a - \alpha \leq x \leq a \\ 1 - (x - a) / \beta & \text{si } a \leq x \leq a + \beta \\ 0 & \text{en otro caso} \end{cases} \quad (3.10)$$

Se usará la notación $\mu_A = (a, \alpha, \beta)$. En la Figura 3.6 se muestra la representación de la función de pertenencia $\mu_A = (0.2, 0.8, 0.5)$.

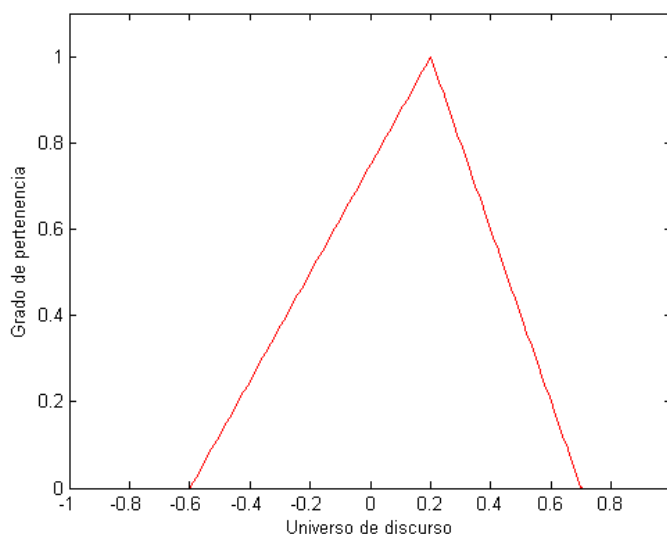


Figura 3.6 Representación de la función de pertenencia triangular $\mu_A = (0.2, 0.8, 0.5)$

Trapezoidal

La función de pertenencia trapezoidal se caracteriza por cuatro parámetros: vértice izquierdo a , vértice derecho b , ancho izquierdo α y ancho derecho β . Un conjunto borroso A es trapezoidal si:

$$\mu_A(x) = \begin{cases} 1 - (a - x) / \alpha & \text{si } a - \alpha \leq x \leq a \\ 1 & \text{si } a \leq x \leq b \\ 1 - (x - b) / \beta & \text{si } b \leq x \leq b + \beta \\ 0 & \text{en otro caso} \end{cases} \quad (3.11)$$

Se usará la notación $\mu_A = (a, b, \alpha, \beta)$. En la Figura 3.7 se muestra la representación de la función de pertenencia $\mu_A = (-0.3, 0.4, 0.6, 0.4)$.

Gaussiana

La función de pertenencia gaussiana se caracteriza por dos parámetros: centro a y ancho α , de forma que un conjunto borroso A es gaussiano si:

$$\mu_A(x) = e^{\left(-\left(\frac{x-a}{\alpha}\right)^2\right)} \quad (3.12)$$

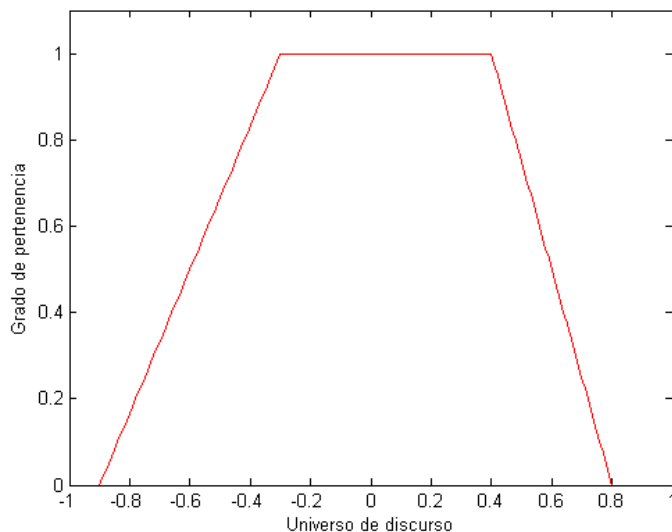


Figura 3.7 Representación de la función de pertenencia trapezoidal $\mu_A = (-0.3, 0.4, 0.6, 0.4)$

Se denotará como $\mu_A = (a, \alpha)$. En la Figura 3.8 se puede apreciar la representación de la función de pertenencia $\mu_A = (0, 0.2)$

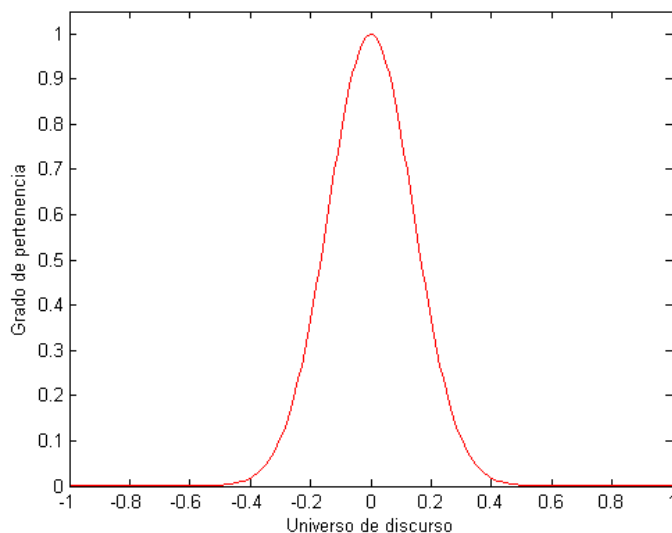


Figura 3.8 Representación de la función de pertenencia gaussiana $\mu_A = (0, 0.2)$

Sigmoidal

Una forma de representar fronteras suaves es mediante la función de pertenencia sigmoidal. Esta función depende de los parámetros a (punto para el que la función vale 0.5) y α (parámetro relacionado con la velocidad de transición de la función de pertenencia) tal y como se muestra en (3.13):

$$\mu_A(x) = \frac{1}{1 + e^{\alpha(a-x)}} \quad (3.13)$$

Dependiendo del signo del parámetro α , la función de pertenencia sigmoidal es abierta a la izquierda o a la derecha, con lo cual es muy adecuada para representar extremos en el universo de discurso de las variables. Si los parámetros a y α aumentan, la zona de transición de la función de pertenencia se hace más vertical. Se denotará como $\mu_A = (a, \alpha)$.

En la Figura 3.9 podemos apreciar un ejemplo de la función sigmoidal $\mu_A = (0, -4)$. Nótese como para cualquier función sigmoidal $\mu_A(a) = 0.5$.

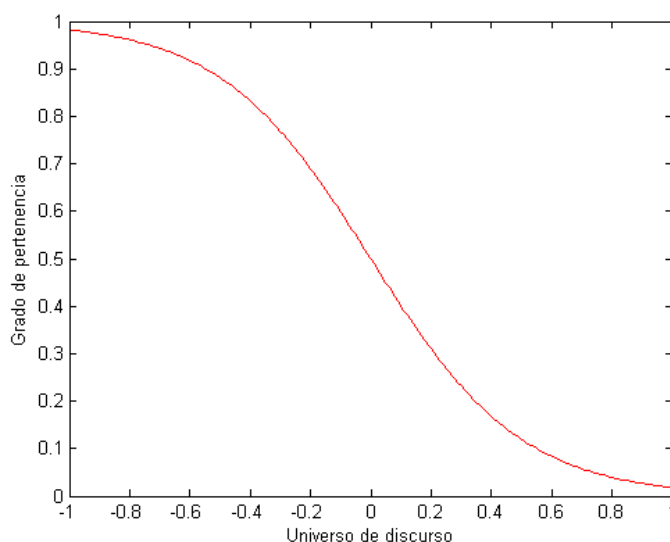


Figura 3.9 Representación de la función de pertenencia sigmoidal $\mu_A = (0, -4)$

Campana

Una función de pertenencia campana se define según la expresión (3.14), siendo a el centro de la función de pertenencia, y $\alpha > 0$ y $\beta \neq 0$ dos parámetros de ajuste. La denotaremos como $\mu_A = (a, \alpha, \beta)$. El parámetro α tiene relación con el ancho de la función por su parte baja, mientras que el parámetro β tiene relación con el ancho de la función por la zona alta. Si $\beta < 0$, la función de pertenencia toma la forma de una campana invertida.

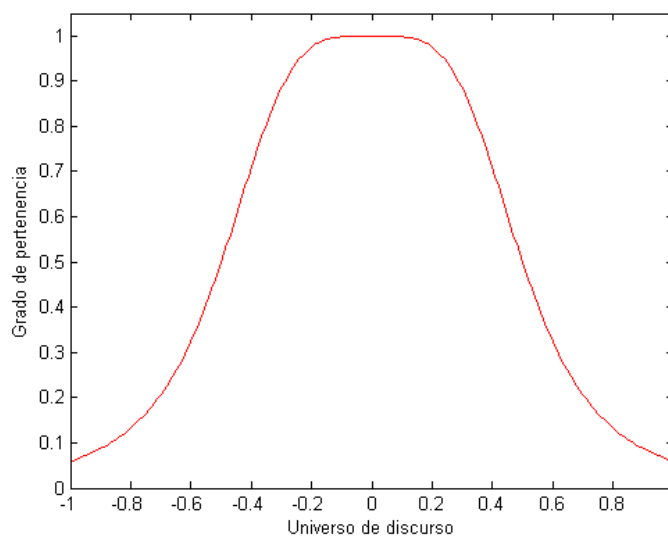


Figura 3.10 Representación de la función de pertenencia campana $\mu_A = (0, 0.5, 2)$

$$\mu_A(x) = \frac{1}{1 + \left| \frac{x-a}{\alpha} \right|^{2\beta}} \quad (3.14)$$

En la Figura 3.10 podemos ver una función de pertenencia campana $\mu_A = (0, 0.5, 2)$.

S

La función de pertenencia S, permite establecer una transición suave desde un valor de pertenencia no nulo a un valor de pertenencia uno. Se define mediante la expresión (3.15), siendo $a \leq b$. Su soporte viene dado por el intervalo (a, ∞) , por lo que es infinito. Se trata de una función muy utilizada para representar etiquetas lingüísticas del tipo *mayor de*.

$$\mu_A(x) = \begin{cases} 0 & \text{si } x \leq a \\ 2\left(\frac{x-a}{b-a}\right)^2 & \text{si } a < x \leq \frac{a+b}{2} \\ 1-2\left(\frac{b-x}{b-a}\right)^2 & \text{si } \frac{a+b}{2} < x < b \\ 1 & \text{si } x \geq b \end{cases} \quad (3.15)$$

En la Figura 3.11 se muestra un ejemplo de función S, con a= -0.2 y b= 0.5.

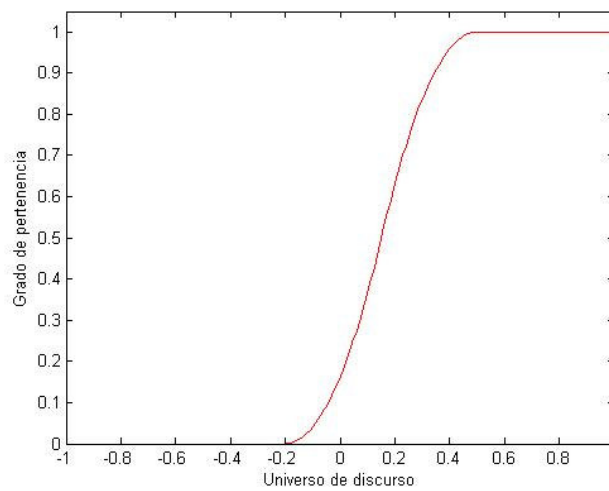


Figura 3.11 Representación de la función de pertenencia S

$$\mu_A = (-0.2, 0.5)$$

Z

La función de pertenencia Z, siendo a ≤ b, permite establecer una transición suave desde un valor de pertenencia uno a un valor de pertenencia nulo. Se define según la expresión (3.16).

$$\mu_A(x) = \begin{cases} 1 & \text{si } x \leq a \\ 1-2\left(\frac{x-a}{a-b}\right)^2 & \text{si } a < x < \frac{a+b}{2} \\ 2\left(\frac{b-x}{a-b}\right)^2 & \text{si } \frac{a+b}{2} \leq x < b \\ 0 & \text{si } x \geq b \end{cases} \quad (3.16)$$

Se trata de una función continua y convexa. Su soporte es $(-\infty, b)$, por lo que es infinito. Es una función muy utilizada para representar las etiquetas lingüísticas del tipo *menor de*.

En la Figura 3.12 se muestra un ejemplo de función Z, $a = -0.2$ y $b = 0.5$.

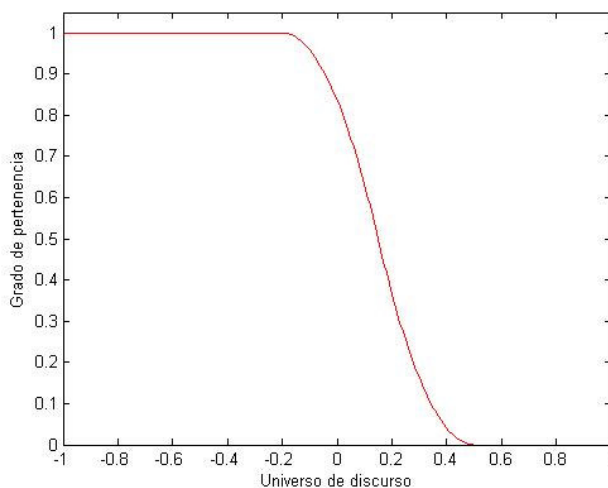


Figura 3.12 Representación de la función de pertenencia Z

$$\mu_A = (-0.2, 0.5)$$

3.3. Operaciones sobre conjuntos borrosos

Muchas de las operaciones presentes en la lógica clásica de conjuntos pueden generalizarse a la lógica borrosa, como por ejemplo las tres operaciones básicas: unión, intersección y complemento; sin embargo, hay operaciones y propiedades de la lógica clásica que no se cumplen al trasladarlas al ámbito borroso, como la ley del medio excluido y el principio de no contradicción. Se estudian a continuación algunas de las operaciones básicas que pueden realizarse sobre conjuntos borrosos.

Igualdad

Se dice que dos conjuntos borrosos A y B , con funciones de pertenencia dadas por $\mu_A(x)$ y $\mu_B(x)$, respectivamente, son iguales si y sólo si $\mu_A(x) = \mu_B(x)$ para todo $x \in X$.

$$A = B \Leftrightarrow \mu_A(x) = \mu_B(x), \forall x \in X \quad (3.17)$$

Inclusión

Dados dos conjuntos borrosos A y B , con funciones de pertenencia dadas por $\mu_A(x)$ y $\mu_B(x)$, respectivamente, se dice que A está contenido en B , o de forma equivalente, que A es un subconjunto de B o que A es menor que B , si y sólo si $\mu_A(x) \leq \mu_B(x)$ para todo $x \in X$.

$$\left. \begin{array}{l} A \subset B \\ A \leq B \end{array} \right\} \Leftrightarrow \mu_A(x) \leq \mu_B(x), \forall x \in X \quad (3.18)$$

Intersección

La intersección entre dos conjuntos borrosos A y B , cuyas funciones de pertenencia son $\mu_A(x)$ y $\mu_B(x)$, respectivamente, se define como un conjunto borroso $C = A \cap B$ cuya función de pertenencia $\mu_C(x)$ será:

$$\mu_C(x) = \mu_{A \cap B}(x) = \min \{ \mu_A(x), \mu_B(x) \}, \forall x \in X \quad (3.19)$$

De forma abreviada podremos denotar:

$$\mu_C = \mu_A \wedge \mu_B \quad (3.20)$$

La operación de intersección definida a través de la función mínimo se suele denominar intersección clásica, ya que fue la definida originalmente por el profesor Zadeh. Posteriormente han aparecido otros operadores de intersección lógica que se pueden englobar en las denominadas T-normas. Estos operadores se verán con mayor detenimiento posteriormente.

Unión

La unión de dos conjuntos borrosos A y B , cuyas funciones de pertenencia son $\mu_A(x)$ y $\mu_B(x)$, respectivamente, puede definirse como un conjunto borroso $C = A \cup B$ cuya función de pertenencia $\mu_C(x)$ será:

$$\mu_C(x) = \mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x)\}, \forall x \in X \quad (3.21)$$

Esta ecuación podremos escribirla de forma abreviada como:

$$\mu_C = \mu_A \vee \mu_B \quad (3.22)$$

La función máximo se utiliza para definir el operador de unión clásico, sin embargo, la operación de unión lógica puede generalizarse a través de las S-normas, como se verá posteriormente.

Complemento

El complemento a uno o negación de un conjunto borroso A , cuya función de pertenencia es $\mu_A(x)$, se define por la expresión:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (3.23)$$

La operación complemento también se puede encontrar representada como:

$$\bar{A}, \neg A, A^c \text{ o } A'$$

Es importante destacar que tanto los conjuntos borrosos como las operaciones de unión, intersección y complemento cumplen las propiedades básicas de sus equivalentes clásicos:

Idempotencia:

$$\begin{aligned} A \cup A &= A \\ A \cap A &= A \end{aligned} \quad (3.24)$$

Identidad:

$$\begin{aligned} A \cup \emptyset = A \quad \text{y, si } A \subset X, \quad A \cap X = A \\ A \cap \emptyset = \emptyset \quad \text{y, si } A \subset X, \quad A \cup X = X \end{aligned} \quad (3.25)$$

Transitividad:

$$\text{Si } A \subset B \subset C, \text{ entonces } A \subset C \quad (3.26)$$

Involución:

$$\overline{\overline{A}} = A \quad (3.27)$$

Propiedad conmutativa:

$$\begin{aligned} A \cup B = B \cup A \\ A \cap B = B \cap A \end{aligned} \quad (3.28)$$

Propiedad asociativa:

$$\begin{aligned} A \cup (B \cap C) = (A \cup B) \cap C \\ A \cap (B \cup C) = (A \cap B) \cup C \end{aligned} \quad (3.29)$$

Con las operaciones y propiedades básicas estudiadas hasta ahora es posible extender a los conjuntos borrosos muchas de las identidades válidas para los conjuntos clásicos. Por ejemplo, se puede comprobar que en la teoría de conjuntos borrosos se cumplen las leyes de De Morgan [213], considerando los operadores clásicos de unión, intersección y negación propuestos inicialmente por Zadeh.

Leyes de De Morgan:

$$\begin{aligned} \overline{(A \cup B)} = \overline{A} \cap \overline{B} \\ \overline{(A \cap B)} = \overline{A} \cup \overline{B} \end{aligned} \quad (3.30)$$

Igualmente, bajo las mismas suposiciones, también se cumplen las leyes distributivas, es decir, se puede afirmar que los conjuntos borrosos forman un retículo distributivo con 0 y 1 [23].

Leyes distributivas:

$$\begin{aligned} C \cap (A \cup B) = (C \cap A) \cup (C \cap B) \\ C \cup (A \cap B) = (C \cup A) \cap (C \cup B) \end{aligned} \quad (3.31)$$

Como se ha expuesto previamente, no todas las propiedades de los conjuntos clásicos se cumplen en los conjuntos borrosos. La ley del medio excluido y el principio de no contradicción son dos propiedades que no se satisfacen cuando hablamos de conjuntos borrosos.

Sea A un subconjunto borroso del conjunto universal² X , $A \subset X$, la ley del medio excluido de conjuntos clásicos asegura que $A \vee \bar{A} = X$, pero si A es conjunto borroso no normal, $\max\{\mu_A(x), 1 - \mu_A(x)\}$ no será 1 para todo $x \in X$, por lo que $A \vee \bar{A} \neq X$. Lo mismo sucede con el principio de no contradicción. En la lógica clásica $A \wedge \bar{A} = \emptyset$, sin embargo, en la lógica borrosa $\min\{\mu_A(x), 1 - \mu_A(x)\}$ no tiene porqué ser cero para todo $x \in X$, por lo que no tiene porqué satisfacerse el principio de no contradicción.

Las funciones lógicas propuestas por el profesor Zadeh en el desarrollo inicial de la teoría de conjuntos borrosos pueden ser sustituidas por otras siempre que cumplan determinadas condiciones. A estas operaciones que generalizan las operaciones de intersección y unión, derivadas de los conceptos presentados por Menger [137] y Schwizer y Sklar [180], se las conoce como normas y conormas triangulares respectivamente [100][101][102].

T-normas

Además de la función mínimo, existen otros operadores que implementan dicha función. Estos son conocidos como normas triangulares o T -normas. Una norma triangular es una aplicación $T: [0,1] \times [0,1] \rightarrow [0,1]$ que cumple las propiedades conmutativa, asociativa, de monotonicidad e identidad unitaria (o condición de frontera). Las T -normas representan un modelo genérico para la intersección lógica (operador lógico Y o AND). Las propiedades conmutativa y asociativa ya han sido introducidas en (3.28) y (3.29). Dados dos pares de valores $x \leq y$, $w \leq z$, la propiedad de monotonicidad exige que se cumpla que $T(x, w) \leq T(y, z)$. Por último, la condición de frontera nos dice que $T(x, 1) = x$ para todo $x \in [0,1]$.

² Un conjunto universal es aquel que contiene todos los elementos, es decir, todos los elementos tienen un grado de pertenencia a dicho conjunto igual a 1.

En la Tabla 3.1 se muestran algunas de las principales T -normas.

Nombre	$T(x, y)$
Mínimo (estándar)	$\min\{x, y\}$
Producto	xy
Producto drástico	$\begin{cases} \min\{x, y\} & \text{si } \max\{x, y\} = 1 \\ 0 & \text{en otro caso} \end{cases}$
Producto acotado (Lukasiewicz)	$\max\{0, (1 + \lambda)(x + y - 1) - \lambda xy\}, \lambda \geq -1$
Hamacher	$\frac{xy}{\lambda + (1 - \lambda)(x + y - xy)}, \lambda \geq 0$
Yager	$1 - \min\left\{1, \sqrt[\lambda]{(1-x)^\lambda + (1-y)^\lambda}\right\}, \lambda \geq 0$
Dubois-Prade	$\frac{xy}{\max\{x, y, \alpha\}}, \alpha \in (0, 1)$
Frank	$\log_\lambda \left(1 + \frac{(\lambda^x - 1)(\lambda^y - 1)}{\lambda - 1}, \lambda > 0, \lambda \neq 1 \right)$
Producto de Einstein	$\frac{xy}{1 + (1-x) + (1-y)}$

Tabla 3.1 T -normas principales

T-Conormas

La unión borrosa también puede definirse mediante otras funciones distintas al máximo. Estas funciones son conocidas como T -conormas o S -normas. Una conorma triangular es una aplicación $S: [0, 1]^2 \rightarrow [0, 1]$ que cumple las propiedades conmutativa, asociativa, de monotonicidad e identidad cero (o condición de frontera). Las T -conormas representan un modelo genérico para la unión lógica (operación lógica O u OR). Se pueden apreciar las S -normas principales en la Tabla 3.2.

Para cada T -norma existe una S -norma dual o conjugada y viceversa. Así, para una T -norma, $T(x,y)$, su S -norma conjugada sería: $S(x,y)=1-T((1-x),(1-y))$.

Nombre	$S(x, y)$
Máximo (estándar)	$\max \{x, y\}$
Suma-producto	$x + y - xy$
Suma drástica	$\begin{cases} \max \{x, y\} & \text{si } \min \{x, y\} = 0 \\ 1 & \text{en otro caso} \end{cases}$
Suma acotada (Lukasiewicz)	$\min \{1, (x + y + \lambda xy)\}, \lambda \geq 0$
Hamacher	$\frac{x + y - (2 - \lambda)xy}{1 - (1 - \lambda)xy}, \lambda \geq 0$
Yager	$\min \left\{ 1, \sqrt[\lambda]{x^\lambda + y^\lambda} \right\}, \lambda \geq 0$
Dubois-Prade	$1 - \frac{(1-x)(1-y)}{\max \{(1-x), (1-y), \alpha\}}, \alpha \in [0, 1]$
Frank	$\log_\lambda \left(1 + \frac{(\lambda^{(1-x)} - 1)(\lambda^{(1-y)} - 1)}{\lambda - 1}, \lambda > 0, \lambda \neq 1 \right)$

Tabla 3.2 T -conormas principales

Aunque las T -normas y S -normas no pueden ordenarse, sí que puede identificarse claramente la mayor y menor de ellas. Así, la mayor de las T -normas es la función mínimo y la menor el producto drástico. En el caso de las S -normas, la mayor es la suma drástica, y la menor la función máximo. El resto de T -normas y S -normas están comprendidas entre éstas.

Producto algebraico

El producto algebraico de dos conjuntos borrosos A y B , cuyas funciones de pertenencia son $\mu_A(x)$ y $\mu_B(x)$, respectivamente, se denota por AB y se define mediante la ecuación (3.32). El conjunto borroso AB cumple $AB \subset A \cap B$.

$$\mu_{(AB)}(x) = \mu_A(x)\mu_B(x) \tag{3.32}$$

Suma algebraica

La suma algebraica de dos conjuntos borrosos A y B , cuyas funciones de pertenencia son $\mu_A(x)$ y $\mu_B(x)$, respectivamente, se denota por $A + B$ y se define mediante la ecuación (3.33). Si A y B son conjuntos borrosos normalizados, el conjunto borroso obtenido mediante la suma algebraica de A y B sólo será normalizado si $\mu_A(x) + \mu_B(x) \leq 1 \quad \forall x \in X$, es decir, a diferencia del producto algebraico, la suma algebraica de dos conjuntos borrosos sólo tendrá sentido si $\mu_{(A+B)}(x) \leq 1 \quad \forall x \in X$.

$$\mu_{(A+B)}(x) = \mu_A(x) + \mu_B(x) \quad (3.33)$$

Suma

La suma de dos conjuntos borrosos A y B , cuyas funciones de pertenencia son $\mu_A(x)$ y $\mu_B(x)$, respectivamente, se denota por $A \oplus B$ y se define mediante la ecuación (3.34). Si A y B son conjuntos borrosos normalizados, el conjunto borroso $A \oplus B$ será normalizado en todo caso, por lo que la operación suma si tiene sentido en todo el universo de discurso de los conjuntos A y B .

$$\begin{aligned} (A \oplus B) &= A + B - AB \\ \mu_{(A \oplus B)}(x) &= \mu_A(x) + \mu_B(x) - \mu_A(x)\mu_B(x), \quad \forall x \in X \end{aligned} \quad (3.34)$$

Nótese que el producto algebraico es una T-norma, por lo que es equivalente a la operación de intersección borrosa, sin embargo la suma algebraica no es una T-conorma, por lo que no sería la operación dual al producto algebraico. En lógica borrosa, la operación dual del producto algebraico es la suma, ya que ésta si es equivalente a la operación unión (es una T-conorma).

3.4. Variables lingüísticas

Las técnicas convencionales para el análisis de sistemas no son apropiadas para resolver problemas estrechamente relacionados con las percepciones, emociones y juicios humanos. Las variables lingüísticas constituyen una alternativa para modelar el pensamiento humano.

Una variable lingüística es una variable cuyos valores son palabras o sentencias pertenecientes al lenguaje natural o artificial. El concepto de variable lingüística fue introducido por Zadeh [216] y desarrollado posteriormente por él mismo en varios trabajos [215][217][218][219]. Estas variables permiten el acercamiento de la lógica borrosa al lenguaje natural, facilitando la utilización lógica del lenguaje ordinario y eludiendo las deficiencias del lenguaje preciso. Esta idea subyace en los primeros trabajos de Zadeh, y se formalizó en su conocido Principio de Incompatibilidad [214]: *“A medida que la complejidad de un sistema aumenta, disminuye nuestra capacidad para hacer afirmaciones precisas, incluso significativas, sobre su comportamiento, hasta que se alcanza un umbral más allá del cual precisión y relevancia son características casi mutuamente excluyentes”*.

La variable lingüística constituye el núcleo central a partir del cual el experto define el conjunto de reglas de un sistema borroso. Está caracterizada por la cuádrupla $\{X, T(X), U, G\}$, donde:

X es el nombre de la variable lingüística.

$T(X)$ representa al conjunto de términos (valores lingüísticos) definidos en X .

U es el dominio físico real sobre el que están definidos los valores que se aplican a la variable lingüística.

G representa una función semántica que da un “significado” (interpretación) a una variable lingüística en función de los elementos a los que X representa.

En la Figura 3.13 se puede ver una posible representación de la variable lingüística Velocidad. Cada término lingüístico se representa de forma cuantitativa mediante una función de pertenencia. En el ejemplo, los parámetros que caracterizan a esta variable son:

$X = \text{Velocidad}$

$T(X) = \{\text{Lenta, Moderada, Rápida}\}$

$U = [0-100] \text{ km/h}$

G puede ser definida por un experto como:

$$\text{Lenta} = \begin{cases} 1 & \text{si } 0 \leq X \leq 40 \\ 1 - (X - 40) / 20 & \text{si } 40 \leq X \leq 60 \\ 0 & \text{en otro caso} \end{cases}$$

$$\text{Moderada} = \begin{cases} 1 - |X - 60| / 20 & \text{si } 40 \leq X \leq 80 \\ 0 & \text{en otro caso} \end{cases}$$

$$\text{Rápida} = \begin{cases} 1 & \text{si } 80 \leq X \leq 120 \\ 1 - (80 - X) / 20 & \text{si } 60 \leq X \leq 80 \\ 0 & \text{en otro caso} \end{cases}$$

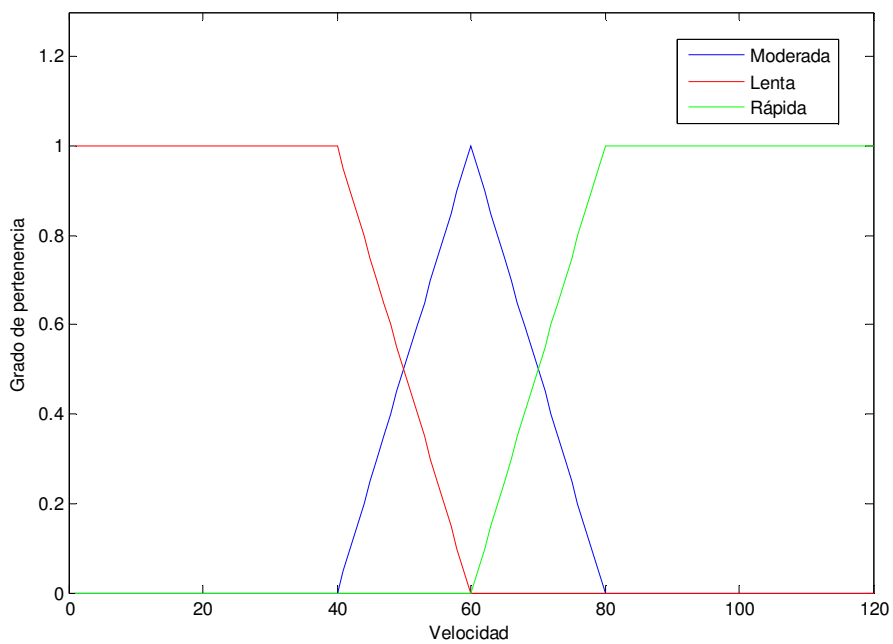


Figura 3.13 Variable lingüística Velocidad

Los términos *Lenta* y *Rápida* se han creado utilizando funciones de pertenencia trapezoidales y la etiqueta *Moderada* mediante una función triangular.

Los términos lingüísticos pueden ser primarios o modificados. Los modificadores lingüísticos se obtienen de aplicar un concentrador, un dilatador o una negación de un término lingüístico primario.

Concentración, Dilatación y Negación de términos lingüísticos

El operador de concentración origina el modificador lingüístico que podemos denominar *más*, *muy* o *bastante*. Dado un conjunto borroso A en X definido por la función de pertenencia $\mu_A(x)$, la operación de concentración se realiza mediante la expresión:

$$\mu_{A^k}(x) = \mu_A(x)^k, \quad \kappa > 1 \quad (3.35)$$

El operador de dilatación modifica la función de pertenencia sobre la que actúa de forma opuesta al operador de concentración, de modo que ahora la modificación lingüística será *menos*, *poco* o *más o menos*. Dado un conjunto borroso A en X definido por la función de pertenencia $\mu_A(x)$, la operación de dilatación se realiza mediante la expresión:

$$\mu_{A^k}(x) = \mu_A(x)^k, \quad 0 \leq \kappa < 1 \quad (3.36)$$

Normalmente se utiliza $k = 2$ para la operación de concentración y $k = 0.5$ para la dilatación.

El operador de negación origina el modificador lingüístico que podemos denominar *no*. Consiste en aplicar una función de negación $N : [0,1] \rightarrow [0,1]$, continua, monótona no creciente, involutiva, es decir, $N(N(x)) = x$, para todo $x \in X$, y que cumple las condiciones de frontera $N(0) = 1$ y $N(1) = 0$.

En la Tabla 3.3 se presentan algunos operadores de negación. Nótese que si $\lambda = 0$ el operador de Sugeno coincide con la operación de negación estándar e igual sucede si $\omega = 1$ con el operador de Yager.

Nombre	$N(x, y)$
Máximo (estándar)	$1 - x$
Umbral	$\begin{cases} 1 & \text{si } x \leq \lambda \\ 0 & \text{si } x > \lambda \end{cases} \quad 0 < \lambda < 1$
Sugeno	$\frac{1-x}{1+\lambda x} \quad \lambda \geq -1$
Yager	$\sqrt[\omega]{(1-x^\omega)} \quad \omega > 0$

Tabla 3.3 Operadores de Negación

Siguiendo con el ejemplo anterior, podemos crear los modificadores lingüísticos *no rápida*, utilizando el operador de negación estándar, *muy lenta*, utilizando la operación de concentración y *más o menos moderada*, aplicando la operación de dilatación. El conjunto de términos modificados se puede representar por:

$$T(X) = \{ \text{Muy Lenta, Más o Menos Moderada, No Rápida} \}$$

La función semántica G queda definida según:

$$\text{Muy Lenta} = \begin{cases} 1 & \text{si } 0 \leq X \leq 40 \\ (1 - (X - 40) / 20)^2 & \text{si } 40 \leq X \leq 60 \\ 0 & \text{en otro caso} \end{cases}$$

$$\text{Más o Menos Moderada} = \begin{cases} (1 - |X - 60| / 20)^{0.5} & \text{si } 40 \leq X \leq 80 \\ 0 & \text{en otro caso} \end{cases}$$

$$\text{No Rápida} = \begin{cases} 0 & \text{si } 80 \leq X \leq 120 \\ 1 - (1 - (80 - X) / 20) & \text{si } 60 \leq X \leq 80 \\ 1 & \text{en otro caso} \end{cases}$$

Si se representa la variable lingüística con los modificadores lingüísticos aplicados se obtienen las funciones de pertenencia que se ven en la Figura 3.14.

Nótese que los modificadores lingüísticos basados en concentraciones o dilataciones del conjunto borroso, no implican un desplazamiento de éste, sino una modificación de su forma. Esto quiere decir que el significado de la variable lingüística *Muy Lenta* del ejemplo

no se refiere a una velocidad mayor, sino a una velocidad de la que se quiere asegurar su buena pertenencia a ese conjunto.

Para un mismo universo de discurso, cuanto mayor sea el número de términos lingüísticos empleados para representar una variable, mayor precisión se podrá obtener del sistema borroso. Evidentemente, al tener un número mayor de etiquetas lingüísticas, cada una de ellas representará un subespacio menor, por lo que se podrá caracterizar el sistema más finamente; sin embargo, también tendrá mayor complejidad. Según Miller [138], el número de valores capaces de discriminar un ser humano varía entre cinco y nueve. Otro detalle a tener en cuenta si se desea hacer un tratamiento lingüístico del sistema borroso, es que normalmente se debe escoger un número impar de etiquetas. De esta forma existirá una etiqueta central que hará las veces de neutral, y el resto se distribuirá a sus lados.

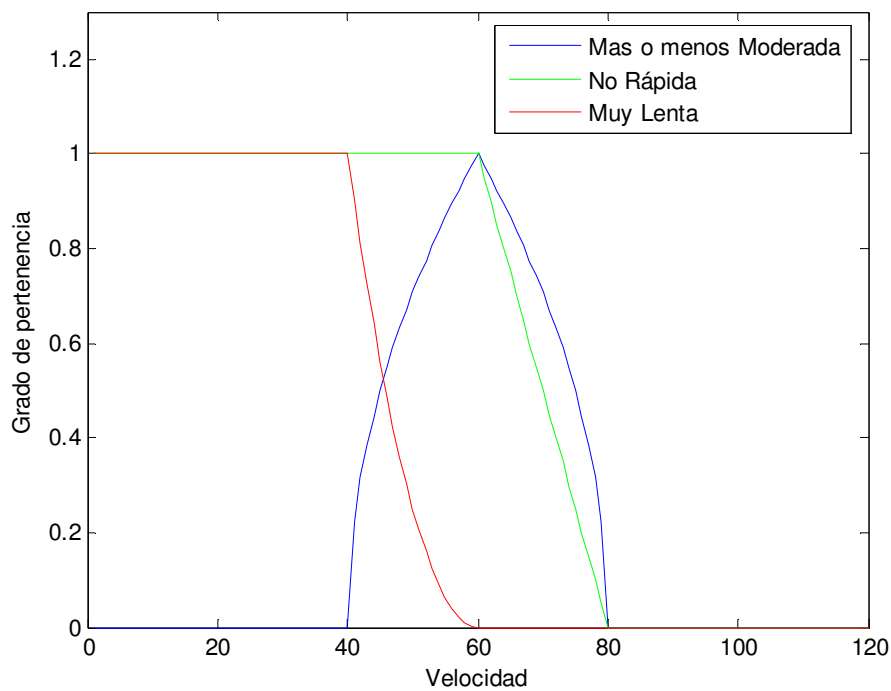


Figura 3.14 Variable Velocidad con modificadores lingüísticos

3.5. Diseño de sistemas borrosos

Tradicionalmente se han diseñado los sistemas borrosos definiendo los conjuntos borrosos de forma discreta (un conjunto de puntos) [107], para lo cual no se necesita definir una función de pertenencia explícita; no obstante, el número de puntos necesarios para representar una función es grande y depende del método de borrosificación (ver apartado 3.5.1) que sea empleado. Como cada regla y cada función de pertenencia están definidas como un conjunto de puntos, entonces para alterar una regla se deben alterar el conjunto de puntos que las define. Al anterior tipo de representación se le denomina *sistema borroso discreto* [29].

En la actualidad, los *sistemas borrosos continuos* han ganado en popularidad debido a que existen un conjunto de algoritmos y aplicaciones para el modelado y control basados en la adaptación de parámetros que originalmente fueron concebidos para las redes neuronales[144]. Los sistemas borrosos continuos están basados en funciones de pertenencia que son una aplicación continua definida por un conjunto de parámetros; con menos requerimientos en memoria pero un costo computacional alto, dependiente del algoritmo de adaptación de parámetros.

Para diseñar un sistema borroso es necesario definir una serie de parámetros en dos fases de desarrollo fundamentales: una *fase imprecisa* y una *fase determinista*. La fase imprecisa es aquella en que, a partir del conocimiento del experto e información heurística se definen los conjuntos borrosos o funciones de pertenencia equivalentes a la variables lingüísticas, los operadores borrosos que serán aplicados, los mecanismos de inferencia, el método de desborrosificación y las reglas que son invariantes. La fase determinista la conforma la función no lineal multivariable que aplica las variables de entrada a la salida, que no está sujeta a la vaguedad ni a la imprecisión.

El diseño de un modelo borroso es un proceso donde existe una continua interacción entre la fase imprecisa y la fase determinista, tal y como se aprecia en la Figura 3.15.

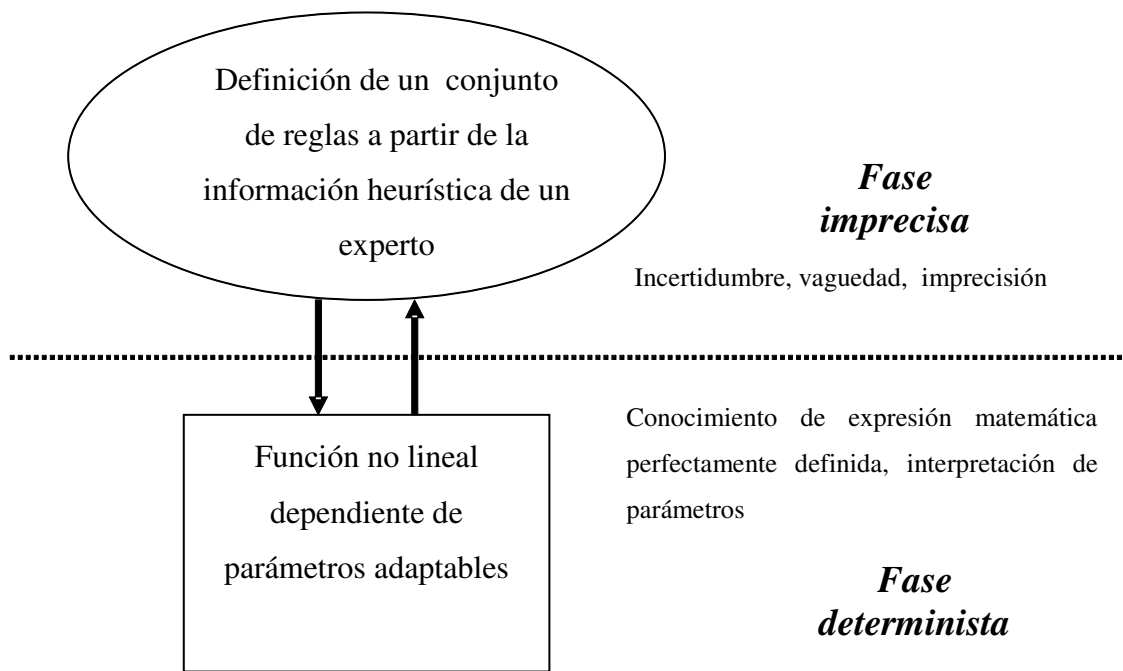


Figura 3.15 Distinción entre la fase imprecisa y determinista para elaborar un modelo borroso.

3.5.1. Razonamiento borroso

Un sistema borroso es un sistema inteligente que, a partir de un conjunto de funciones de pertenencia definidas en un determinado universo de discurso y determinadas reglas definidas por un experto realiza una inferencia a partir de un conjunto de datos.

Una regla borrosa es una sentencia condicional que tiene la estructura Si-Entonces (If-Then), siendo el conjunto de reglas de un sistema borroso lo que se conoce como *base de reglas* o *base de conocimiento* del sistema. En su forma más simple, una regla borrosa viene dada por la expresión (3.37). Ésta se compone de una premisa (condición de la regla) y de una consecuencia (conclusión de la regla). Esta expresión puede hacerse más compleja mediante la aplicación de operadores de negación, unión, intersección, modificadores lingüísticos, etc. Un ejemplo de construcción más compleja de una regla borrosa se muestra en la ecuación (3.38).

$$\text{Si } x \text{ es } A \text{ entonces } y \text{ es } B \tag{3.37}$$

$$\text{Si } \textit{precio es alto} \text{ y } \textit{demanda es baja} \text{ Entonces } \textit{ventas es pequeña} \tag{3.38}$$

La premisa de la regla se conoce como *antecedente*, mientras que la consecuencia se conoce como *consecuente*. En el ejemplo anterior, “Si *precio es alto y demanda es baja*” se corresponde con el antecedente de la regla, en este caso formado por la unión mediante el operador \wedge y de dos premisas. Las variables lingüísticas de entrada son *precio* y *demanda*, etiquetadas lingüísticamente en esta regla como *alto* y *baja*, respectivamente. “*Ventas es pequeña*” es el consecuente de la regla, donde *ventas* se corresponde con la variable lingüística de salida y *pequeña* con la etiqueta lingüística que se le asignará si se cumple el antecedente. El consecuente de una regla puede ser un conjunto borroso, un punto borroso, o una función dependiente de las entradas.

Nombre	$\mu_{\text{Implicación}}(x, y)$
Mamdani (mínimo)	$\min\{\mu_A(x), \mu_B(y)\}$
Larsen (producto)	$\mu_A(x) \cdot \mu_B(y)$
Reichencach	$1 - \mu_A(x) + \mu_A(x) \cdot \mu_B(y)$
Lukasiewicz	$\min\{1, 1 - \mu_A(x) + \mu_B(y)\}$
Willmott	$\max\{1 - \mu_A(x), \min\{\mu_A(x), \mu_B(y)\}\}$
Kleene-Dienes	$\max\{1 - \mu_A(x), \mu_B(y)\}$

Tabla 3.4 Operadores de implicación

El operador que relaciona la premisa con la consecuencia se conoce como *operador de implicación*, y se representa lingüísticamente con el adverbio *Entonces (then)*. Existen muchas maneras de definir el concepto de implicación [140][141][195]. Basándose en cada una de ellas se pueden generar distintas funciones en base a T -normas y T -conormas. En la Tabla 3.4 se presentan varios operadores de implicación, siendo los más importantes y utilizados los operadores Mamdani y Larsen.

Al contrario de lo que sucede en la lógica clásica, la lógica borrosa no requiere que las reglas se cumplan de forma exacta, sino que pueden cumplirse en cierto grado. Esta propiedad de la lógica borrosa es lo que caracteriza al razonamiento aproximado. Esto implica que se puede inferir una conclusión aunque el hecho no verifique la regla plenamente. Evidentemente, si el grado de cumplimiento del antecedente no es completo,

el grado de acción del consecuente tampoco lo será. De esta forma, el consecuente final se parecerá más a la conclusión formal de la regla cuanto mayor sea el grado de cumplimiento de ésta.

3.5.2. Sistema de inferencia borroso

Los sistemas de inferencia borrosos son algoritmos basados en la lógica borrosa y el razonamiento aproximado. En función de la aplicación a la que se destine el sistema de inferencia éste puede denominarse de distintas formas: *modelo borroso*, *sistema experto borroso*, *controlador borroso*, *memoria asociativa borrosa*, etc. Los sistemas de inferencia borrosos se componen básicamente de dos elementos, el *motor de inferencia* y la *base de conocimiento*, sin embargo, según la aplicación para la que vaya a emplearse el sistema de inferencia, puede ser necesario añadir elementos adicionales, como el *borrosificador* y el *desborrosificador* que permiten conectar el sistema de inferencia borroso con el mundo real [136]. A continuación se estudiarán cada uno de los elementos de un sistema de inferencia borroso, cuya configuración puede verse en la Figura 3.16.

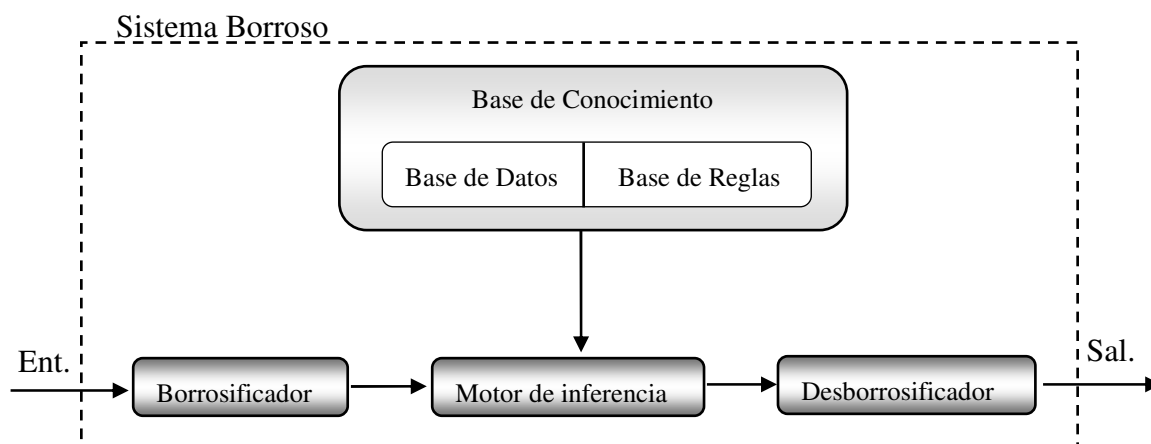


Figura 3.16 Sistema de inferencia borroso

Borrosificador

Cuando las entradas que deben aplicarse a un sistema borroso son valores numéricos concretos, no borrosos, es necesario establecer una interfaz con el motor de inferencia borroso que elabore conjuntos borrosos a partir de entradas no borrosas. Esta interfaz se conoce como borrosificador (*fuzzifier*).

El borrosificador, por lo tanto, convierte los valores de las variables de entrada definidas en la premisa de las reglas (a través de los términos lingüísticos definidos por las funciones de pertenencia) en un determinado grado de pertenencia, realizando un cambio en el dominio de definición de cada variable del universo de discurso a una entidad borrosa definida en el intervalo $[0,1]$. El cambio de dominio se realiza en cada antecedente por separado. Si la premisa es una sentencia compuesta, entonces se aplica el operador lógico que relaciona a la entidad borrosa que representa cada antecedente, el cual ha sido definido por el experto. El resultado de aplicar el operador borroso entre los antecedentes de la regla devuelve un valor que define el grado de veracidad de la regla, el cual cambia en función de los valores que tomen las variables de entrada.

El sistema de borrosificación más simple consiste en sustituir el valor numérico por un conjunto borroso de tipo *singleton*, o punto borroso.

Otro posible borrosificador sería un algoritmo que asignara a cada entrada una representación borrosa mediante términos lingüísticos definidos en su correspondiente universo de discurso. Este esquema es más complejo y aporta poco al sistema por lo que no es frecuente su utilización.

Por último, existe la posibilidad de emplear la capacidad de la lógica borrosa para manejar la incertidumbre como método de borrosificación. En este caso, se sustituye la variable numérica de entrada por un conjunto borroso que represente mediante una distribución de posibilidad (una función de pertenencia) los posibles valores que ésta podría tomar realmente. Este borrosificador puede resultar bastante útil pero supone una mayor complejidad del sistema de inferencia.

Inferencia borrosa

El motor de inferencia borrosa es el mecanismo que permite obtener la salida de un sistema borroso en función de sus reglas y las entradas que le sean aplicadas. La salida puede ser un conjunto borroso o un valor numérico, según requiera la aplicación. Los pasos que se siguen en durante el mecanismo de inferencia borrosa se describen a continuación:

1. Se calcula el grado de cumplimiento de cada antecedente en función de la entrada del sistema. En caso de utilizar un borrosificador *singleton*, este valor se obtendrá directamente de la intersección de la entrada con la función de pertenencia del antecedente; si por el contrario, la entrada está representada

por una función de pertenencia no *singleton*, será necesario aplicar un operador de intersección, una *T*-norma, sobre dichos conjuntos.

2. Tras calcular los antecedentes de cada regla se componen según la definición lingüística de la misma, es decir, aplicando los operadores de intersección para condiciones *Y*, y los de unión para condiciones *O*. Esta medida proporciona un valor que representa la calidad con la que las reglas son satisfechas por la condición de entrada del sistema borroso (su grado de activación).
3. Se obtiene el consecuente resultante de cada regla mediante el operador de implicación, *Entonces*. Los operadores de implicación más utilizados son las *T*-normas mínimo y producto. Al emplear el operador mínimo, la función de pertenencia del consecuente es recortada a la altura definida por el grado de veracidad de la regla; si se emplea el operador producto, la salida es escalada según dicho grado de veracidad.
4. Por último, la regla composicional u operador de agregación, combina los resultados de todas las reglas, o sea, el resultado de la base de reglas, en un único conjunto borroso. Las *T*-conormas más utilizadas para obtener el conjunto borroso de salida suelen ser el máximo y la suma acotada.

El motor de inferencia permite la fusión de la información de múltiples reglas en una conclusión única. Esta conclusión se obtiene a partir de la combinación de los consecuentes de las reglas del sistema, ponderadas según una función del grado de cumplimiento de cada una de ellas.

Base de conocimiento

La base de conocimiento (*knowledge base*) contiene las relaciones existentes entre las entradas y salidas del sistema. El proceso de inferencia obtendrá las salidas asociadas a las entradas del mismo basándose en este conocimiento.

La base de conocimiento puede dividirse en dos partes: la base de datos y la base de reglas. La base de datos (*data base*) contiene la definición de las variables lingüísticas empleadas en las reglas, es decir, las funciones de pertenencia que definen cada una de las etiquetas lingüísticas. La base de reglas (*rule base*) almacena la colección de reglas

lingüísticas del sistema. Ésta puede representarse como una lista de reglas (lo más habitual), o mediante una tabla o matriz de decisión.

Existen varios métodos para la obtención de las reglas de un sistema borroso, entre los que cabe destacar:

1. Basados en el conocimiento de un experto, o heurísticos.
2. Basados en las acciones de control de un operador, en función de los datos de entrada-salida observados.
3. Basados en aprendizaje mediante algoritmos automáticos, utilizando datos de entrada-salida.

Desborrosificador

Cuando la salida del sistema de inferencia es un conjunto borroso pero la aplicación requiere que se un valor numérico concreto, es necesario utilizar un desborrosificador (*defuzzifier*). Éste convierte dicho conjunto en un valor numérico del universo de discurso de salida, representativo de la conclusión obtenida.

Existen muchos métodos de desborrosificación [103][136] y siguen apareciendo nuevos métodos en la literatura [67][117][124][198][207]. Se presentan a continuación los desborrosificadores más comunes:

1. *Método del centro de gravedad o del centroide*: La salida se obtiene calculando el centro de gravedad del conjunto borroso de salida. Si representamos el conjunto borroso de salida como $D(y)$, su valor desborrosificado (y_0) se obtiene a partir de la ecuación (3.39). Para valores discretos la ecuación sería la mostrada en (3.40). Este método es el más empleado, ya que, aunque es costoso computacionalmente comparado con otros métodos de desborrosificación, proporciona variaciones suaves y continuas de los valores de salida.

$$y_0 = \frac{\int y \mu_{D(y)} dy}{\int \mu_{D(y)} dy} \quad (3.39)$$

$$y_0 = \frac{\sum y \mu_{D(y)}}{\sum \mu_{D(y)}} \quad (3.40)$$

2. *Método de la máxima pertenencia*: Si $D(y)$ es el conjunto borroso de salida, el valor numérico de salida y_0 se corresponde con el valor de y para el que $\mu_{D(y)}$ alcanza su valor máximo. Pueden existir múltiples valores para los que $\mu_{D(y)}$ tenga un mismo valor máximo, por lo que se hace necesario utilizar algún método para elegir el valor de y_0 . Estos métodos son, el *menor de los máximos*, el *centro de los máximos* y el *mayor de los máximos*. Este desborrosificador es fácil de implementar pero genera cambios bruscos en la salida, lo que en algunas aplicaciones no es admisible.
3. *Método del centroide indexado*: Este método calcula el centro de gravedad de la parte correspondiente al conjunto borroso inferido cuyo grado de pertenencia sea mayor de un determinado λ .

3.5.3. Tipos de sistemas borrosos

En función de la forma de las reglas y del tipo de entradas y salidas, se distinguen cuatro tipos de sistemas borrosos [44][210], relacionados esquemáticamente como se muestra en la Figura 3.17: los sistemas puros, los de tipo Mamdani, los de Sugeno o TSK y los modelos aditivos de Kosko o SAM, además de algunas variantes como los modelos de Tsukamoto.

Sistema borroso puro

Los sistemas borrosos puros son sistemas donde las entradas y salidas son conjuntos borrosos y por lo tanto no se necesita realizar ninguna transformación de las mismas. Están compuestos sólo por la base de conocimiento y el motor de inferencia.

Sistema borroso de tipo Mamdani

La potencia práctica de la teoría borrosa de Zadeh quedó demostrada cuando Mamdani y Assilian crearon el primer sistema borroso aplicado a un problema de control [126][128]. Su trabajo consistió en controlar una máquina de vapor mediante un conjunto de reglas lingüísticas obtenidas de la experiencia de los operadores de la máquina. Por ello, el esquema de control borroso que utilizaron [127] y que se corresponde con la Figura 3.16, se conoce como modelo de control de Mamdani-Assilian, o de forma abreviada, como modelo de Mamdani.

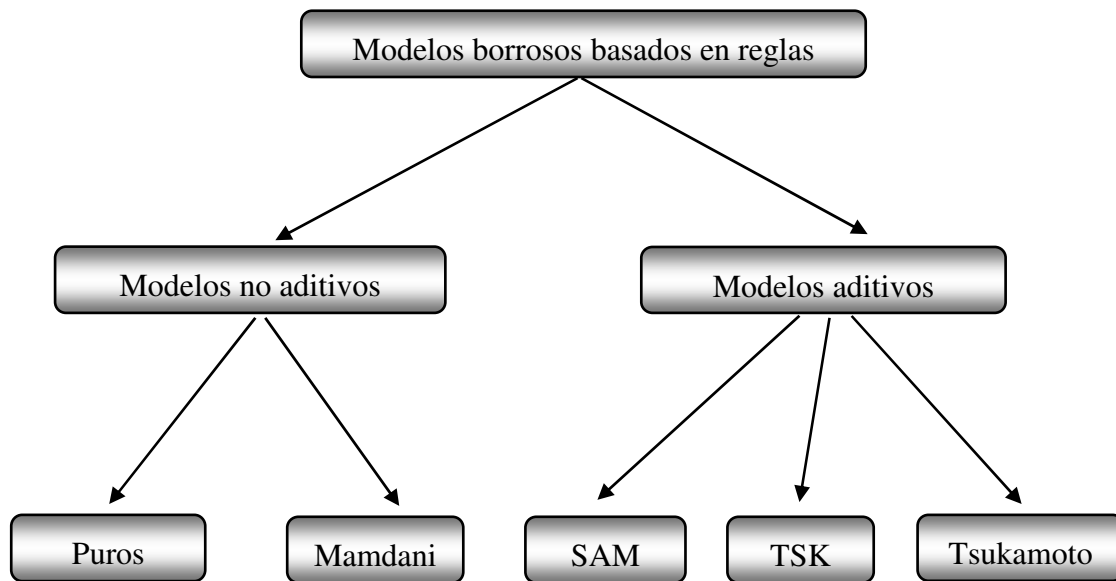


Figura 3.17 Tipos de modelos borrosos

Los sistemas borrosos de tipo Mamdani aportan una gran cantidad de ventajas:

1. Se comportan como aproximadores universales [59][116][156][212][220].
2. Pueden utilizarse en aplicaciones reales, ya que tratan con facilidad entradas y salidas reales.
3. Proporcionan un marco natural para la inclusión del conocimiento de expertos en forma de reglas lingüísticas.
4. Hay gran facilidad para elegir el método de inferencia borrosa, así como las interfaces de borrosificación y desborrosificación.

Las principales limitaciones de este modelo son [44]:

1. Falta de flexibilidad debido a la rigidez con que se particionan los espacios de entrada y salida.
2. No hay una distinción clara entre el conocimiento experto y la definición de las variables lingüísticas incluidas en las reglas borrosas.
3. Es complicado obtener una partición borrosa adecuada de los espacios de entrada cuando las variables de entrada dependen unas de otras.

4. El tamaño de la base de conocimiento depende directamente del número de variables y términos lingüísticos que existan en el sistema.
5. Para una correcta interpretabilidad del sistema no se deben usar un número excesivo de reglas, lo que en ocasiones va en contra de la precisión del sistema borroso implementado.

El conjunto de reglas de un sistema Mamdani se configura como:

$$\text{Regla } r : \text{Si } x_1 \text{ es } A_1^r \text{ y } x_2 \text{ es } A_2^r \text{ y } \dots \text{ y } x_n \text{ es } A_n^r \text{ entonces } y \text{ es } B^r \quad (3.41)$$

donde $r = 1 \dots R$, x_1, x_2, \dots, x_n son las entradas del sistema, $A_1^r, A_2^r, \dots, A_n^r$ son los conjuntos borrosos del antecedente de las regla r definidos en los universos de discurso de sus entradas asociadas, y es la salida del sistema, y B^r el conjunto borroso consecuente de la regla r , definido en el universo de discurso de la salida y .

El grado de activación de la regla r , ω^r , se obtiene mediante la aplicación del operador de intersección, que normalmente será alguna de las T -normas presentadas. $\omega^r = \alpha_1^r \wedge \alpha_2^r \wedge \dots \wedge \alpha_n^r$, donde cada α_i^r representa el grado de emparejamiento entre la entrada x_i y el conjunto borroso antecedente A_i^r . El conjunto borroso de salida de la regla que genera el motor de inferencia, η_{B^r} , viene dado por el operador de implicación: $\eta_{B^r} = \omega^r \wedge \eta_{B^r}$, siendo η_{B^r} la función de pertenencia que define al conjunto borroso consecuente B^r . La salida borrosa del motor de inferencia, η_B , se calcula agregando mediante una regla composicional, normalmente una T -conorma, los efectos de todas las reglas del sistema: $\mu_B = \bigcup_{r=1}^R \eta_{B^r}$. Finalmente, el desborrosificador calcula la salida numérica del sistema, y . Destacar que en los sistemas de tipo Mamdani tanto los antecedentes, A_i , como los consecuentes, B , están formados por conjuntos borrosos, mientras que las entradas, x_i , y salidas, y , son valores numéricos.

Uniendo la interpretabilidad de sus reglas a la capacidad aproximadora universal de estos modelos, podemos concluir que los modelos Mamdani son herramientas muy poderosas para el análisis y control de sistemas complejos, aunque una buena

aproximación con un modelo Mamdani puede requerir un número elevado de reglas, lo que merma la interpretabilidad del sistema conforme aumenta su complejidad.

Modelo aditivo estándar, SAM

El modelo aditivo estándar (Standard Additive Model, SAM), es un caso particular de modelo borroso aditivo. Fue propuesto por Bart Kosko [106][104] y su esquema de bloques puede verse en la Figura 3.18.

El esquema de inferencia de un modelo SAM es similar al de los modelos TSK que se verán a continuación, ya que en ambos la salida se genera en base a la suma ponderada de las consecuencias de cada una de las reglas de la base de conocimiento. La estructura, sin embargo, es idéntica a la de los modelos Mamdani, por lo que pueden considerarse una variación de éstos. Las principales diferencias [210] entre los esquemas de inferencia de los sistemas SAM y los Mamdani se pueden resumir en:

1. La entrada (x) en un modelo SAM se considera siempre un punto borroso.
2. Se utiliza el operador producto para la inferencia, en vez del mínimo que suele usarse en los modelos Mamdani.
3. La combinación final (D) de las conclusiones parciales de cada regla borrosa (D_r), se calcula mediante una suma ponderada según los pesos ω , mientras los modelos Mamdani suelen utilizar el operador máximo.
4. La salida (y) se calcula mediante la adición de las salidas parciales ponderadas, frente a los modelos Mamdani que deben especificar un desborrosificador.

Modelo Takagi-Sugeno-Kang (TSK)

Los sistemas TSK, modelos Takagi-Sugeno (TS) o de forma más simplificada, modelos Sugeno, se caracterizan por utilizar como consecuente una función de las variables de entrada del sistema. Este tipo de sistema borroso fue propuesto por Takagi y Sugeno [184] en 1985 incorporándose posteriormente Kang [183] en 1988.

Una regla de un sistema TSK tiene la forma:

$$\text{Regla } r : \text{Si } x_1 \text{ es } A_1^r \text{ y } x_2 \text{ es } A_2^r \text{ y } \dots \text{ y } x_n \text{ es } A_n^r \text{ entonces } y^r = g^r(x) \quad (3.42)$$

donde x_1, x_2, \dots, x_n son las entradas del sistema e y^r es la salida inferida por la regla $r = 1 \dots R$. $A_1^r, A_2^r, \dots, A_n^r$ son los conjuntos borrosos del antecedente de las regla r , definidos en los universos de discurso de sus entradas asociadas. La consecuencia inferida a partir de la regla se calcula con la función $g^r(x)$, dependiente de las entradas del sistema.

Las funciones consecuentes $g^r(x)$ más utilizadas son los polinomios de grado 0 y de grado 1. La utilización de polinomios de grado superior o de funciones no polinómicas añaden complejidad al sistema y dificultan su interpretabilidad, por lo que no es muy común su utilización.

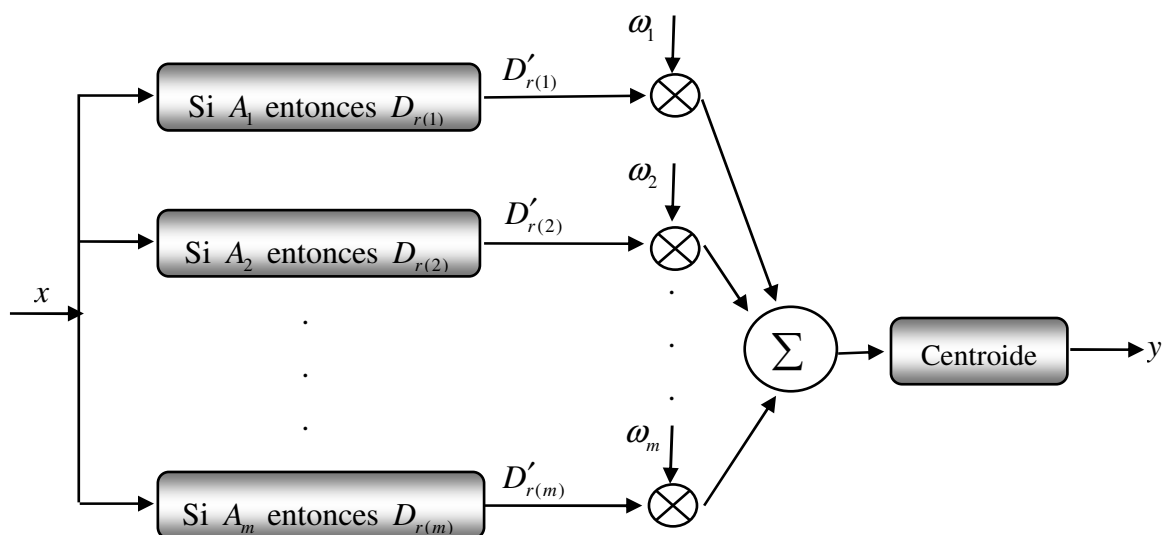


Figura 3.18 Diagrama de bloques de un modelo aditivo estándar (SAM)

Si la función consecuyente es un polinomio de orden 0, se le denomina *consecuyente afín*, y las reglas tienen la forma:

$$\text{Regla } r : \text{ Si } x_1 \text{ es } A_1^r \text{ y } x_2 \text{ es } A_2^r \text{ y } \dots \text{ y } x_n \text{ es } A_n^r \text{ entonces } y^r = b_0^r \quad (3.43)$$

Si se emplea un *consecuyente lineal*, o de orden 1, las reglas tienen la forma:

$$\text{Regla } r : \text{ Si } x_1 \text{ es } A_1^r \text{ y } x_2 \text{ es } A_2^r \text{ y } \dots \text{ y } x_n \text{ es } A_n^r \text{ entonces } y^r = b_0^r + b_1^r x_1 + \dots b_n^r x_n \quad (3.44)$$

Los elementos $b_0^r, b_1^r, \dots, b_n^r$ son parámetros reales, siendo b_i^r el término que multiplica a la variable i -ésima del sistema de la regla r , y b_0^r el término afín de la regla r .

La salida de un sistema TSK se calcula mediante la suma ponderada o el promedio ponderado del efecto individual de todas sus reglas. La suma ponderada es la suma de los valores obtenidos por la función consecuyente, ponderadas por el grado de activación de cada una de las reglas. El promedio ponderado consiste en dividir la suma ponderada por el sumatorio de las fuerzas de activación de las reglas. Esto permite que las reglas que se cumplen en mayor grado afecten más a la salida del sistema que las reglas que se cumplen en menor medida.

Si la base de las reglas está formada por R reglas y ω^r es el grado de cumplimiento de la regla r , la salida aplicando la suma ponderada se obtiene mediante la aplicación de la ecuación:

$$y = \sum_{r=1}^R \omega^r y^r \quad (3.45)$$

La salida aplicando el promedio ponderado se obtiene aplicando la ecuación:

$$y = \frac{\sum_{r=1}^R \omega^r y^r}{\sum_{r=1}^R \omega^r} \quad (3.46)$$

El grado de cumplimiento de cada regla, ω^r , se obtiene mediante la aplicación de un operador de conjunción que se modela mediante una T -norma; normalmente la función mínimo o producto: $\omega^r = T\{A_1^r(x_1), A_2^r(x_2), \dots, A_n^r(x_n)\}$.

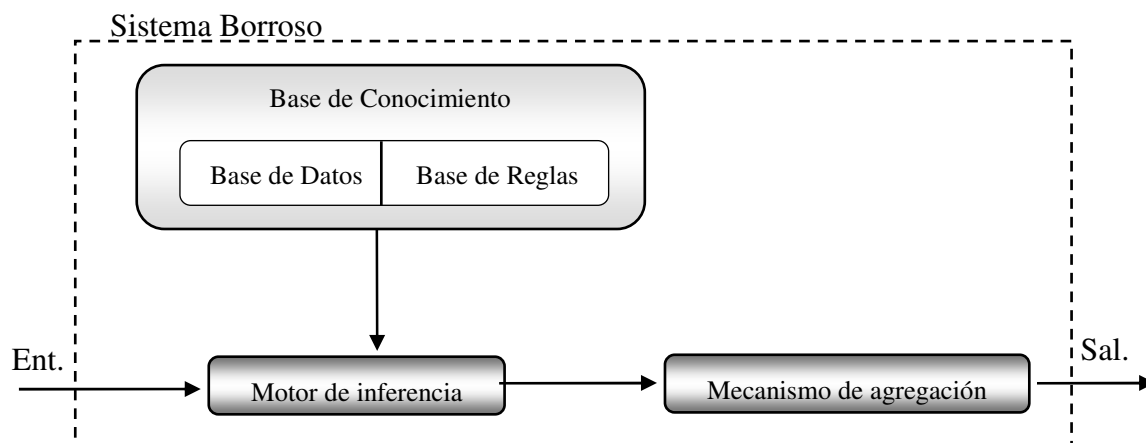


Figura 3.19 Sistema borroso de Takagi-Sugeno-Kang (TSK)

La estructura de un sistema borroso TSK es ligeramente distinta de la estructura típica vista en apartados anteriores, como podemos apreciar en la Figura 3.19. No es necesario el borrosificador, ya que las entradas al sistema son siempre numéricas, y su efecto sobre las salidas se calcula directamente mediante la aplicación de las ecuaciones anteriores. La salida de cada una de las reglas es un valor numérico, de forma que tampoco se requiere de un desborrosificador. Sin embargo, es imprescindible algún mecanismo para agregar todas las conclusiones parciales de las reglas en una única salida. Para ello se usa la suma ponderada o el promedio ponderado. En ocasiones se denomina a esta operación desborrosificador, sobre todo al promedio ponderado, que se le suele denominar desborrosificador centro promedio, pero no es estrictamente correcto, ya que su entrada no es un conjunto borroso.

Los sistemas borrosos TSK son una poderosa herramienta para el modelado de sistemas complejos y altamente no lineales, ya que son aproximadores universales tanto de la función [105][204][211][221] como de su derivada [109][110][135][159].

La utilización en los sistemas TSK de un consecuente polinómico permite aproximar funciones con gran precisión y con un número mucho menor de reglas que los sistemas de tipo Mamdani. No obstante, este aumento de precisión se consigue a costa de empeorar la interpretabilidad de las reglas.

Modelo de Tsukamoto

Modelo propuesto en 1979 por Tsukamoto [194] y que se caracteriza por representar el consecuente de cada regla borrosa mediante un conjunto borroso con una función de pertenencia monótona. La salida inferida por cada regla es un valor numérico inducido por el grado de activación de la regla. La salida global se toma como la media ponderada de todas las salidas parciales. Este tipo de modelo permite eliminar el consumo de tiempo correspondiente a la desborrosificación, sin embargo, no son modelos muy utilizados por no ser transparentes como los de Mamdani o Sugeno. El sistema de inferencia no sigue estrictamente la regla composicional de inferencia y la salida es siempre numérica, incluso cuando las entradas son borrosas.

3.5.4. Metodología para el diseño de sistemas borrosos

Se presentan aquí cinco principios fundamentales para construir un sistema borroso basado en reglas. Para ello se utilizará como ejemplo un caso de la vida real: Calificación de un alumno en función de su trabajo en una asignatura cuatrimestral.

Analizar el sistema a modelar

1. *Identificar las entradas y salidas:* En cualquier proceso de diseño se debe comenzar por describir el sistema general, tanto las partes que sean borrosas como las que no lo sean. Se debe hacer especial análisis en qué variables serán entradas y cuáles salidas, considerando la disponibilidad para la medición, etc.
2. *Analizar y simplificar el problema:* Una vez que se hayan definido las entradas y salidas, se debe realizar un proceso iterativo de forma que se analicen las posibles alternativas que puedan simplificar el planteamiento del sistema a diseñar.

Se debe realizar una descomposición funcional del sistema en subsistemas (con sus correspondientes entradas y salidas) que se puedan modelar fácilmente de forma independiente. Si un subsistema no es fácilmente modelable, se descompondrá en sucesivos subsistemas.

3. *Identificar los subsistemas borrosos:* Una vez que el sistema ha sido funcionalmente descompuesto, se deben identificar aquellos subsistemas que se puedan modelar utilizando la lógica borrosa. Un buen candidato para un

subsistema borroso es aquel en que el flujo de la información sea impreciso, ruidoso o pueda ser descrito lingüísticamente.

En nuestro ejemplo las variables de entrada al sistema de inferencia borroso son dos, que denominaremos *asistencia* (representa el número de veces que el alumno ha asistido a clase en el cuatrimestre) y *práctica* (calificación que el alumno ha obtenido en los trabajos prácticos realizados a lo largo del cuatrimestre), y la salida es una que llamaremos *calificación* (nota definitiva que obtiene el alumno por su trabajo en la asignatura).

Definir los universos de entrada y salida

1. *Definir el universo de discurso*: De forma general, el universo de discurso se define en el rango de interés de la variable característica del sistema que va a ser diseñado. Se debe tener especial cuidado en que el rango no sea muy pequeño, puesto que entonces pueden procesarse datos fuera de escala que deteriorarían el comportamiento general del sistema.

Si por el contrario, el intervalo del universo de discurso que se selecciona es muy grande, puede ocurrir que la respuesta del sistema sea plana (lo que significa que para cambios en la entrada no existirán cambios significativos en la salida). Esto último suele ocurrir en los extremos del intervalo definido, donde la función de pertenencia no captura la dinámica del sistema.

El universo de discurso de la variable de salida debe ser cuidadosamente seleccionado, debido a que una desproporción en la definición de la función de pertenencia (especialmente en los extremos) puede sobrecargar la masa del conjunto borroso de salida sobre el cual se aplicará el proceso de desborrosificación.

2. *Escalar o normalizar el universo de discurso*: El escalado o normalización del universo de discurso puede ser necesario o aconsejable en ciertas aplicaciones. Por ejemplo, si una de las variables de entrada es la temperatura y un sensor debe entregar un nivel de voltaje proporcional a la temperatura de entrada; como el nivel de voltaje deber ser convertido a un código digital equivalente, puede ser deseable normalizar el universo de discurso de la variable temperatura en un código digital.

Si en un cuatrimestre hay 15 semanas de clase, la asistencia del alumno vendrá dada por un número entre 0 y 30, suponiendo 2 clases a la semana. La nota obtenida en prácticas será un valor entre 0 y 10, ¿qué calificación final obtendrá el alumno? Esta calificación vuelve a ser un valor entre 0 y 10.

Determinar el número y distribución de las funciones de pertenencia

De forma general, la mayoría de las aplicaciones utilizan un número impar de términos lingüísticos (de 3 a 7); el hecho de utilizar un mínimo de 3 radica en que el lenguaje humano considera al hacer un juicio dos extremos y un medio entre ellos; no se utilizan más de 7 debido a que el hombre posee una memoria corta (puede procesar hasta 7 símbolos a la vez) cuando debe interpretar una figura técnica a través de un término descriptor y el hecho de representar un número impar se debe a que la mayoría de los términos lingüísticos se definen de forma simétrica con un medio entre los extremos.

Para escoger entre 3, 5 o 7 términos lingüísticos, se puede realizar:

1. Formular un conjunto de reglas básicas que pueden dar una idea de qué términos (y cuántos) se necesitan para definir un conjunto de reglas.
2. Si realizar el paso anterior es difícil, se pueden definir tres funciones de pertenencia para cada variable de entrada y cinco para las variables de salida, lo cual constituye el mínimo número de términos lingüísticos para las variables borrosas en la mayoría de las aplicaciones. Posteriormente se podrán adicionar nuevos términos en los sucesivos pasos del diseño.

Con respecto a la distribución de las funciones de pertenencia se debe tener en cuenta que cuando se desea una alta sensibilidad a las variaciones en cierta región de la variable de entrada, debe incrementarse la densidad de las funciones de pertenencia en tal región, por ejemplo cuando se requiere una mayor acción de control.

Es deseable que los términos lingüísticos sean representados a través de un conjunto de funciones de pertenencia que sean: enteras, consistentes, no indefinidas y normales.

Entereza: Si para cualquier valor del universo de discurso al menos uno de los términos lingüísticos alcanza un grado de pertenencia desigual a cero.

Consistencia: Si para cualquier entrada en la cual el término lingüístico alcanza su máximo grado de pertenencia, los restantes términos alcanzan un grado de pertenencia cero.

Indefinición: Un sistema borroso será indefinido si existe un punto en su dominio en el cual todas las reglas tienen un grado de activación cero en ese punto.

Normalidad: Un término lingüístico será normal si la función de pertenencia que lo representa es normal.

Para cumplir algunas de las anteriores propiedades se pueden seguir los siguientes pasos:

1. Para cada término lingüístico, definir un valor dentro del universo de discurso que mejor lo represente y fijar un grado de pertenencia uno a ese punto.
2. Definir un grado de pertenencia cero en los valores del universo de discurso donde los restantes términos lingüísticos tienen un grado de pertenencia uno.
3. No definir un grado de pertenencia uno para dos términos lingüísticos en un mismo universo de discurso.

Para seleccionar el tipo de funciones de pertenencia se debe tener presente si el conjunto de reglas representan la intuición humana o forman parte de una estructura matemática que pretende modelar un sistema a partir de datos de entrada-salida.

Para representar un determinado conocimiento a través de un conjunto de reglas, las funciones de pertenencia más utilizadas son la singleton, triangulares, trapezoidales y Gaussianas.

Para el ejemplo se han utilizado funciones triangulares en el centro del universo de discurso y trapezoidales en los extremos. En la Figura 3.21 se puede ver la definición de la variable lingüística *asistencia*. La variable *práctica* se muestra en la Figura 3.20, mientras la variable de salida *calificación* se representa en la Figura 3.22.

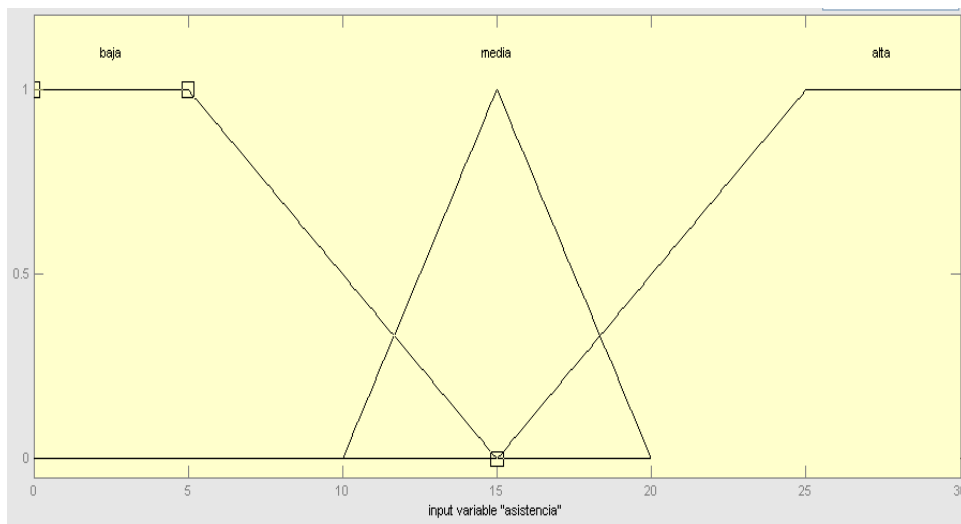


Figura 3.21 Representación de la variable de entrada *asistencia*

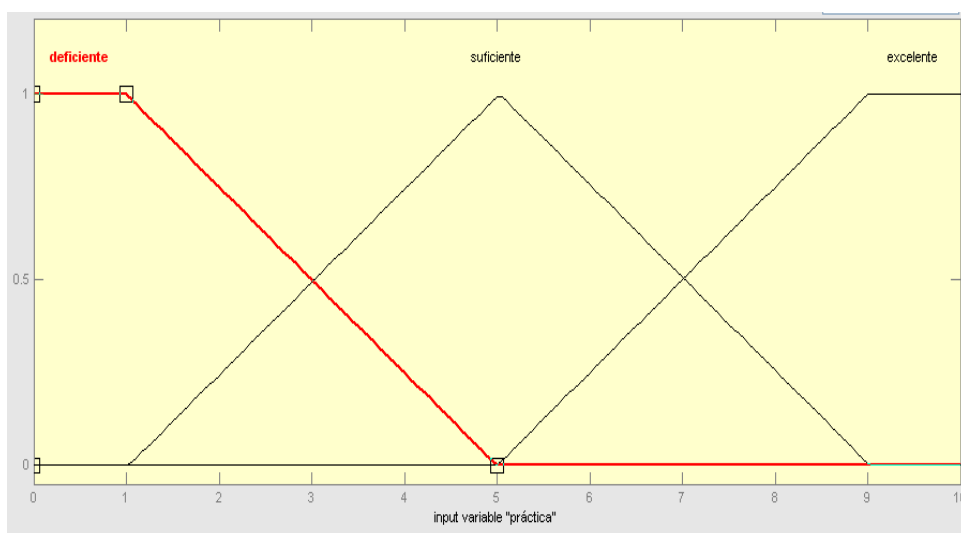


Figura 3.20 Representación de la variable de entrada *práctica*

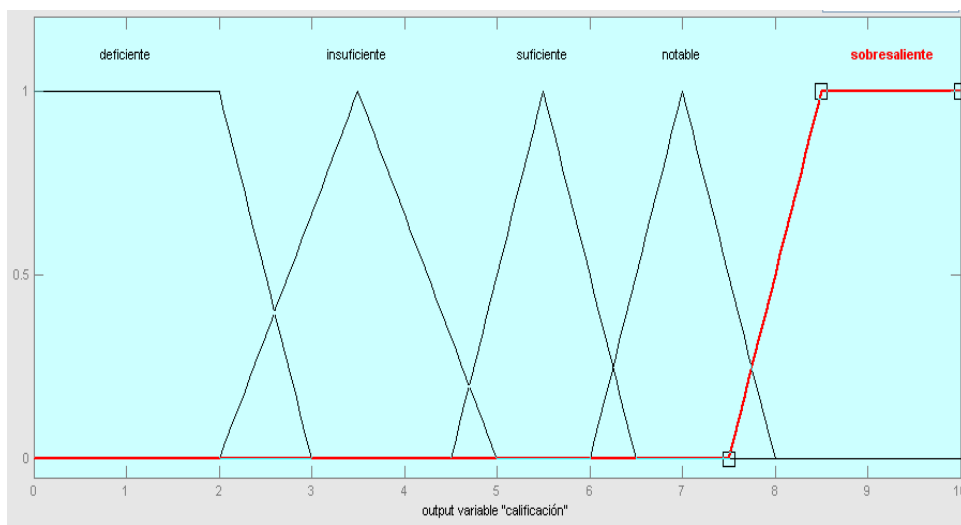


Figura 3.22 Representación de la variable de salida *calificación*

Crear la base de reglas

A la hora de crear la base de reglas es conveniente realizar los siguientes pasos:

1. Describir las reglas que sean obvias: Cuando el sistema tenga dos entradas y una salida se puede preparar una matriz a la que se le denomina Memoria Asociativa Borrosa (MAB). Para implementar una MAB, se colocan en la columna una de las variables de entrada y en la fila la otra variable de entrada (Tabla 3.5). Las celdas de la matriz creada se rellena con los términos lingüísticos de la variable de salida y cada celda representará una regla borrosa.

		<i>asistencia</i>		
		baja	media	alta
<i>práctica</i>	deficiente	deficiente		insuficiente
	suficiente		suficiente	
	excelente		notable	sobresaliente

Tabla 3.5 Memoria Asociativa Borrosa para las reglas del ejemplo

La MAB puede utilizarse para crear un conjunto de reglas con más de dos entradas y una salida. Para ello se crea una matriz con dos entradas y una salida, manteniendo las restantes entradas constantes. Por ejemplo, si a un conjunto de reglas con dos variables de entrada y una de salida se le incorporase otra variable de entrada con tres términos lingüísticos, entonces deberían crearse tres matrices como la que se muestra en la Tabla 3.5, una para cada término lingüístico de la nueva variable.

Sin embargo, la utilización de la MAB es poco práctica en aquellas aplicaciones que posean muchas entradas y salidas, debido a que se requerirían muchas matrices asociativas borrosas que describan todas las posibles combinaciones de entradas y salidas (un sistema con cinco entradas y tres términos lingüísticos cada entrada, con tres salidas tendría un número de reglas igual a $3^5 * 3 = 729$ reglas).

2. Describir las reglas que sean vagas o menos obvias, pero intuitivamente correctas: Una vez descritas las reglas obvias, debemos identificar las reglas que sean vagas. Para ello, un buen auxilio puede ser el considerar en la MAB que no debe haber transiciones bruscas entre celdas.

Un caso especial dentro de las reglas menos obvias es la incorporación de una aserción. Por una aserción se entiende una regla con antecedente pero sin consecuente, debido a lo cual ésta siempre tiende a activarse e influir en el espacio de salida truncado resultante de aplicar un método de inferencia.

La aserción influye sobre el funcionamiento total del sistema, limitando el espacio de salida que define la respuesta del sistema.

Según la MAB el conjunto de reglas obvias para el ejemplo serían:

Si asistencia es baja y práctica es deficiente entonces calificación es deficiente
Si asistencia es media y práctica es suficiente entonces calificación es suficiente
Si asistencia es media y práctica es excelente entonces calificación es notable (3.47)
Si asistencia es alta y práctica es deficiente entonces calificación es insuficiente
Si asistencia es alta y práctica es excelente entonces calificación es sobresaliente

Ajustar el conjunto de reglas

Es el paso más complejo en el diseño del sistema borroso y se basa en observar la salida del modelo implementado. Este paso puede incluir:

1. Ajuste de las reglas y funciones de pertenencia para cada variable del sistema borroso: Entre las acciones que se pueden realizar en esta fase se encuentran la incorporación de α -cortes a algunas de las funciones de pertenencia, limitar la contribución de algunas reglas o aplicar modificadores lingüísticos.
2. Reevaluación del mecanismo de implicación y composición (método de inferencia): La adecuada selección del mecanismo de inferencia depende de la aplicación y de la experiencia del diseñador del sistema borroso.
3. Reevaluación del método de desborrosificación: La desborrosificación determina el valor escalar que representa al conjunto borroso de salida. Constituye, pues, un factor de vital importancia a la hora de ajustar el modelo. Un elemento que puede utilizarse para ajustar el mecanismo de

desborrosificación es comparar la salida del modelo con respecto a valores que este debe entregar, de acuerdo a los criterios del experto. Tomando algunos valores de salida como referencia podemos seleccionar el método de desborrosificación que más se aproxime a los mismos.

Examinar la conducta del modelo

En este paso se trata de identificar los comportamientos indeseables en el modelo ante diferentes combinaciones de variables de entrada. Se debe prestar particular atención a las siguientes áreas de las variables de entrada:

1. Regiones terminales del universo de discurso
2. Valores extremos del soporte de las funciones de pertenencia
3. Zona de solape entre funciones de pertenencia
4. Valores en los antecedentes en los que se active el mayor número de reglas posible

3.6. Conclusiones

En este capítulo se ha visto como, tomando como base la imprecisión del razonamiento humano pero bajo un planteamiento matemático, la lógica borrosa es capaz de generar la respuesta a una situación basándose en el conocimiento adquirido de ésta, aunque éste sea inexacto o incompleto.

Como se ha visto, la lógica borrosa se basa en el concepto de conjunto borroso. En este capítulo se han estudiado dichos conjuntos, tanto la forma de definirlos a partir de su función de pertenencia, como algunas de las operaciones que se pueden realizar con ellos. Posteriormente se ha estudiado el concepto de variable lingüística, introducido por Zadeh en [216], el cual permite transformar el lenguaje natural en conocimiento cuantitativo utilizable desde el punto de vista matemático, pero sin las restricciones que impone el formalismo preciso de la lógica clásica.

Se han presentado los principales tipos de sistemas borrosos existentes, así como los métodos de inferencia de cada uno de ellos. La descripción del modelo borroso de tipo Mamdani puede consultarse en las referencias [59][116][156][212][220], en las referencias [104][106] las de los modelos de tipo SAM y en [194] las de los modelos de tipo

Tsukamoto. Se ha prestado especial atención a los sistemas Takagi-Sugeno-Kang (TSK), aproximadotes universales de cualquier función y su derivada, tal como se demuestra en las referencias [105][110][184][204][211][221], ya que estos serán la herramienta básica del resto de capítulos de este texto.

Por último, se presenta una metodología resumida en cinco principios fundamentales para construir un modelo borroso basado en reglas.

De entre las referencias incluidas en el capítulo cabe destacar [216], donde Zadeh introdujo el concepto de variable lingüística, y los trabajos [183] y [184] de Takagi, Sugeno y Kang porque en base a ellos crearon los modelos borrosos de tipo TSK.

4. Adaptación paramétrica en sistemas borrosos

4.1. Introducción

El núcleo de los sistemas adaptativos borrosos lo constituye la adaptación de parámetros, a partir de un algoritmo de aprendizaje cuya ley de adaptación es similar a la empleada en las redes neuronales.

Los sistemas adaptativos deben tener algunas o todas de las siguientes características:

1. Se deben adaptar automáticamente de forma tal que puedan abordarse los problemas asociados al cambio en el medio en que se plantea el problema y los cambios en los requerimientos del sistema.
2. Deben ser entrenados para realizar tareas específicas de filtrado y toma de decisiones, es decir, deben ser programados por un proceso de entrenamiento. Debido a que los sistemas adaptativos no requieren de los procedimientos de síntesis elaborados que usualmente necesitan los sistemas no adaptativos, se puede decir que son auto-diseñados en cierto grado.
3. Deben extrapolar un modelo de conducta que resuelva nuevas situaciones debido a que son entrenados con un número finito y representativo de patrones de entrenamiento.
4. Una vez entrenados, son más complejos de analizar que los sistemas no adaptativos, pero ofrecen la posibilidad de incrementar sustancialmente la efectividad del sistema cuando las características de la señal de entrada son variantes en el tiempo y representan una dinámica no modelada.

Los mecanismos de adaptación de parámetros han sido utilizados con éxito sobre diferentes funciones no lineales que pueden ser representadas a través de un sistema borroso equivalente [202][203][119][71][184][154][208][170]. Sin embargo, no se aprecia un esfuerzo por mantener las experiencias del diseño de las reglas durante la fase imprecisa. Resultaría muy interesante incorporar al mecanismo de adaptación, técnicas de ajuste de los parámetros para no perder interpretabilidad.

Parece que siempre han estado reñidos los términos interpretabilidad y precisión; esto es, cuanto más interpretabilidad menos precisión y viceversa. Esta Tesis pretende aunar ambas cualidades de modo que se puedan conseguir modelos neuro-borrosos en los que la precisión y la interpretabilidad de sus reglas puedan ir de la mano. Para ello se propone una nueva metodología que permite ajustar las funciones de pertenencia haciendo sus reglas interpretables y dotadas de semántica.

Los Sistemas de Inferencia Borrosa (SIB), en inglés Fuzzy Inference Systems (FIS) pueden ser diseñados a partir del conocimiento de un experto o a partir de datos. Cuando el sistema es complejo, los SIB basados únicamente en el conocimiento de un experto suelen estar faltos de precisión. Por ello, la precisión es probablemente el principal incentivo para usar reglas borrosas inferidas de forma automática a partir de datos. En la literatura hay disponibles muchos métodos para diseñar SIB a partir de datos. Su eficiencia suele estar caracterizada mediante un índice de funcionamiento. Sin embargo, la cooperación y entendimiento humanos necesitan otro criterio: la interpretabilidad. En el campo de la lógica borrosa, implícitamente se asume que las reglas borrosas son, por naturaleza, fáciles de interpretar. Sin embargo, ¿es posible definir la interpretabilidad de forma clara? Ciertamente, no existe un índice numérico para medir la interpretabilidad de una regla o base de reglas, aunque todos los autores coinciden en una serie de condiciones que parecen muy claras e intuitivas y que se resumen a continuación:

1) La partición borrosa debe ser legible, en el sentido de que los conjuntos borrosos puedan ser interpretados con etiquetas lingüísticas con significado semántico propio.

2) Las etiquetas lingüísticas del problema estudiado deben ser significativas para los expertos, con el fin de poder compararlas entre ellas y dirigir el descubrimiento del conocimiento.

3) Cada conjunto borroso debe tener un rango definido en su universo de discurso.

4) Las funciones de pertenencia deben ser claramente diferentes.

La construcción de un sistema neuro-borroso consiste en dos fases principales [208]: la identificación de la estructura del modelo borroso (structure identification) y la estimación de los parámetros del modelo borroso (parameter identification). Los parámetros son identificados (o adaptados) usando diferentes métodos: el gradiente, Gauss-Newton, Levenberg-Marquardt, etc. [139][148][146][202][205].

Probablemente, "Adaptive-Network-based Fuzzy Inference Systems (ANFIS)", [91] es el método más usado para el entrenamiento de sistemas neuro-borrosos aunque existen otros paquetes comerciales como: Peltarion Synapse, FuzzyTECH, etc. La aplicación de técnicas de redes neuronales para construir reglas borrosas tiene el inconveniente de que se pierden algunos de los más importantes rasgos de los sistemas borrosos. Un problema común en muchos sistemas neuro-borrosos es que la interpretabilidad del sistemas borroso se deteriora, lo cual se debe a que los algoritmos de entrenamiento importados de las redes neuronales [115] no prestan atención a la semántica de las reglas borrosas. No obstante, en la literatura hay algunos ejemplos que intentan mitigar la falta de interpretabilidad de los modelos neuro-borrosos; así por ejemplo, en [182] se presenta un algoritmo de entrenamiento basado en el método de gradiente descendente donde las funciones de pertenencia no son independientes unas de otras pero las particiones borrosas si lo son. Se trata de mantener los parámetros que van a ser adaptados en un intervalo del universo de discurso basándose en un cierto criterio.

En la literatura no hay muchos ejemplos de construcción de modelos borrosos a partir de datos en los que la interpretabilidad del modelo sea el objetivo principal a cumplir, incluso sacrificando la precisión [152][19][79][43][96]. Toda esta bibliografía utiliza el grado de similaridad (según diferentes definiciones) para incrementar la interpretabilidad de los modelos. En este trabajo se aborda el problema de la interpretabilidad desde otro punto de vista [197], en concreto a partir de la razón de solape, un concepto definido por primera vez por Marsh [129]. Se da incluso un paso más y se propone una nueva metodología para ajustar los parámetros adaptados en un sistema neuro-borroso minimizando una función de coste. Así, se define y calcula el área de solape (el área total contenida en funciones de pertenencia vecinas) y se propone un algoritmo de ajuste considerando este área.

4.2. Adaptación de una base de reglas borrosas

Como se ha mencionado previamente, para la obtención de modelos borrosos, se puede representar el conocimiento de un experto a partir de datos de entrada-salida. Utilizando ese conjunto de datos se pueden obtener los parámetros que definen las funciones de pertenencia de las variables de entrada y los consecuentes de las variables de salida, utilizando un método de adaptación paramétrica.

El núcleo del modelo borroso lo constituye la base de reglas borrosas R , formada por el conjunto:

$$R = \{R^1, R^2, \dots, R^l, \dots, R^M\} \quad (4.1)$$

donde cada regla tiene la forma:

$$R^l : \text{SI } x_1 \text{ es } F_1^l \text{ y } x_2 \text{ es } F_2^l \text{ y } \dots \text{ y } x_n \text{ es } F_n^l \text{ ENTONCES } y \text{ es } G^l \quad (4.2)$$

siendo $l = 1, 2, \dots, M$ el número de reglas, F_i^l los conjuntos borrosos en los universos de discurso $U_i \subset R$ de las variables de entrada, G^l un subconjunto borroso en el universo de discurso de la variable de salida $V \subset R$, $x = (x_1, \dots, x_n) \in U_1 \times \dots \times U_n$ para las $i = 1, 2, \dots, n$ variables de entrada e $y \in V$ la variable de salida.

Uno de los métodos más populares de adaptación paramétrica es el utilizado por Wang [202], que utiliza un desborrosificador centro promedio, con método de inferencia producto. Aplicando al conjunto de reglas de la forma (4.2) este método de adaptación se obtiene:

$$f(x) = \frac{\sum_{l=1}^M y^l \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right]}{\sum_{l=1}^M \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right]} \quad (4.3)$$

donde $\mu_{F_i^l}(x_i, \theta_i^l)$ representa a la función de pertenencia que se define a partir de un conjunto borroso F_i^l . Esta función de pertenencia puede ser de Gauss, triangular, campana, etc. Las características de la función de pertenencia quedan definidas por θ_i^l , que representa los parámetros (centro, ancho, etc.) que definen la función de pertenencia $\mu_{F_i^l}$. y^l es el centro del conjunto borroso G^l , es decir, el punto en V para el que la función de pertenencia G^l alcanza su máximo valor. Se asume que $\mu_{G^l}(y^l) = 1$.

Si denominamos $\xi^l(x) = \prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l)$ obtenemos la expresión:

$$\hat{y}(x) = f(x) = \frac{\sum_{l=1}^M y^l \xi^l(x)}{\sum_{l=1}^M \xi^l(x)} \quad (4.4)$$

Como se puede observar en la Figura 4.1, un sistema borroso cuyo modelo matemático equivalente viene dado por la expresión (4.4), puede ser representado como una red neuronal de flujo directo de tres capas hacia delante [147][214], es decir, un perceptrón de 3 capas [202]. Naturalmente, el comportamiento de los nodos y las sinapsis en un sistema neuro-borroso no son iguales que en el perceptrón, sin embargo, pueden emplearse los mismos algoritmos de adaptación.

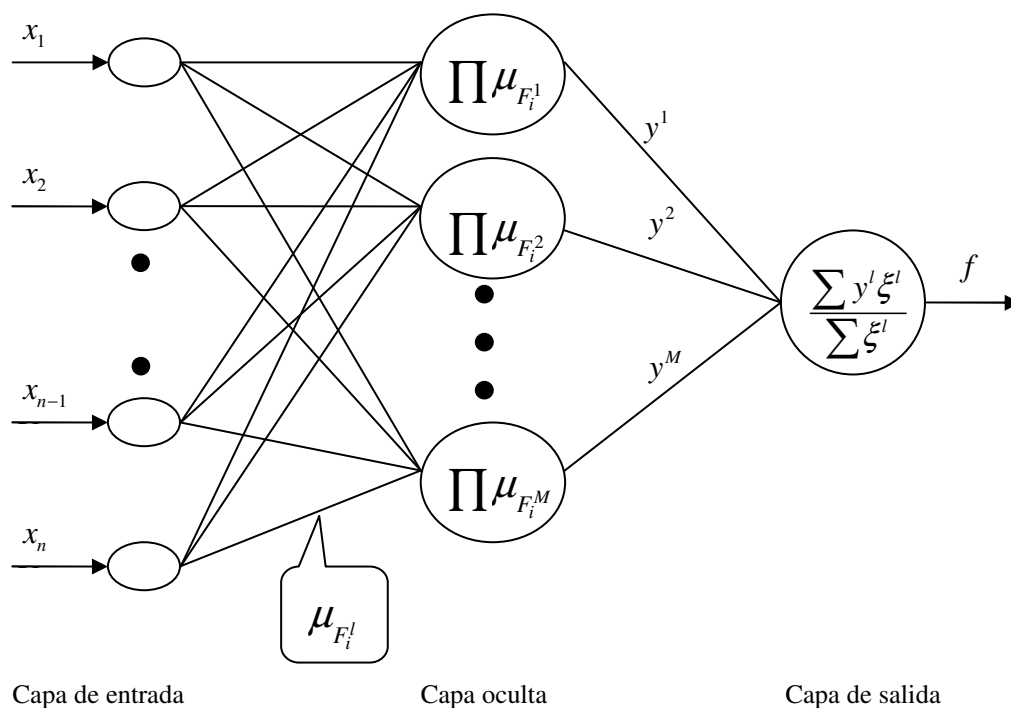


Figura 4.1 Red neuronal equivalente al sistema borroso

Con el fin de adaptar los parámetros del sistema neuro-borroso definido en la Figura 4.1, se define una función de coste. Aunque existen diferentes funciones de coste, dependiendo fundamentalmente de si el mecanismo de adaptación es secuencial o por lotes. Aquí se usará será el error cuadrático medio, que responde a:

$$J = \frac{1}{2} (y(x) - \hat{y}(x))^2 \quad (4.5)$$

donde $y(x)$ es la salida real del sistema que se pretende modelar e $\hat{y}(x)$ es la salida estimada por el modelo.

A partir de la función de ajuste que implementa el método de gradiente descendente [69], se pueden obtener las expresiones de adaptación paramétrica de los antecedentes:

$$\theta_i^l(k+1) = \theta_i^l(k) - \eta \left. \frac{\partial J(k)}{\partial \theta_i^l(k)} \right|_k \quad (4.6)$$

Y de forma similar de los consecuentes de las reglas:

$$y^l(k+1) = y^l(k) - \eta \left. \frac{\partial J(k)}{\partial y^l(k)} \right|_k \quad (4.7)$$

Donde $\theta_i^l(k)$ representa el parámetro que va a ser adaptado y η es un factor de escala del término derivativo, denominado factor de aprendizaje del algoritmo, y que afecta, entre otros, al tiempo de convergencia y estabilidad del modelo resultante.

Las expresiones (4.6) y (4.7) constituyen el fundamento para obtener una base de reglas como modelo no lineal borroso de un sistema a partir de datos de entrada-salida.

4.2.1. La adaptación de parámetros del antecedente de las reglas

Para adaptar los parámetros del antecedente de las reglas se tendrá que aplicar (4.6), por lo que se hace necesario calcular:

$$\left. \frac{\partial J(k)}{\partial \theta_i^l(k)} \right|_k = \left. \frac{\partial \left(\frac{1}{2} (y(x) - \hat{y}(x))^2 \right)}{\partial \theta_i^l(k)} \right|_k \quad (4.8)$$

Si se denomina:

$$A = \sum_{l=1}^M y^l \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \quad (4.9)$$

y

$$B = \sum_{l=1}^M \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \quad (4.10)$$

entonces:

$$\hat{y}(x) = \frac{A}{B} \quad (4.11)$$

Sustituyendo (4.11) en (4.8) resulta:

$$\left. \frac{\partial J(k)}{\partial \theta_i^l(k)} \right|_k = e \frac{\frac{\partial A}{\partial \theta_i^l(k)} B - A \frac{\partial B}{\partial \theta_i^l(k)}}{B^2} \quad (4.12)$$

donde

$$e = y(x) - \hat{y}(x) \quad (4.13)$$

Nótese que, en principio, el orden de las variables para obtener el error no es significativo, dado que el mismo está elevado al cuadrado (4.5).

4.2.2. Función de pertenencia de Gauss

En el caso de utilizar funciones de pertenencia de Gauss, éstas se definen por:

$$\mu_{F_i^l}(x_i, \theta_i^l) = \exp \left[- \left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \quad (4.14)$$

donde los parámetros que caracterizan a la función son $\theta_i^l = \gamma_i^l, \beta_i^l$, centros y anchos de las funciones respectivamente.

La función estimada, si las funciones utilizadas son funciones de pertenencia de Gauss, se puede representar por:

$$\hat{y}(x) = f(x) = \frac{\sum_{l=1}^M y^l \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \gamma_i^l, \beta_i^l) \right]}{\sum_{l=1}^M \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \gamma_i^l, \beta_i^l) \right]} = \frac{\sum_{l=1}^M y^l \left[\prod_{i=1}^n \exp \left[- \left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right]}{\sum_{l=1}^M \left[\prod_{i=1}^n \exp \left[- \left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right]} \quad (4.15)$$

por lo que:

$$A = \sum_{l=1}^M y^l \left[\prod_{i=1}^n \exp \left[- \left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right] \quad \text{y} \quad B = \sum_{l=1}^M \left[\prod_{i=1}^n \exp \left[- \left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right]$$

Cuando se adaptan los centros de la función de Gauss (γ_i'),

$$\left. \frac{\partial J(k)}{\partial \gamma_i'(k)} \right|_k = e \frac{\frac{\partial A}{\partial \gamma_i'(k)} B - A \frac{\partial B}{\partial \gamma_i'(k)}}{B^2} \quad (4.16)$$

$$\frac{\partial A}{\partial \gamma_i'(k)} = y^l \frac{\frac{\partial \prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right]}{\partial \gamma_i'}}{\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right]} = 2 y^l \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right] \right] \frac{(x_i - \gamma_i')}{(\beta_i')^2} \quad (4.17)$$

$$\frac{\partial B}{\partial \gamma_i'(k)} = \frac{\frac{\partial \prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right]}{\partial \gamma_i'}}{\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right]} = 2 \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right] \right] \frac{(x_i - \gamma_i')}{(\beta_i')^2} \quad (4.18)$$

Sustituyendo (4.17) y (4.18) en (4.16) se obtiene:

$$\left. \frac{\partial J(k)}{\partial \gamma_i'(k)} \right|_k = e \frac{2 y^l \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right] \right] \frac{(x_i - \gamma_i')}{(\beta_i')^2} B - 2 \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right] \right] \frac{(x_i - \gamma_i')}{(\beta_i')^2} A}{B^2}$$

Simplificando:

$$\left. \frac{\partial J(k)}{\partial \gamma_i'(k)} \right|_k = 2e \frac{\left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right] \right] \frac{(x_i - \gamma_i')}{(\beta_i')^2} (y^l - \hat{y}(x))}{B} \quad (4.19)$$

Por lo que la ley general de adaptación de parámetros (4.6) aplicada al centro de las funciones de Gauss (según (4.19)) es:

$$\gamma_i'(k+1) = \gamma_i'(k) - 2\eta e \frac{\left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i'}{\beta_i'} \right)^2 \right] \right] \frac{(x_i - \gamma_i')}{(\beta_i')^2} (y^l - \hat{y}(x))}{B} \quad (4.20)$$

De igual forma, si se adaptan los anchos de la función de Gauss (β_i'),

$$\left. \frac{\partial J(k)}{\partial \beta_i^l(k)} \right|_k = e \frac{\frac{\partial A}{\partial \beta_i^l(k)} B - A \frac{\partial B}{\partial \beta_i^l(k)}}{B^2} \quad (4.21)$$

$$\frac{\partial A}{\partial \beta_i^l(k)} = y^l \frac{\frac{\partial \prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right]}{\partial \beta_i^l}}{\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right]} = 2y^l \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right] \frac{(x_i - \gamma_i^l)^2}{(\beta_i^l)^3} \quad (4.22)$$

$$\frac{\partial B}{\partial \beta_i^l(k)} = \frac{\frac{\partial \prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right]}{\partial \beta_i^l}}{\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right]} = 2 \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right] \frac{(x_i - \gamma_i^l)^2}{(\beta_i^l)^3} \quad (4.23)$$

Sustituyendo (4.22) y (4.23) en (4.21) se obtiene:

$$\left. \frac{\partial J(k)}{\partial \beta_i^l(k)} \right|_k = e \frac{2y^l \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right] \frac{(x_i - \gamma_i^l)^2}{(\beta_i^l)^3} B - 2 \left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right] \frac{(x_i - \gamma_i^l)^2}{(\beta_i^l)^3} A}{B^2}$$

Simplificando:

$$\left. \frac{\partial J(k)}{\partial \beta_i^l(k)} \right|_k = 2e \frac{\left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right] \frac{(x_i - \gamma_i^l)^2}{(\beta_i^l)^3} (y^l - \hat{y}(x))}{B} \quad (4.24)$$

Por lo que la ley general de adaptación de parámetros (4.6) aplicada al ancho de las funciones de Gauss (según (4.24)) es:

$$\beta_i^l(k+1) = \beta_i^l(k) - 2\eta e \frac{\left[\prod_{i=1}^n \exp \left[-\left(\frac{x_i - \gamma_i^l}{\beta_i^l} \right)^2 \right] \right] \frac{(x_i - \gamma_i^l)^2}{(\beta_i^l)^3} (y^l - \hat{y}(x))}{B} \quad (4.25)$$

4.2.3. Función de pertenencia triangular

Dentro de las funciones triangulares se encuentran un tipo particular de las mismas que resultan especialmente interesantes para su estudio. Este tipo particular de funciones de pertenencia triangulares son las que generan triángulos isósceles y/o equiláteros.

La expresión que define este tipo de funciones triangulares es:

$$\mu_{F_i^l}(x_i, \theta_i^l) = \begin{cases} 1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} & \text{para } \gamma_i^l - \frac{\beta_i^l}{2} < x_i < \gamma_i^l + \frac{\beta_i^l}{2} \\ 0 & \text{en otro caso} \end{cases} \quad (4.26)$$

donde los parámetros que caracterizan a la función son $\theta_i^l = \gamma_i^l, \beta_i^l$, centros y anchos de las funciones respectivamente.

Para este tipo de funciones de pertenencia triangulares la función estimada queda como:

$$\hat{y}(x) = f(x) = \frac{\sum_{l=1}^M y^l \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \gamma_i^l, \beta_i^l) \right]}{\sum_{l=1}^M \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \gamma_i^l, \beta_i^l) \right]} = \frac{\sum_{l=1}^M y^l \left[\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \right]}{\sum_{l=1}^M \left[\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \right]} \quad (4.27)$$

Si se pretenden adaptar los centros (γ_i^l) de las funciones triangulares, siendo:

$$A = \sum_{l=1}^M y^l \left[\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \right] \quad \text{y} \quad B = \sum_{l=1}^M \left[\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \right]$$

entonces:

$$\left. \frac{\partial J(k)}{\partial \gamma_i^l(k)} \right|_k = e \frac{\frac{\partial A}{\partial \gamma_i^l(k)} B - A \frac{\partial B}{\partial \gamma_i^l(k)}}{B^2} \quad (4.28)$$

$$\frac{\partial A}{\partial \gamma_i^l(k)} = y^l \frac{\partial \prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right)}{\partial \gamma_i^l} = -2y^l \frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \operatorname{sgn}(x_i - \gamma_i^l)}{1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l}} \quad (4.29)$$

$$\frac{\partial B}{\partial \gamma'_i(k)} = \frac{\partial \prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right)}{\partial \gamma'_i} = -2 \frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right) \operatorname{sgn}(x_i - \gamma'_i)}{1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \beta'_i} \quad (4.30)$$

donde el diferencial de la función no lineal módulo se representa por el signo del argumento.

Sustituyendo (4.29) y (4.30) en (4.28) y desarrollando se obtiene:

$$\left. \frac{\partial J(k)}{\partial \gamma'_i(k)} \right|_k = -2e \frac{\left[\frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right)}{\left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right)} \right] \frac{\operatorname{sgn}(x_i - \gamma'_i)}{\beta'_i} (y^l - \hat{y}(x))}{B} \quad (4.31)$$

Por lo que la ley general de adaptación de parámetros (4.6) aplicada a los centros de las funciones triangulares (según (4.31)) es:

$$\gamma'_i(k+1) = \gamma'_i(k) + 2\eta e \frac{\left[\frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right)}{\left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right)} \right] \frac{\operatorname{sgn}(x_i - \gamma'_i)}{\beta'_i} (y^l - \hat{y}(x))}{B} \quad (4.32)$$

De igual forma, si se adaptan los anchos de las funciones triangulares (β'_i),

$$\left. \frac{\partial J(k)}{\partial \beta'_i(k)} \right|_k = e \frac{\frac{\partial A}{\partial \beta'_i(k)} B - A \frac{\partial B}{\partial \beta'_i(k)}}{B^2} \quad (4.33)$$

$$\frac{\partial A}{\partial \beta'_i(k)} = y^l \frac{\partial \prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right)}{\partial \beta'_i} = y^l \frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \right) \left(\frac{2|x_i - \gamma'_i|}{\beta'_i} \right)}{1 - \frac{2|x_i - \gamma'_i|}{\beta'_i} \beta'_i} \quad (4.34)$$

$$\frac{\partial B}{\partial \beta_i^l(k)} = \frac{\partial \prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right)}{\partial \beta_i^l} = \frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \left(\frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right)}{1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l}} \quad (4.35)$$

Sustituyendo (4.34) y (4.35) en (4.33) y simplificando se obtiene:

$$\left. \frac{\partial J(k)}{\partial \beta_i^l(k)} \right|_k = e^{-\frac{\left[\frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \right] \left(\frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) (y^l - \hat{y}(x))}{\left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \frac{\beta_i^l}{\beta_i^l}}}} B} \quad (4.36)$$

Por lo que la ley general de adaptación de parámetros (4.6) aplicada a los anchos de las funciones triangulares (según (4.36)) es:

$$\beta_i^l(k+1) = \beta_i^l(k) - \eta e^{-\frac{\left[\frac{\prod_{i=1}^n \left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \right] \left(\frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) (y^l - \hat{y}(x))}{\left(1 - \frac{2|x_i - \gamma_i^l|}{\beta_i^l} \right) \frac{\beta_i^l}{\beta_i^l}}} B} \quad (4.37)$$

4.2.4. La adaptación de parámetros del consecuente de las reglas

El consecuente de un conjunto de reglas aporta la influencia de la variable de salida sobre el sistema de inferencia borroso. Este consecuente puede ser una función singleton, una función de tipo TSK o una función no lineal. En cualquier caso, para adaptar los parámetros del consecuente de las reglas tendremos que aplicar (4.7), teniendo en cuenta el tipo de consecuente con el que se represente y^l .

Consecuente singleton

El consecuente representa un valor en el universo de discurso de la variable de salida. Se trata de una función de pertenencia cuyo grado de pertenencia es uno únicamente en el valor que define el consecuente y cero para cualquier otro valor de la variable de salida.

Para poder aplicar (4.7), se necesita calcular:

$$\left. \frac{\partial J(k)}{\partial y'(k)} \right|_k = \left. \frac{\partial \left(\frac{1}{2} (y(x) - \hat{y}(x))^2 \right)}{\partial y'(k)} \right|_k \quad (4.38)$$

Sea A la expresión (4.9), B la expresión (4.10) y e la expresión (4.13), entonces:

$$\left. \frac{\partial J(k)}{\partial y'(k)} \right|_k = e \frac{\frac{\partial A}{\partial y'(k)} B - A \frac{\partial B}{\partial y'(k)}}{B^2} \quad (4.39)$$

$$\frac{\partial A}{\partial y'(k)} = \prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \quad (4.40)$$

$$\frac{\partial B}{\partial y'(k)} = 0 \quad (4.41)$$

Si se sustituye (4.40) y (4.41) en (4.39), se obtiene:

$$\left. \frac{\partial J(k)}{\partial y'(k)} \right|_k = e \frac{\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l)}{B} \quad (4.42)$$

Por lo que la ley general de adaptación de la parte precedente (4.7) cuando el consecuente es singleton (según (4.42)) es:

$$\boxed{y'(k+1) = y'(k) - \eta e \frac{\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l)}{B}} \quad (4.43)$$

Consecuente TSK

Un consecuente TSK (su nombre se debe a que fue propuesto originalmente por Takagi, Sugeno y Kang) es un polinomio que, dependiente de las variables de entrada, genera una función de pertenencia singleton en el universo de discurso de la variable de salida, es decir:

$$y^l = a_0^l + a_1^l x_1 + \dots + a_i^l x_i + \dots + a_n^l x_n \quad (4.44)$$

La ley de adaptación de los parámetros consecuentes está definida por la expresión (4.6) dónde los parámetros θ_i^l son los a_i^l .

Se calcula:

$$\left. \frac{\partial J(k)}{\partial a_i^l(k)} \right|_k = \left. \frac{\partial \left(\frac{1}{2} (y(x) - \hat{y}(x))^2 \right)}{\partial a_i^l(k)} \right|_k \quad (4.45)$$

A queda definida por la expresión:

$$A = \sum_{l=1}^M (a_0^l + a_1^l x_1 + \dots + a_i^l x_i + \dots + a_n^l x_n) \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \quad (4.46)$$

B se corresponde con la expresión (4.10) y e con la expresión (4.13), por lo que:

$$\left. \frac{\partial J(k)}{\partial a_i^l(k)} \right|_k = e \frac{\frac{\partial A}{\partial a_i^l(k)} B - A \frac{\partial B}{\partial a_i^l(k)}}{B^2} \quad (4.47)$$

$$\frac{\partial A}{\partial a_0^l(k)} = \prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \quad \text{Para el término afín} \quad (4.48)$$

$$\frac{\partial A}{\partial a_i^l(k)} = \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] x_i \quad (4.49)$$

$$\frac{\partial B}{\partial a_i^l(k)} = 0 \quad (4.50)$$

Si se sustituye (4.48) y (4.50) en (4.47), se obtiene:

$$\left. \frac{\partial J(k)}{\partial a_0^l(k)} \right|_k = e \frac{\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l)}{B} \quad \text{Para el término afín} \quad (4.51)$$

Y si se sustituye (4.49) y (4.50) en (4.47), se obtiene:

$$\left. \frac{\partial J(k)}{\partial a_i^l(k)} \right|_k = e \frac{\left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] x_i}{B} \quad (4.52)$$

Por lo tanto, la ley de adaptación de la parte precedente cuando el consecuente es TSK (según (4.51) y (4.52)) es:

$$a_0^l(k+1) = a_0^l(k) - \eta e \frac{\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l)}{B} \quad \text{Para el término afín} \quad (4.53)$$

$$a_i^l(k+1) = a_i^l(k) - \eta e \frac{\left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] x_i}{B} \quad (4.54)$$

El consecuente TSK genera un estimador borroso complejo con una capacidad de aproximación superior al consecuente singleton en la medida que existen más grados de libertad (parámetros ajustables); no obstante, la posibilidad de analizar el resultado de la adaptación de los parámetros de acuerdo a la teoría de la lógica borrosa es difícil, por la dependencia explícita de la variable de salida con respecto a las variables de entrada.

El consecuente no lineal

El polinomio utilizado en el consecuente TSK puede ser sustituido por una función no lineal parametrizable, de esta forma se obtiene un consecuente no lineal.

Un ejemplo de consecuente no lineal puede ser:

$$y^l = \exp \left[- \left(\frac{x_1 - a_1^l}{b_1^l} \right)^2 \right] + \dots + \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] + \dots + \exp \left[- \left(\frac{x_n - a_n^l}{b_n^l} \right)^2 \right] \quad (4.55)$$

Sea

$$A = \sum_{l=1}^M \left[\sum_{i=1}^n \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] \right] \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \quad (4.56)$$

B se corresponde con la expresión (4.10) y e con la expresión (4.13), por lo que:

$$\left. \frac{\partial J(k)}{\partial a_i^l(k)} \right|_k = e \frac{\frac{\partial A}{\partial a_i^l(k)} B - A \frac{\partial B}{\partial a_i^l(k)}}{B^2} \quad (4.57)$$

$$\left. \frac{\partial J(k)}{\partial b_i^l(k)} \right|_k = e \frac{\frac{\partial A}{\partial b_i^l(k)} B - A \frac{\partial B}{\partial b_i^l(k)}}{B^2} \quad (4.58)$$

$$\frac{\partial A}{\partial a_i^l(k)} = 2 \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \left(\frac{x_i - a_i^l}{b_i^{2l}} \right) \quad (4.59)$$

$$\frac{\partial B}{\partial a_i^l(k)} = 0 \quad (4.60)$$

$$\frac{\partial A}{\partial b_i^l(k)} = 2 \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \left(\frac{(x_i - a_i^l)^2}{b_i^{3l}} \right) \quad (4.61)$$

$$\frac{\partial B}{\partial b_i^l(k)} = 0 \quad (4.62)$$

Sustituyendo (4.59) y (4.60) en (4.57), se tiene:

$$\left. \frac{\partial J(k)}{\partial a_i^l(k)} \right|_k = e \frac{2 \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \left(\frac{x_i - a_i^l}{b_i^{2l}} \right)}{B} \quad (4.63)$$

De igual forma, al sustituir (4.61) y (4.62) en (4.58), se obtiene:

$$\left. \frac{\partial J(k)}{\partial b_i^l(k)} \right|_k = e \frac{2 \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \left(\frac{(x_i - a_i^l)^2}{b_i^{3l}} \right)}{B} \quad (4.64)$$

La ley de adaptación de la parte precedente cuando el consecuente es no lineal de la forma (4.55) (según (4.63) y (4.64)) es:

$$a_i^l(k+1) = a_i^l(k) - \eta e \frac{2 \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \left(\frac{x_i - a_i^l}{b_i^{2l}} \right)}{B} \quad (4.65)$$

$$b_i^l(k+1) = b_i^l(k) - \eta e \frac{2 \exp \left[- \left(\frac{x_i - a_i^l}{b_i^l} \right)^2 \right] \left[\prod_{i=1}^n \mu_{F_i^l}(x_i, \theta_i^l) \right] \left(\frac{(x_i - a_i^l)^2}{b_i^{3l}} \right)}{B} \quad (4.66)$$

4.3. Conclusiones

La adaptación de parámetros es el núcleo de cualquier sistema adaptativo borroso. Se han representado muy diferentes funciones no lineales a través de sistemas borrosos tras utilizar mecanismos de adaptación de parámetros, sin embargo en estos mecanismos de adaptación se prima la precisión del modelo antes que la interpretabilidad del mismo.

En este capítulo se mencionan los principales trabajos que han tratado de aportar interpretabilidad al proceso de adaptación de parámetros [182][153][19][79][43][96]. No obstante, el problema de la interpretabilidad desde el punto de vista de esta Tesis no es abordado en ninguno de ellos.

De los diferentes métodos de adaptación paramétrica, el método desarrollado por Wang [202] ha sido el método elegido en esta Tesis. Se detalla el procedimiento de adaptación de parámetros tanto para el antecedente de las reglas como para los consecuentes, pero centrándonos en funciones gaussianas y funciones triangulares.

La adaptación de los consecuentes es estudiada tanto si nos encontramos con un consecuente singleton, de tipo TSK o un consecuente no lineal, aunque principalmente serán tratados los consecuentes de tipo TSK.

La principal referencia de este capítulo es [202], donde Wang desarrolla su método de adaptación paramétrica utilizando un desborrosificador centro promedio, con método de inferencia producto.

5. La Razón de Solape

5.1. Introducción

Como ya se mencionó en el capítulo anterior, cuando los parámetros que definen las funciones de pertenencia, centros y anchos, son el resultado de un método adaptativo, una adecuada distribución de las Funciones de Pertenencia (FP) debe ser considerada, para no perder interpretabilidad en el resultado. Marsh [129] estableció 4 reglas que deben cumplir las FP para considerar que están correctamente distribuidas en el universo de discurso:

1. Cada punto del universo de discurso debe pertenecer al dominio de al menos una FP, adicionalmente, cada punto no debería pertenecer al dominio de más de dos FP.
2. No deben existir dos FP con el mismo punto de máximo grado de pertenencia.
3. Cuando se solapen dos FP, la suma de los grados de pertenencia para cualquier punto en el solape debe ser menor o igual a 1.
4. El solape no debe cruzar el punto de máximo grado de pertenencia de cualquiera de las FP.

En la Figura 5.1 se puede apreciar una adecuada distribución de las FP en el universo de discurso, atendiendo a las cuatro reglas de Marsh.

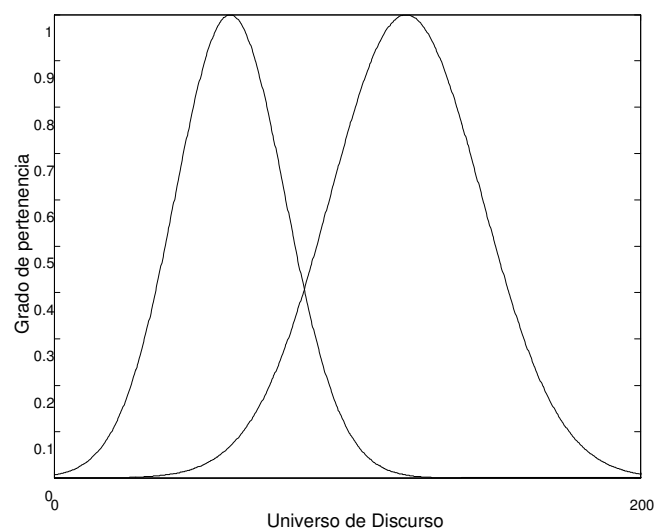


Figura 5.1 Distribución deseable de Funciones de Pertenencia

En la Figura 5.2 se muestra un ejemplo para cada una de las 4 reglas de Marsh en las que no se está cumpliendo una distribución adecuada de las FP.

Para tratar de garantizar el cumplimiento de estas cuatro reglas, será necesario ajustar los parámetros después de aplicar la ley de adaptación de los mismos.

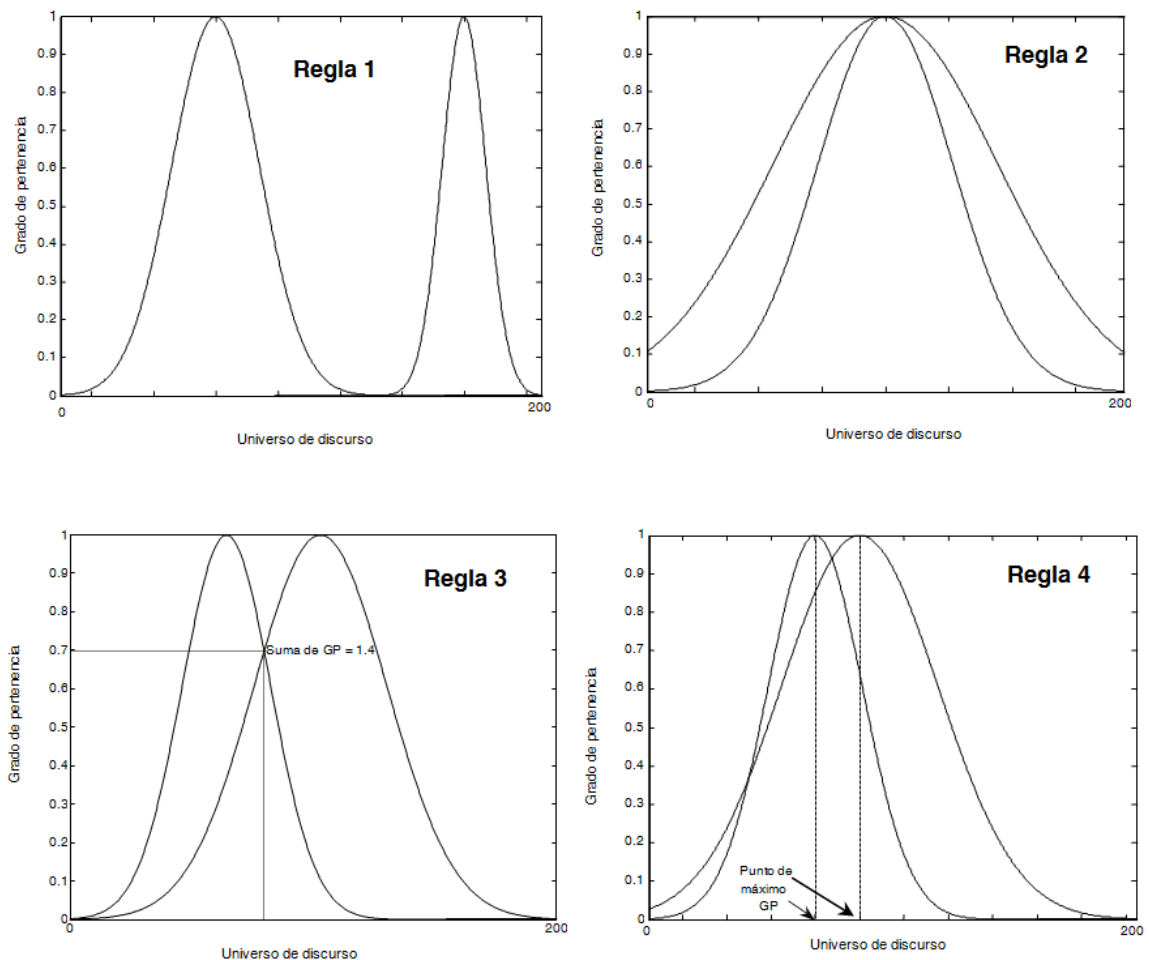


Figura 5.2 Distribución no deseable de Funciones de Pertenencia

Se tendrá que utilizar alguna medida para averiguar el grado de solape que se produce entre las distintas FP tras la adaptación de los parámetros que las caracterizan. Un índice para intentar cumplir los cuatro objetivos mencionados anteriormente puede ser el uso de la Razón de Solape (RS) como medida para saber si dos FP adyacentes están o no demasiado solapadas [129]. La RS entre dos FP adyacentes para un universo de discurso U_i puede ser definida como:

$$RS = \frac{\text{Rango de Solape (RnS)}}{\text{Rango de Definición de Funciones Adyacentes (RfA)}} \quad (5.1)$$

siendo RnS el intervalo del universo de discurso donde dos funciones adyacentes poseen un soporte común y RfA el soporte generado por dos funciones adyacentes.

El estudio se centrará en dos tipos de FP, las FP triangulares que generan triángulos isósceles y/o equiláteros (4.26) y las FP Gaussianas (4.14).

En la Figura 5.3 se puede apreciar la representación de FP triangulares y Gaussianas sobre el mismo universo de discurso donde se destacan tanto RnS como RfA para las FP triangulares.

Si se definen las variables sop_min y sop_max como los límites del soporte inferior y superior de una FP, es decir, los valores para los que la FP se hace cero, entonces la RS para las FP j y $j+1$ adyacentes cumple:

$$RS(j) = \frac{sop_max(j) - sop_min(j+1)}{sop_max(j+1) - sop_min(j)} \quad (5.2)$$

Los soportes para cada FP de tipo triangular están claramente identificados:

$$sop_min(j) = \gamma^j - \frac{\beta^j}{2}; \quad sop_max(j) = \gamma^j + \frac{\beta^j}{2} \quad (5.3)$$

Para el caso de las FP Gaussianas estos soportes dependen del Grado de Pertenencia (GP) que se considere cero, ya que una FP Gaussiana alcanza un GP cero teóricamente en el infinito.

Para poder aplicar (5.2) es necesario definir un soporte, es decir, se tendrá que indicar cuando una FP Gaussiana tiene un GP tan bajo que se considere cero. Para ello se supone que la FP Gaussiana se hace cero para ψ veces el ancho (β^j), o sea, si, por ejemplo, $\psi = 2$ se está considerando que el GP para una FP Gaussiana en el punto $\gamma^j - 2\beta^j$ es cero. Este parámetro ψ se define como **Factor de Ancho** y representa para cuantas veces el ancho, la FP alcanza el valor cero.

Nótese que en el caso de FP triangulares el valor de ψ es $\frac{1}{2}$. Este valor de ψ queda a consideración del diseñador para el caso de FP Gaussianas, aunque es conveniente que sea ≥ 2 para que el GP de la función alcance un valor suficientemente cercano a cero.

Los soportes (5.3) pueden expresarse, por tanto, de la siguiente forma:

$$sop_min(j) = \gamma^j - \psi\beta^j; \quad sop_max(j) = \gamma^j + \psi\beta^j \tag{5.4}$$

De sustituir (5.4) en (5.2) resulta:

$$RS(j) = \frac{\gamma^j - \gamma^{j+1} + \psi(\beta^j + \beta^{j+1})}{\gamma^{j+1} - \gamma^j + \psi(\beta^j + \beta^{j+1})} \tag{5.5}$$

Para las FP triangulares representadas en la Figura 5.3 (nótese que las dos FP tienen un GP en el punto de intersección de 0.5), el valor de la RS es 0.3333 (de aplicar (5.5) con $\psi = \frac{1}{2}$). En el caso de las FP Gaussianas, aplicando (5.5) para $\psi = 2$ y $\psi = 2.5$, la razón de solape resulta $RS=0.7777$ y $RS=0.8181$ respectivamente.

En resumen, se puede concluir que la expresión (5.5) será la fórmula general que nos proporciona la Razón de Solape entre dos funciones de pertenencia adyacentes. Como ya se ha mencionado el Factor de Ancho dependerá del tipo de función de pertenencia. En el caso de funciones triangulares este factor es siempre $\frac{1}{2}$, pero en funciones Gaussianas debe ser un valor ≥ 2 , en función del criterio del diseñador.

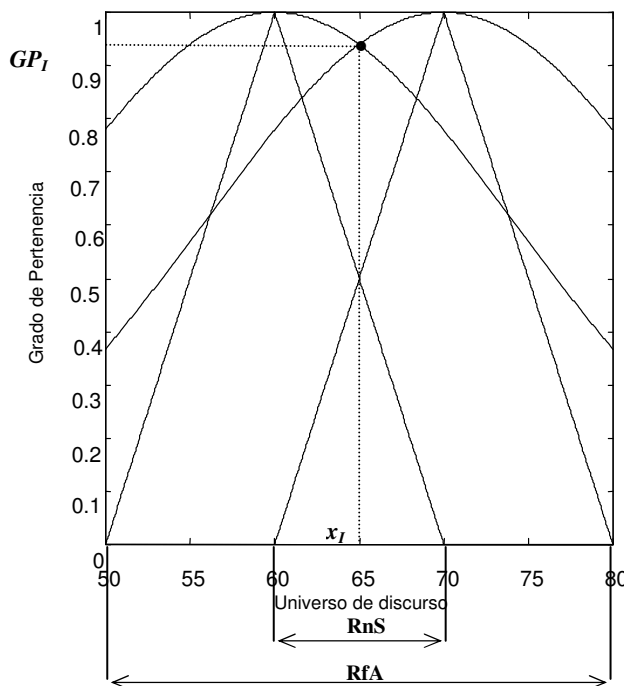


Figura 5.3 Representación de FP triangulares y Gaussianas sobre el mismo universo de discurso

5.2. Obtención del parámetro ψ

Una de las decisiones que se tienen que tomar al usar la RS es proporcionar un valor para el Factor de Ancho ψ en el caso de FP Gaussianas. Para facilitar al diseñador esta elección se ha realizado un estudio de los valores más adecuados para este Factor de Ancho.

El GP para una FP Gaussiana dado un valor ψ se obtiene de aplicar (4.14) siendo $x_i = \text{sop_min}(j)$ o bien $x_i = \text{sop_max}(j)$, como se muestra en (5.3). De ello resulta:

$$GP_{\psi} = \exp\left(-\left(\frac{(\gamma^j - \psi\beta^j) - \gamma^j}{\beta^j}\right)^2\right) \quad (5.6)$$

Por lo que:

$$GP_{\psi} = \exp(-\psi^2) \quad (5.7)$$

Lo que indica que el GP de la función para un valor que se encuentra a ψ veces su ancho sólo depende del valor de este parámetro ψ .

De (5.7) se obtiene la expresión:

$$\psi = \sqrt{-\ln(GP_{\psi})} \quad 0 < GP_{\psi} < 1 \quad (5.8)$$

Así, se obtiene la fórmula que permite al diseñador elegir el valor adecuado de ψ según el GP que considere 0 para una FP Gaussiana.

La Figura 5.4 muestra un ejemplo del significado de las variables definidas previamente. Para un valor $\psi = 2$ el GP_{ψ} es de 0.0183 (1.83% del GP_{\max}) mientras para un valor $\psi = 2.5$ el GP_{ψ} es de 0.0019 (0.19% del GP_{\max}), siendo $GP_{\max} = 1$.

5.3. Ajuste de parámetros basándose en la RS

Con la ayuda de la incorporación de la RS en el proceso de adaptación paramétrica enunciado en el tema 4, se define la siguiente función de coste para ser minimizada:

$$J = \frac{1}{2}(RS - RS^d)^2 \quad (5.9)$$

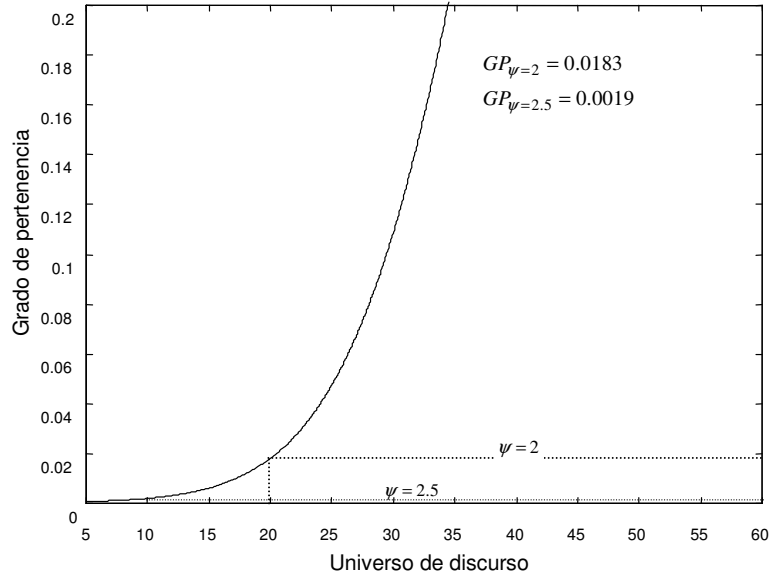


Figura 5.4 Representación del GP_{ψ} con $\psi = 2$ y $\psi = 2.5$ para una FP
Gaussiana de $\gamma = 60$ y $\beta = 60$

donde RS^d es el valor deseado de razón de solape y RS responde a (5.5). La adaptación de cada parámetro resulta de aplicar el algoritmo de gradiente descendente aplicado a (5.5):

$$\xi_i^j(k+1) = \xi_i^j(k) - \eta_{RS} (RS - RS^d) \frac{\partial(RS)}{\partial \xi_i^j(k)} \quad (5.10)$$

siendo ξ_i^j cualquiera de los parámetros γ_i^j, β_i^j . La variable representada por η_{RS} es el factor de aprendizaje para la razón de solape.

El resultado de aplicar (5.10) sobre cada parámetro implicado en el ajuste es:

$$\gamma_i^j(k+1) = \gamma_i^j(k) - \eta_{RS} (RS - RS^d) 2\psi \frac{\beta_i^j + \beta_i^{j+1}}{(\gamma_i^{j+1} - \gamma_i^j + \psi(\beta_i^j + \beta_i^{j+1}))^2} \quad (5.11)$$

$$\beta_i^j(k+1) = \beta_i^j(k) - \eta_{RS} (RS - RS^d) 2\psi \frac{\gamma_i^{j+1} - \gamma_i^j}{(\gamma_i^{j+1} - \gamma_i^j + \psi(\beta_i^j + \beta_i^{j+1}))^2} \quad (5.12)$$

$$\gamma_i^{j+1}(k+1) = \gamma_i^{j+1}(k) + \eta_{RS} (RS - RS^d) 2\psi \frac{\beta_i^j + \beta_i^{j+1}}{(\gamma_i^{j+1} - \gamma_i^j + \psi(\beta_i^j + \beta_i^{j+1}))^2} \quad (5.13)$$

$$\beta_i^{j+1}(k+1) = \beta_i^{j+1}(k) - \eta_{RS} (RS - RS^d) 2\psi \frac{\gamma_i^{j+1} - \gamma_i^j}{(\gamma_i^{j+1} - \gamma_i^j + \psi(\beta_i^j + \beta_i^{j+1}))^2} \quad (5.14)$$

5.4. Elección del parámetro RS^d

Además de ψ otro parámetro crítico para la adaptación con restricciones es la elección de la RS^d . Con el presente estudio se intenta ayudar al diseñador en la elección de este parámetro, principalmente para el caso de FP Gaussianas, ya que como se mencionó anteriormente, para las FP triangulares el valor de la RS^d debe ser de $\frac{1}{3}$.

En un universo de discurso normalizado la distribución adecuada de las FP sería que los centros de las mismas estuviesen uniformemente distribuidos entre -1 y 1 y los anchos fueran tales que la RS entre cada dos FP adyacentes fuese la RS deseada por el diseñador, es decir, RS^d .

Utilizando este criterio, la distribución de centros adecuada para M reglas puede ser expresada como:

$$\text{Centro}(l) = \frac{2(l-1)}{M-1} - 1 ; \quad l=1..M \quad (5.15)$$

Así se garantiza que el centro para la primera FP se encuentra situado en el punto -1 y el centro para la última FP en el punto 1 . El resto de los centros están distribuidos uniformemente en el intervalo mencionado.

El ancho adecuado para cada una de las FP se consigue en función de la RS^d y del parámetro ψ . Este ancho se obtiene de la expresión (5.5) para dos funciones adyacentes:

$$RS^d = \frac{\text{Centro}(l) - \text{Centro}(l+1) + \psi(\text{Ancho}(l) + \text{Ancho}(l+1))}{\text{Centro}(l+1) - \text{Centro}(l) + \psi(\text{Ancho}(l) + \text{Ancho}(l+1))}$$

Si los centros tienen la forma (5.15) y $\text{Ancho}(l) = \text{Ancho}(l+1)$ entonces:

$$RS^d = \frac{\left(-\frac{2}{M-1} + \psi(2\text{Ancho}(l)) \right)}{\left(\frac{2}{M-1} + \psi(2\text{Ancho}(l)) \right)}$$

De donde se puede obtener la expresión de $\text{Ancho}(l)$, por lo que la distribución de los anchos debería estar basada en la siguiente condición:

$$\text{Ancho}(l) = \frac{(RS^d + 1)}{\psi(1 - RS^d)(M-1)} ; \quad l=1..M \quad (5.16)$$

El punto en el que dos FP adyacentes se interceptan (x_l , ver Figura 5.3) se obtiene de igualar dos FP Gaussianas adyacentes en las que los centros y anchos serán los correspondientes a aplicar (5.15) y (5.16) respectivamente. Considerando lo anterior, los puntos de intersección en el universo de discurso pueden ser calculados como:

$$x_l(l) = \frac{2l - M}{M - 1} ; \quad l = 1..M \quad (5.17)$$

El Grado de Pertenencia de la función en el punto de intersección, GP_l , se calcula sustituyendo (5.15), (5.16) y (5.17) en (4.14), esto es:

$$GP_l = \exp \left(- \left(\frac{\psi(1 - RS^d)}{RS^d + 1} \right)^2 \right) = \exp \left(- \left(\frac{\sqrt{-\ln(GP_\psi)}(1 - RS^d)}{RS^d + 1} \right)^2 \right) \quad (5.18)$$

Si se establece una RS^d se tendrá para cada posible valor de GP_ψ un valor de GP_l . En la Figura 5.5 se muestra la evolución del GP_l en función del GP_ψ para distintas RS^d , lo que facilita la elección de la RS^d una vez decidido el GP_ψ y el GP_l . Por ejemplo, para un $GP_\psi = 0.013$ y un $GP_l = 0.45$, la RS^d será de 0.4, tal y como se muestra en la Figura 5.5.

5.5. Ajuste paramétrico de los consecuentes

El proceso de ajuste de centros y anchos de las FP en la parte precedente de las reglas ha sido implementado. Se debe realizar un proceso de ajuste de los consecuentes para asegurar una distribución uniforme de los mismos en un universo de discurso normalizado dentro del intervalo [-1,1].

De forma similar a la aplicada a las FP de la parte precedente del conjunto de reglas, el ajuste basado en la RS puede extenderse a las FP de la parte consecuente. En el caso de que el consecuente sea un escalar (singleton) o del tipo Takagi-Sugeno-Kang (TSK) no es posible aplicar la RS ya que el concepto de solape no tiene sentido. Para usar la RS es necesario transformar los consecuentes en FP equivalentes. Se generan con este propósito FP triangulares isósceles de la forma (4.26), cuyos centros se corresponden con los valores de los consecuentes y cuyos anchos son los apropiados para obtener la RS deseada por el diseñador.

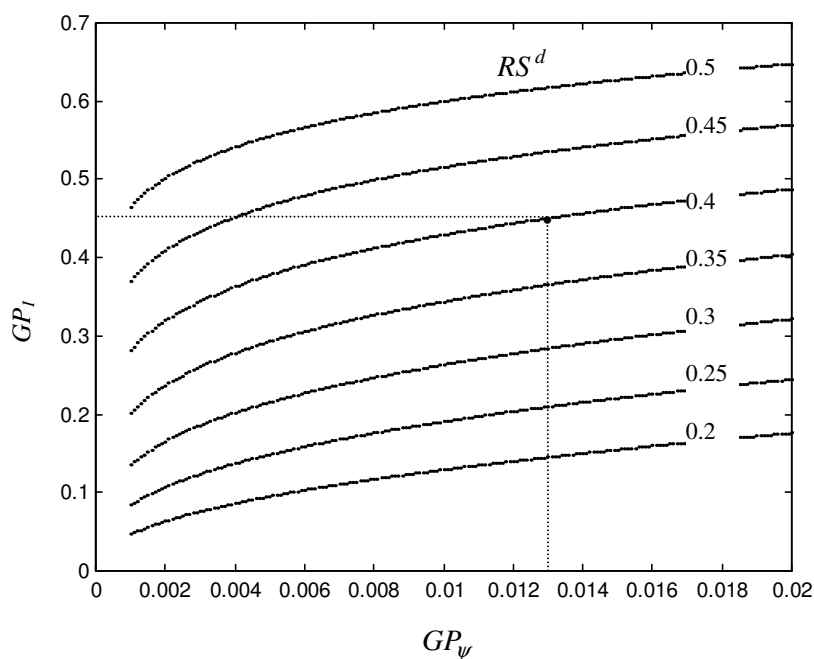


Figura 5.5 Gráfica para determinar la RS^d para cada dos FP adyacentes dado un GP_ψ y un GP_l

La expresión (5.15), donde cada centro es equivalente a cada consecuente de la regla, se puede aplicar para distribuir los consecuentes en el universo de discurso normalizado. Los anchos de las FP triangulares equivalentes se obtienen de (5.16), donde $\psi = \frac{1}{2}$, ya que para este valor las FP triangulares se hacen cero, y donde $RS^d = \frac{1}{3}$ para que el GP en el punto de intersección de dos campanas adyacentes sea 0.5.

Se obtiene así, considerando el criterio previamente expuesto, la distribución adecuada de los consecuentes para las M reglas:

$$\kappa(l) = \frac{2(l-1)}{M-1} - 1; \quad l = 1..M \quad (5.19)$$

Y además los anchos de las FP triangulares equivalentes centradas en los consecuentes anteriores:

$$\chi(l) = \frac{4}{M-1}; \quad l = 1..M \quad (5.20)$$

En el caso de FP triangulares, los soportes están claramente identificados y responden a:

$$sop_min(j) = \kappa^j - \frac{\chi^j}{2}; \quad sop_max(j) = \kappa^j + \frac{\chi^j}{2} \quad (5.21)$$

Por lo que, sustituyendo (5.21) en (5.2) se obtiene la expresión para la RS de los consecuentes (RSC):

$$RSC(j) = \frac{\kappa^j - \kappa^{j+1} + \frac{1}{2}(\chi^j + \chi^{j+1})}{\kappa^{j+1} - \kappa^j + \frac{1}{2}(\chi^j + \chi^{j+1})} \quad (5.22)$$

En la Figura 5.6 se representa la distribución de los consecuentes en un universo de discurso normalizado para 5 reglas. También se muestran sus 5 FP triangulares equivalentes centradas en cada uno de los consecuentes, todas con el mismo ancho $\chi(i) = 1$.

El ajuste paramétrico se realizará a partir de la siguiente función de coste:

$$J = \frac{1}{2}(RSC - RSC^d)^2 = \frac{1}{2}(RSC - \frac{1}{3})^2 \quad (5.23)$$

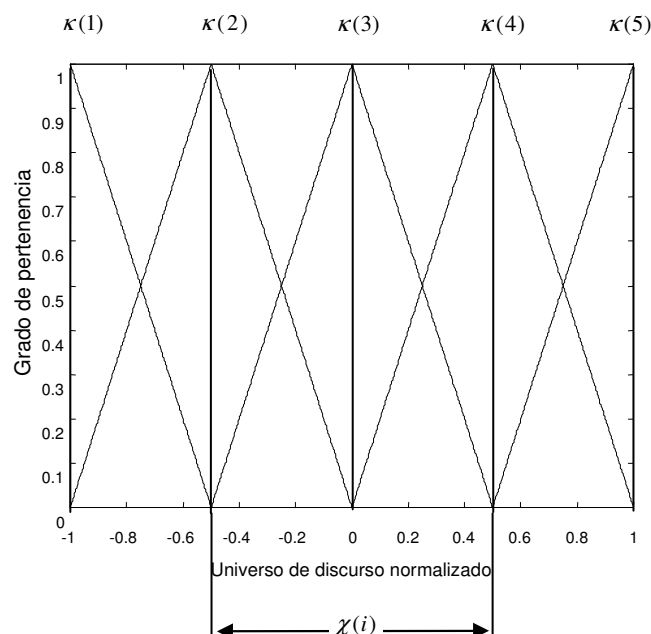


Figura 5.6 Representación de los consecuentes y sus FP triangulares equivalentes para $M=5$

Y la adaptación de cada parámetro resulta:

$$\xi_i^j(k+1) = \xi_i^j(k) - \eta_{RSC} \left(RSC - \frac{1}{3} \right) \frac{\partial(RSC)}{\partial \xi_i^j(k)} \quad (5.24)$$

donde ξ_i^j representa los parámetros de las funciones de pertenencia triangulares que van a ser ajustados (κ_i^j, χ_i^j) y η_{RSC} es el factor de aprendizaje para la RS aplicado al consecuente.

En el caso de que el consecuente sea un singleton, el resultado de aplicar (5.24) sobre cada parámetro implicado en el ajuste es:

$$\kappa_i^j(k+1) = \kappa_i^j(j) - \eta_{RSC} \left(RSC - \frac{1}{3} \right) \frac{\chi_i^j + \chi_i^{j+1}}{(\kappa_i^{j+1} - \kappa_i^j + \frac{1}{2}(\chi_i^j + \chi_i^{j+1}))^2} \quad (5.25)$$

$$\chi_i^j(k+1) = \chi_i^j(k) - \eta_{RSC} \left(RSC - \frac{1}{3} \right) \frac{\kappa_i^{j+1} - \kappa_i^j}{(\kappa_i^{j+1} - \kappa_i^j + \frac{1}{2}(\chi_i^j + \chi_i^{j+1}))^2} \quad (5.26)$$

De forma similar, para un consecuente del tipo TSK $(\kappa = a_0 + a_1x_1 + \dots + a_lx_l + \dots + a_nx_n)$, el resultado de aplicar (5.24) sobre cada parámetro involucrado en el ajuste es:

$$a_{0i}^j(k+1) = a_{0i}^j(k) - \eta_{RSC} \left(RSC - \frac{1}{3} \right) \frac{\chi_i^j + \chi_i^{j+1}}{(\kappa_i^{j+1} - \kappa_i^j + \frac{1}{2}(\chi_i^j + \chi_i^{j+1}))^2} \quad (5.27)$$

$$a_{1i}^j(k+1) = a_{1i}^j(k) - \eta_{RSC} \left(RSC - \frac{1}{3} \right) \frac{(\chi_i^j + \chi_i^{j+1})x_1^j}{(\kappa_i^{j+1} - \kappa_i^j + \frac{1}{2}(\chi_i^j + \chi_i^{j+1}))^2} \quad (5.28)$$

$$\chi_i^j(k+1) = \chi_i^j(k) - \eta_{RSC} \left(RSC - \frac{1}{3} \right) \frac{\kappa_i^{j+1} - \kappa_i^j}{(\kappa_i^{j+1} - \kappa_i^j + \frac{1}{2}(\chi_i^j + \chi_i^{j+1}))^2} \quad (5.29)$$

5.6. Algoritmo de adaptación con ajuste basado en la RS

Se ha implementado un algoritmo que obtiene un modelo borroso a partir de datos de entrada-salida, adaptando los parámetros de las FP y ajustándolos para que las mismas mantengan un adecuado solape dentro del universo de discurso. Los consecuentes también son ajustados basándose en los criterios expuestos en el apartado 5.5. Los pasos para implementar el algoritmo se muestran en la Figura 5.7:

-
- 1: Se introducen los parámetros de entrada;
 - 2: Normalización de los valores de entrada-salida;
 - 3: Se inicializan centros y anchos con FP uniformemente distribuidas;
 - 4: Se inicializan consecuentes;
 - 5: **Mientras** $(e(y, \hat{y}) > E_{max})$ **hacer**
 - 6: Se estima la función;
 - 7: Se adaptan los parámetros;
 - 8: Se ajustan centros y anchos basándose en la RS;
 - 9: Se ajustan los consecuentes basándose en la RS;
 - 10: Se estima de nuevo la función con los parámetros adaptados y ajustados;
 - 11: Se calcula el error;
 - 12: **Fin Mientras**
-

Figura 5.7 Algoritmo de adaptación con ajuste de antecedentes y consecuentes basándose en la RS

donde E_{max} representa el error máximo permitido en la estimación del modelo y donde $e(y, \hat{y})$ es la suma de cuadrados del error entre la función real y la función estimada.

Se comentará a continuación más detalladamente cada uno de estos pasos, profundizando en cada uno de los aspectos relevantes de cada punto e incorporando aportaciones al mismo.

Parámetros de entrada y normalización

Inicialmente necesitamos introducir una serie de parámetros de entrada para que el algoritmo pueda ejecutarse. Estos parámetros son los siguientes:

1. *Número de épocas*: Número de veces que se pasa el patrón de entrenamiento para adaptar los parámetros si no encuentra previamente un estimación adecuada de la función.
2. *Número de reglas (M)*: Cantidad de reglas de la forma (4.2) con las que se va a tratar de estimar la función. El algoritmo genera una FP para cada regla de cada una de las variables de entrada-salida.
3. *Factor de aprendizaje en la adaptación (η)*: Valor entre 0 y 1. Con este valor se puede regular la magnitud de variación con que se adaptan los parámetros implicados en el ajuste. Si este valor es pequeño la adaptación se realizará más lentamente pero con mayor precisión y estabilidad en el modelo resultante, sin embargo en ocasiones puede ser interesante proporcionar valores altos para llegar más rápidamente a la estimación de la función.
4. *Error máximo (E_{max})*: Será el valor que nos permitirá comparar con el error entre la función real y la función estimada ($e(y, \hat{y})$), utilizado aquí como criterio de parada.
5. *Factor de Aprendizaje para la Razón de Solape (μ_{RS})*: Valor entre 0 y 1. Con este valor se puede regular la velocidad a la que se ajustan los centros y los anchos basándose en la RS. Si este valor es pequeño se prima más a la estimación de la función que al ajuste de las FP, por lo que la distribución final de las FP puede no ser uniforme. Un valor más alto consigue ajustar mejor los centros y los anchos y consigue una mejor distribución de las FP aunque a costa de estimar peor la función.
6. *Factor de ancho (ψ)*: Como se ha mencionado anteriormente, en la definición de la RS es necesario calcular los soportes máximo y mínimo (5.4). Para funciones triangulares estos valores están claramente identificados, ya que para 0.5 veces el ancho (β^j) el GP de la función se hace 0, por lo que el valor de ψ es, claramente, 0.5. En el caso de funciones Gaussianas este valor depende de cuando se considere que se hace cero el GP de la función, por lo que será una variable a considerar para el resto del proceso de estimación. Como se explicó en la Figura 5.4, valores inferiores a 2 para ψ generan GP en las

funciones Gaussianas que podemos considerar cero, por lo que se recomienda el uso de $2 \leq \psi \leq 2.5$.

7. *Razón de Solape Deseada (RS^d)*: Una de las decisiones más importantes será la elección de la RS^d . Según Marsh [129], cuando se solapan dos FP, la suma de los grados de pertenencia para cualquier punto en el solape debe ser menor o igual a 1, por lo que el GP en la intersección de dos FP debe ser menor o igual a 0.5. Si previamente se ha decidido el valor adecuado de ψ , se puede consultar la Figura 5.5 para determinar el valor de la RS^d más correcto. Según esta gráfica, en el caso de funciones de pertenencia Gaussianas y dependiendo de los valores que se estimen adecuados para GP_I y GP_ψ , la RS^d debería estar entre 0.4 y 0.5. Para las FP triangulares se considerará $RS^d = \frac{1}{3}$.
8. *Factor de Aprendizaje para la Razón de Solape de los consecuentes (η_{RSC})*: Valor entre 0 y 1. Con este valor se puede regular la velocidad a la que se ajustan los consecuentes basándose en la RS. Si este valor es pequeño se prima más a la estimación de la función que al ajuste de los consecuentes, lo que puede hacer que la distribución final de los mismos no sea uniforme. Un valor más alto consigue ajustar mejor la distribución de los consecuentes pero la estimación de la función podría ser peor.

Posteriormente se normalizan todos los valores de entrada-salida, para que se encuentren situados entre -1 y 1 . Los resultados finales del algoritmo también se encuentran normalizados, por lo que habrá que desnormalizarlos.

Inicialización de centros y anchos con FP uniformemente distribuidas

El paso siguiente consiste en inicializar los distintos parámetros que definen las FP triangulares (4.26) o Gaussianas (4.14), es decir, centros (γ_i^l) y anchos (β_i^l) e inicializar los consecuentes (y^l).

1. *Inicialización de centros*: Se distribuyen los centros en el universo de discurso de forma uniforme. Para ello se tiene en cuenta el número de reglas (M). El primer centro se sitúa en la posición -1 (ya que los datos están normalizados

entre -1 y 1) y los siguientes a una distancia de $2/(M-1)$, tal y como se expresa en (5.15).

2. *Inicialización de anchos*: Se inicializan todos los anchos con el mismo valor, buscando que al iniciarse el algoritmo las funciones tengan un solape adecuado. Se pretende que las FP iniciales tengan todas el mismo ancho y que la RS entre cada dos FP adyacentes sea la RS^d . Para ello se utiliza la expresión (5.16) que garantiza una RS inicial que cumple con la RS^d .

Inicialización de consecuentes

Como se ha explicado en el apartado 5.5 se transforman los consecuentes en FP triangulares isósceles equivalentes, cuyos centros se corresponden con los valores de los consecuentes y cuyos anchos son los apropiados para obtener la RS deseada por el diseñador. La distribución adecuada de los consecuentes para las M reglas se corresponde con la expresión (5.19), situando el primer consecuente en -1 y los restantes a una distancia de $2/(M-1)$. Los anchos de las FP triangulares equivalentes centradas en los consecuentes anteriores se calculan según la expresión (5.20), lo que garantiza un adecuado solape de las mismas al iniciarse el algoritmo.

Estimación de la función

Se realiza la estimación de la función atendiendo a la formulación (4.3) de donde se obtiene la función estimada \hat{y} .

Adaptación de los parámetros

Para adaptar los centros tendremos que aplicar (4.20) o bien (4.32), dependiendo de si las FP son Gaussianas o triangulares respectivamente. La adaptación de los anchos se regirá por (4.25) en el caso de FP Gaussianas y por (4.37) para FP triangulares. En el caso de consecuentes singleton, estos se adaptarán siguiendo la expresión (4.43), sin embargo, en el caso de consecuentes del tipo TSK la adaptación se realizará basándose en las expresiones (4.53) para el término afín y (4.54).

Ajuste de centros y anchos basándose en la RS

En cada iteración se consulta la RS entre cada dos FP adyacentes. Si la RS es superior a la RS^d se realiza el ajuste de los centros y anchos de cada una de las FP que producen un solape superior al deseado, atendiendo a las expresiones (5.11), (5.12), (5.13) y (5.14).

Ajuste de los consecuentes basándose en la RS

En el caso de que el solape entre dos FP triangulares adyacentes equivalentes sea superior al solape deseado ($RSC^d = \frac{1}{3}$) se realizará el correspondiente ajuste, que responderá a (5.25) y (5.26) si los consecuentes son escalares (singleton) o responderá a (5.27), (5.28) y (5.29) si son del tipo TSK.

Cálculo del error

El error que se calcula es la suma de cuadrados del error entre la función real y la función estimada. Este error puede ser sustituido por cualquier otro que se estime más oportuno (error cuadrático medio, error porcentual medio cuadrático, error medio, error medio absoluto, etc.).

5.7. Conclusiones

Para poder incorporar al proceso de adaptación de parámetros un procedimiento de ajuste de los mismos con el fin de buscar una mejor distribución de las funciones de pertenencia y con ello ganar en interpretabilidad del modelo, se propone un nuevo índice que nos permita evaluar el solape entre dos funciones de pertenencia adyacentes. Este índice, denominado *Razón de Solape*, fue propuesto por Marsh [129] y posteriormente desarrollado en [170], aunque únicamente se trataban funciones de pertenencia triangulares.

En esta Tesis se utiliza la Razón de Solape tanto en funciones triangulares como en funciones gaussianas. En el caso de funciones triangulares la obtención de los soportes no presenta ningún problema, sin embargo, en el caso de funciones de pertenencia gaussianas

este soporte depende del grado de pertenencia que se considere cero, ya que una función gaussiana alcanza un grado de pertenencia cero teóricamente en el infinito.

Este problema nos lleva a la definición de un nuevo parámetro que se ha denominado *Factor de Ancho* (ψ) y representa para cuántas veces el ancho la función de pertenencia alcanza el valor cero.

Para el proceso de ajuste se tiene que proporcionar un nuevo parámetro, la *Razón de Solape Deseada*. Este dato debe ser proporcionado por el diseñador y significa el valor que se considera más adecuado para la Razón de Solape entre dos funciones de pertenencia adyacentes.

Para ayudar al diseñador en la búsqueda del mejor valor para la Razón de Solape Deseada, se aporta en este trabajo una gráfica que dado un Grado de Pertenencia en ψ (GP_ψ) y un grado de pertenencia en la intersección de las dos funciones (GP_I) nos determina el valor de la Razón de Solape Deseada.

El ajuste se incorpora no sólo a la parte precedente de las reglas sino también a los consecuentes, con un procedimiento novedoso que permite utilizar la Razón de Solape aunque los consecuentes sean singleton o de tipo TSK.

El capítulo concluye con el desarrollo del algoritmo que permite la obtención del modelo borroso, realizando la adaptación de los parámetros mediante el mecanismo de adaptación desarrollado por Wang [202] y adaptando en cada paso los parámetros, tanto de la parte precedente como de los consecuentes, con la ayuda de la Razón de Solape. Se detallan los principales parámetros de entrada y la inicialización que se lleva a cabo antes de comenzar a iterar, de forma que inicialmente se tenga una distribución adecuada de las funciones, que se tratará de mantener durante el mecanismo de adaptación para lograr una mayor interpretabilidad.

La principal referencia de este capítulo es [129], donde Marsh establece las cuatro reglas que deben cumplir las funciones de pertenencia para que estén correctamente distribuidas en el universo de discurso y define la Razón de Solape como índice para medir el grado de solape entre dos funciones adyacentes.

6. El Área de Solape

6.1. Un nuevo índice en el ajuste paramétrico de FP

Hasta ahora se ha evaluado el solape entre dos funciones adyacentes de forma aproximada, ya que se ha considerado sólo la distancia entre el máximo soporte de la función j y el mínimo soporte de la función vecina $j+1$. Sin embargo sería ideal encontrar un índice que nos pueda proporcionar mayor información sobre el solape real existente entre dos funciones adyacentes.

Marsh [129] fue uno de los pioneros en estudiar este problema y proponer la Robustez de Solape (RoS) como índice para medir el solape (basado en el área) entre dos funciones adyacentes:

$$RoS = \frac{\int_{L_{j+1}}^{U_j} \mu_{F_j} + \int_{L_{j+1}}^{U_j} \mu_{F_{j+1}}}{2(U_j - L_{j+1})} \quad (6.1)$$

donde L_j y U_j representan el soporte mínimo y máximo, respectivamente, de la función j . Sin embargo, como se muestra en la Figura 2.1 (La RoS se corresponde con la superficie sombreada) este índice no proporciona información sobre el solape real entre dos funciones adyacentes j y $j+1$.

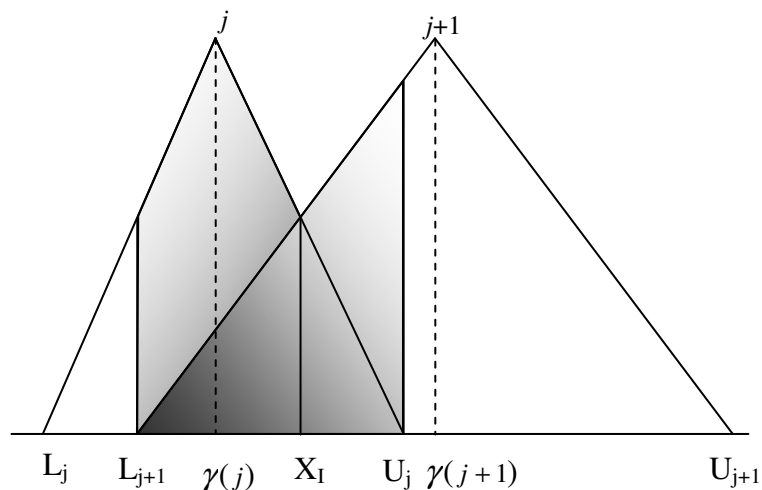


Figura 6.1 Representación de la Robustez de Solape para dos FP triangulares adyacentes

En realidad lo que se pretende conseguir con este índice es medir la superficie de solapamiento que se produce entre dos funciones adyacentes, por lo que se ha optado por modificar la expresión para que reflejara con más exactitud el verdadero área de solape de las dos funciones.

Teniendo en cuenta lo anterior, en este trabajo se propone un nuevo índice para medir el solape entre dos funciones adyacentes. A este nuevo índice se la denomina Área de Solape (AS) (ver Figura 6.2) y calcula una medida real del área de la intersección de dos funciones adyacentes, ya que no tiene en cuenta el área que se encuentra fuera del área de solape.

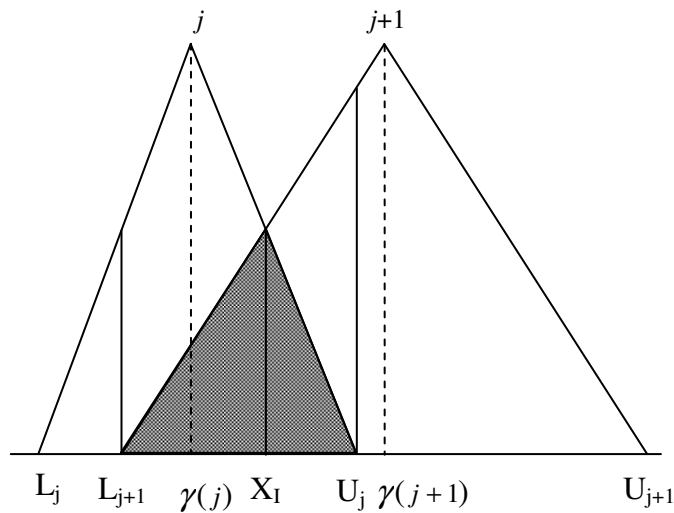


Figura 6.2 Representación del Área de Solape para dos FP triangulares adyacentes

En el caso de FP Gaussianas, la intersección de la FP con el eje X no está claramente definido, como se explicó en el capítulo anterior, debido a que estas funciones alcanzan el GP cero teóricamente en el infinito. En la Figura 6.3 se puede apreciar como la superficie de solape será diferente para distintos valores de ψ . Los soportes mínimos y máximos en el caso de FP Gaussianas serían L_1 y U_1 para $\psi = 2$. Igualmente, si $\psi = 2.5$ entonces los soportes serían L_2 y U_2 .

El nuevo índice denominado AS puede ser expresado como:

$$AS = \frac{\text{Área de intersección de dos funciones Adyacentes (AifA)}}{\text{Área Menor (AM)}} \tag{6.2}$$

donde A_{ifA} representa el área en el que dos funciones adyacentes se solapan y AM la menor de las áreas de las dos funciones.

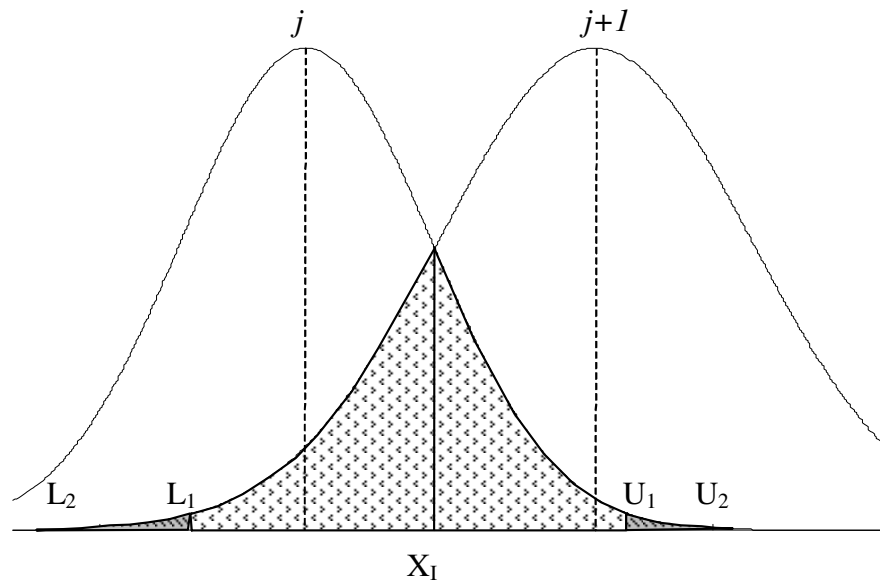


Figura 6.3 Representación del Área de Solape para dos FP Gaussianas adyacentes

Para que el AS sea un índice que nos mida la proporción de solape existente entre dos funciones, se ha tomado como denominador el área menor de las dos funciones, ya que este valor será el máximo área de solape que se podría dar cuando una función se encuentre totalmente incluida dentro de la otra. Cuando esto ocurre (Figura 6.4) el área de intersección coincide con la menor de las áreas, por lo que AS será 1.

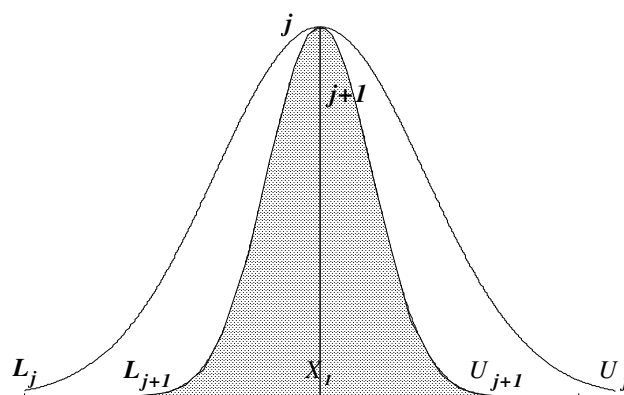


Figura 6.4 Representación de dos FP Gaussianas adyacentes con $AS=1$

El AS dependerá por lo tanto de la función cuyo área sea menor. Este factor sólo depende del ancho de la función (β^j), por lo tanto, el denominador será el área de la FP j si $\beta^j \leq \beta^{j+1}$, en caso contrario el denominador será el área de la FP $j+1$.

Otro factor a tener en cuenta en el cálculo del AS es la posición en el universo de discurso de la FP con menor área. La posición en el universo de discurso viene determinado por el centro de la función (γ^j), de forma que, si $\gamma^j \leq \gamma^{j+1}$ se considera que la FP j se encuentra más a la izquierda en el universo de discurso.

Estas consideraciones anteriores proporcionan cuatro posibles casos cuando se calcula el AS:

1. La FP j tiene un área menor (por lo tanto, $\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).
2. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).
3. La FP j tiene un área menor ($\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).
4. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).

Así, para el primer caso, el AS para dos FP adyacentes se representa como:

$$AS(j) = \frac{\int_{L_{j+1}}^{X_l} \mu_{F_{j+1}} + \int_{X_l}^{U_j} \mu_{F_j}}{\int_{L_j}^{U_j} \mu_{F_j}} \quad (6.3)$$

Para el resto de los casos, ver Apéndice I.

Para FP triangulares, la expresión (6.3) para el AS resulta:

$$AS(j) = \frac{\int_{L_{j+1}}^{X_I} \left(1 - \frac{2|x - \gamma^{j+1}|}{\beta^{j+1}}\right) + \int_{X_I}^{U_j} \left(1 - \frac{2|x - \gamma^j|}{\beta^j}\right)}{\int_{L_j}^{U_j} \left(1 - \frac{2|x - \gamma^j|}{\beta^j}\right)} \quad (6.4)$$

donde, de acuerdo con (5.4), $U_j = \gamma^j + \psi\beta^j$ y $L_j = \gamma^j - \psi\beta^j$ y como se ha comentado anteriormente para FP triangulares $\psi = \frac{1}{2}$.

Para FP Gaussianas, la expresión (6.3) para el AS resulta:

$$AS(j) = \frac{\int_{L_{j+1}}^{X_I} \exp\left[-\left(\frac{x - \gamma^{j+1}}{\beta^{j+1}}\right)^2\right] + \int_{X_I}^{U_j} \exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right]}{\int_{L_j}^{U_j} \exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right]} \quad (6.5)$$

donde, de acuerdo con (5.4), $U_j = \gamma^j + \psi\beta^j$ y $L_j = \gamma^j - \psi\beta^j$.

Para FP Gaussianas ψ es un valor definido por el diseñador y debe encontrarse dentro del intervalo [2,2.5].

Para averiguar el punto en el que dos funciones adyacentes se intersectan igualamos las expresiones de las mismas:

$$1 - \frac{2|x - \gamma^{j+1}|}{\beta^{j+1}} = 1 - \frac{2|x - \gamma^j|}{\beta^j} \quad \text{en el caso de FP triangulares,}$$

$$\exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right] = \exp\left[-\left(\frac{x - \gamma^{j+1}}{\beta^{j+1}}\right)^2\right] \quad \text{para FP Gaussianas.}$$

En ambos casos, desarrollando, obtenemos los puntos de intersección de las funciones:

$$x = X_I = \frac{\beta^{j+1}\gamma^j - \beta^j\gamma^{j+1}}{\beta^{j+1} - \beta^j} \quad (6.6)$$

$$x = X_I = \frac{\beta^j\gamma^{j+1} + \beta^{j+1}\gamma^j}{\beta^j + \beta^{j+1}} \quad (6.7)$$

De los dos posibles puntos de intersección, el que nos interesa es el expresado en (6.7)

Para el resto de los casos ver Apéndice II.

6.2. Ajuste de parámetros basándose en el AS

La función de coste que utiliza el nuevo índice en el proceso de adaptación paramétrica enunciado en el Capítulo 4, se define mediante la expresión:

$$J = \frac{1}{2} (AS - AS^d)^2 \quad (6.8)$$

donde AS^d es el valor deseado de Área de Solape.

La adaptación de cada parámetro resulta:

$$\xi_i^j(k+1) = \xi_i^j(k) - \eta_{AS} (AS - AS^d) \frac{\partial (AS)}{\partial \xi_i^j(k)} \quad (6.9)$$

siendo $\xi_{i_n}^j$ cualquiera de los parámetros γ_i^j, β_i^j . η_{AS} es el factor de aprendizaje para el área de solape y AS^d el área de solape que deseamos para las FP.

Para FP Gaussianas, la ley de ajuste, resultado de aplicar (6.9) a cada parámetro implicado, y asumiendo que el área de la función j es menor que el de la función $j+1$ y que la función j se encuentra a la izquierda de la función $j+1$ en el universo de discurso, resulta en:

$$\gamma_i^j(k+1) = \gamma_i^j(k) - \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_j} \quad (6.10)$$

$$\gamma_i^{j+1}(k+1) = \gamma_i^{j+1}(k) + \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_j} \quad (6.11)$$

$$\beta_i^j(k+1) = \beta_i^j(k) + \eta_{AS} (AS - AS^d) \left(\frac{\beta_{j+1} \text{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_j^2 \text{erf}(\psi)} + \frac{\beta_{j+1}}{2\beta_j^2} + \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \text{erf}(\psi) \sqrt{\pi} \beta_j} \right) \quad (6.12)$$

$$\beta_i^{j+1}(k+1) = \beta_i^{j+1}(k) - \eta_{AS} (AS - AS^d) \left(\frac{\operatorname{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_j \operatorname{erf}(\psi)} + \frac{1}{2\beta_j} - \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \operatorname{erf}(\psi) \sqrt{\pi} \beta_j} \right) \quad (6.13)$$

donde $\operatorname{erf}(x)$ es un número cuyo valor se calcula como:

$$\operatorname{erf}(x) = 2 \frac{\int_0^x \exp(-t^2) dt}{\sqrt{\pi}} \quad (6.14)$$

Para el resto de los casos, ver Apéndice III.

6.3. Elección del parámetro AS^d

Para averiguar cuál sería el valor más adecuado para el AS^d se estudian los valores de AS para las funciones iniciales que se encuentran uniformemente distribuidas en el universo de discurso atendiendo a los centros (5.15) y anchos (5.16) que se consideran óptimos dada una RS^d y un factor de ancho ψ . Los valores del AS^d para valores óptimos de RS^d y ψ se encuentran entre 0.2 y 0.3, por lo que el diseñador tendrá que seleccionar el valor deseado de AS^d dentro de estos márgenes.

6.4. Algoritmo de adaptación con ajuste basado en el AS

El algoritmo presentado en la sección 5.6 se ha modificado para insertar en la adaptación paramétrica de la parte precedente el nuevo índice (AS) que nos mide el área de solape entre dos funciones adyacentes, manteniendo la RS como índice para ajustar la parte precedente. Este nuevo algoritmo se presenta en la Figura 6.5.

-
- 1: **Se introducen los parámetros de entrada;**
 - 2: Normalización de los valores de entrada-salida;
 - 3: Se inicializan centros y anchos con FP uniformemente distribuidas;
 - 4: Se inicializan consecuentes;
 - 5: **Mientras** $(e(y, \hat{y}) > E_{max})$ **hacer**
 - 6: Se estima la función;
 - 7: Se adaptan los parámetros;
 - 8: **Se ajustan centros y anchos basándose en el AS;**
 - 9: Se ajustan los consecuentes basándose en la RS;
 - 10: Se estima de nuevo la función con los parámetros adaptados y ajustados;
 - 11: Se calcula el error;
 - 12: Fin Mientras
-

Figura 6.5 Algoritmo de adaptación con ajuste de antecedentes y consecuentes basándose en la RS

donde E_{max} representa el error máximo permitido en la estimación del modelo y donde $e(y, \hat{y})$ es la suma de cuadrados del error entre la función real y la función estimada.

Las dos líneas que sufren alguna modificación son la 1 y la 8. Estas modificaciones se comentan a continuación:

Parámetros de entrada

Dos nuevas variables intervienen en el algoritmo si se utiliza el AS como índice para medir el solapamiento entre dos FP:

1. *Factor de Aprendizaje para el Área de Solape* (η_{AS}): Valor entre 0 y 1. Su función será controlar el ajuste de los centros y anchos de las FP. Valores más altos producirán ajustes más bruscos de los parámetros implicados en el ajuste, garantizando la correcta distribución de las FP, pero provocando una peor estimación de la función.
2. *Área de Solape Deseada* (AS^d): El valor que se propone para el AS^d se obtiene de calcular el AS para FP con los centros (5.15) y anchos (5.16) propuestos como óptimos cuando se usa la RS. Este valor, por lo tanto,

depende de dos parámetros (del valor elegido para ψ , y del valor elegido para la RS^d). Tras analizar que valores son los más adecuados, se proponen valores tales que $0.2 \leq AS^d \leq 0.3$. Un valor que produce buenos resultados es 0.25.

Ajuste de centros y anchos basándose en el AS

En cada iteración, y para cada dos FP adyacentes el algoritmo estudia que caso de los 4 posibles se está dando. Realiza el cálculo del AS correspondiente a su caso (ver Apéndice II) y si es superior al AS^d realiza el ajuste de los centros y anchos implicados, atendiendo a (6.10), (6.11), (6.12) y (6.13) si nos encontramos en el primer caso (para el resto de los casos ver Apéndice III).

6.5. Conclusiones

Con el propósito de buscar un índice que midiera con mayor exactitud el verdadero solape entre dos funciones de pertenencia adyacentes, Marsh propone la Robustez de Solape [129]. Lo que se pretende con este índice es medir la superficie de solapamiento que se produce entre dos funciones, por lo que se ha optado por modificar el índice propuesto por Marsh y aportar un nuevo índice que mida la superficie de solapamiento real entre dos funciones, el *Área de Solape*.

El cálculo del Área de Solape se vuelve más complejo cuando se trata de funciones gaussianas, por lo que se ha optado en este trabajo en desarrollar el ajuste de los parámetros centrándonos en funciones de pertenencia de este tipo.

Cuando se utilice el Área de Solape entre dos funciones de pertenencia adyacentes se tendrán que tener en cuenta dos factores: que función tiene un área menor de las dos implicadas y la posición que ésta ocupa con respecto a la otra en el universo de discurso. Estas consideraciones nos proporcionan cuatro posibles casos a la hora de calcular el Área de Solape. Cada uno de los casos se ha expuesto en profundidad.

Aparece un nuevo valor que tendrá que decidir el diseñador, el *Área de Solape Deseada*. Esta medida nos indica cuál será, a juicio del diseñador el Área de Solape más adecuada entre dos funciones de pertenencia adyacentes.

El capítulo termina con la descripción detallada del algoritmo que incorpora el procedimiento de ajuste utilizando la Razón de Solape, cuáles son los nuevos parámetros de entrada y como se realizará el nuevo ajuste de los parámetros.

Nuevamente, tenemos como principal referencia del capítulo el trabajo de Marsh [129], en el que se menciona la Robustez de Solape y proporciona la base para poder proponer el Área de Solape como índice para medir el solape entre funciones de pertenencia adyacentes.

7. Aplicaciones

Este capítulo presenta resultados prácticos a partir de las técnicas desarrolladas en los apartados anteriores. En primer lugar se mostrarán ejemplos de casos sencillos y conocidos en el campo de la estimación de funciones mediante sistemas borrosos. Se realizarán comparativas entre los resultados obtenidos realizando el ajuste de las funciones de pertenencia y sin el ajuste de las mismas. Se tratará de justificar la mejora en la interpretabilidad del modelo borroso a costa de la pérdida de precisión en la estimación de la función. En segundo lugar se ha elegido un ejemplo más complejo, un modelo a escala de un helicóptero, basado en el obtenido por Kim y Tilbury en [98]. Se trata de un modelo de múltiple entrada y múltiple salida (MIMO), con el que se pretende demostrar la validez de las técnicas presentadas en este trabajo.

7.1. Ecuación diferencial de segundo orden

El primer ejemplo con el que se va a trabajar es una planta descrita por una ecuación diferencial de segundo orden de la forma [145]:

$$y(k+1) = g[y(k), y(k-1)] + u(k) \quad (7.1)$$

donde

$$g[y(k), y(k-1)] = \frac{y(k)y(k-1)[y(k)+2.5]}{1+y^2(k)+y^2(k-1)} \quad (7.2)$$

$$y \ u(k) = \text{sen}(2\pi k / 25)$$

Si tratamos de modelar, buscando simplemente estimar la función, sin aplicar ningún tipo de ajuste sobre los parámetros (centros y anchos de las funciones de pertenencia y consecuentes), se obtiene un excelente estimación de la función, como se puede apreciar en la Figura 7.1.

Sin embargo, al visualizar las campanas que se han generado para cada una de las variables de entrada del modelo (Figura 7.2 y Figura 7.3), observamos que no presentan una adecuada distribución en el universo de discurso y no cumplen ninguna de las reglas establecidas por Marsh (ver Apartado 5.1 y Figura 5.2).

Si representamos los consecuentes en el universo de discurso, estos tampoco quedan uniformemente distribuidos, incluso, algunos de ellos se encuentran agrupados en una misma zona (Figura 7.4).

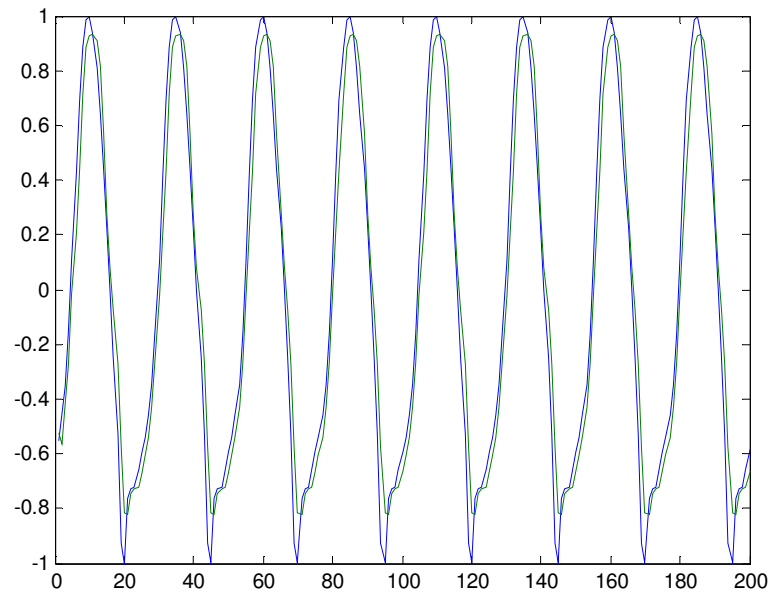


Figura 7.1 Comparación entre la función real y la función estimada para la ecuación (7.1) sin aplicar ajustes sobre los parámetros

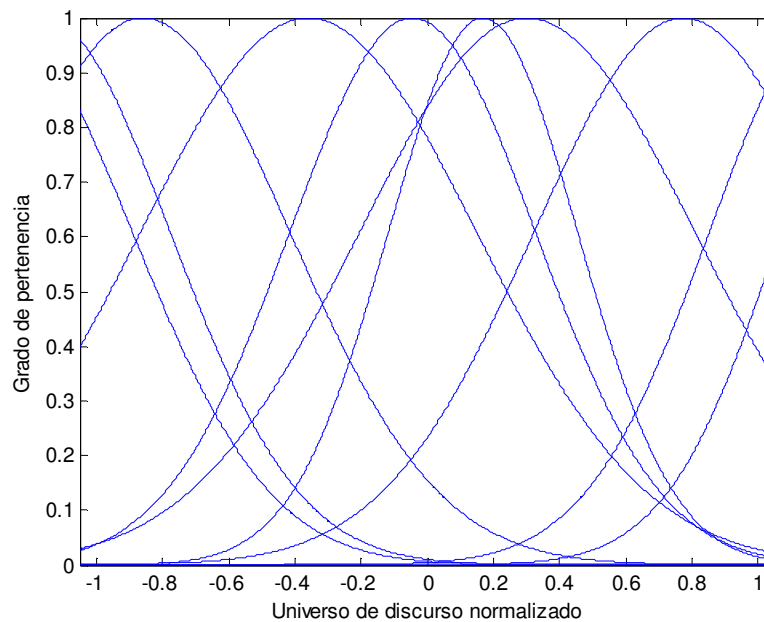


Figura 7.2 Representación de las FP para la 1ª variable de entrada sin aplicar ajustes sobre los parámetros

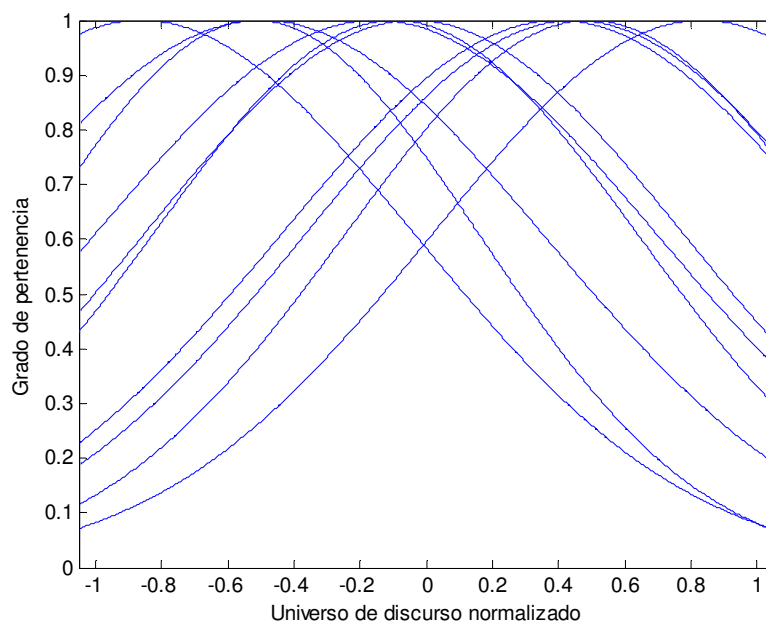


Figura 7.3 Representación de las FP para la 2ª variable de entrada sin aplicar ajustes sobre los parámetros

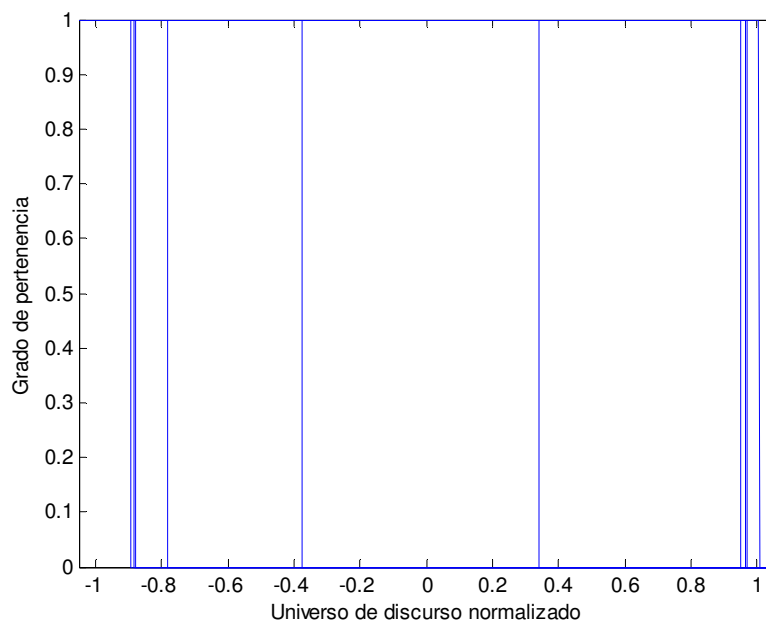


Figura 7.4 Representación de los consecuentes sin aplicar ajustes sobre los parámetros

Estos resultados se han obtenido tras realizar 300 iteraciones, utilizando 10 reglas y con un factor de aprendizaje para la estimación de la función de $\eta = 0.005$. El error que se genera es de 2.13.

Aplicando la razón de solape en el procedimiento de estimación de la función para ajustar los parámetros involucrados en la parte precedente de las reglas, se obtiene el resultado representado en la Figura 7.5.

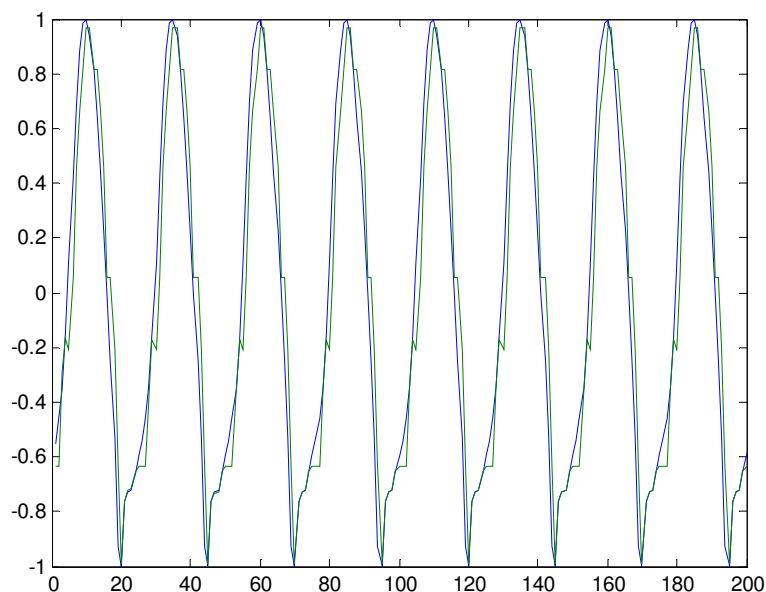


Figura 7.5 Comparación entre la función real y la función estimada para la ecuación (7.1) utilizando la razón de solape en el ajuste de los parámetros de la parte precedente

El error que se consigue al estimar la función aplicando el ajuste sobre los parámetros es un poco mayor (3,31), tras realizar el mismo número de iteraciones (300) y utilizar el mismo número de reglas (10), sin embargo, la distribución de las FP en el universo de discurso mejora considerablemente, como podemos apreciar en la Figura 7.6 y en la Figura 7.7.

Para obtener estos resultados se han utilizado como valores iniciales del algoritmo un factor de aprendizaje para la estimación de la función $\eta = 0.002$, un factor de aprendizaje para la razón de solape $\eta_{RS} = 0.01$, una razón de solape deseada de $RS^d = 0.4$ y un factor de ancho $\psi = 2$.

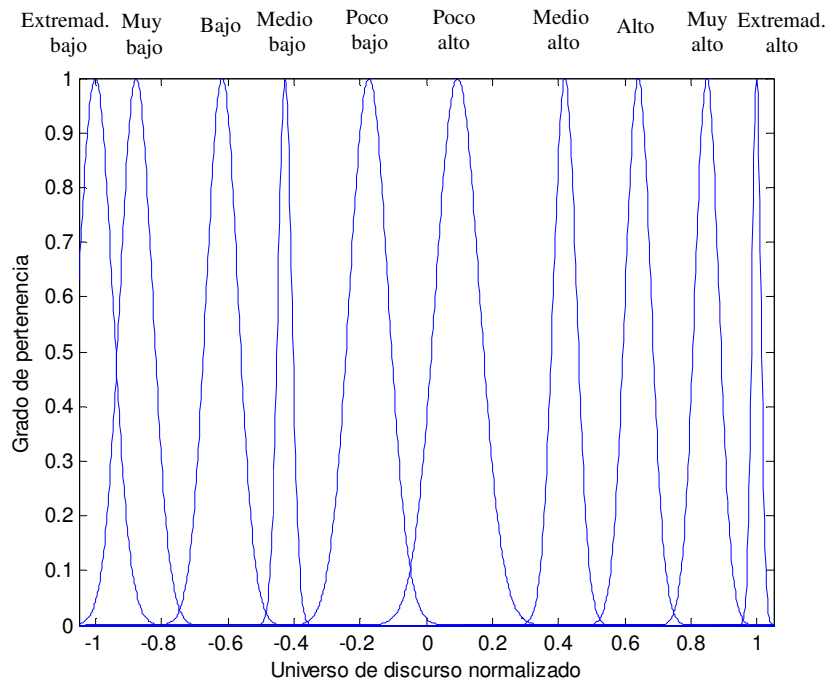


Figura 7.6 Representación de las FP para la 1ª variable de entrada aplicando la razón de solape en el ajuste de los parámetros de la parte precedente

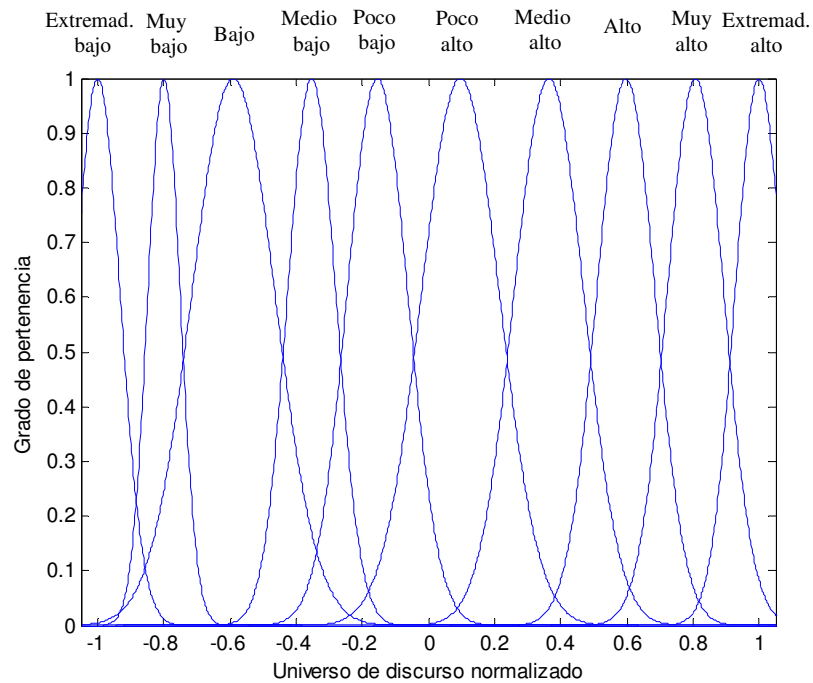


Figura 7.7 Representación de las FP para la 2ª variable de entrada aplicando la razón de solape en el ajuste de los parámetros de la parte precedente

El procedimiento de ajuste no se ha realizado sobre los consecuentes, por lo que éstos siguen situados de forma poco uniforme en el universo de discurso. Esta circunstancia la podemos apreciar al visualizar la distribución de los consecuentes en la Figura 7.8.

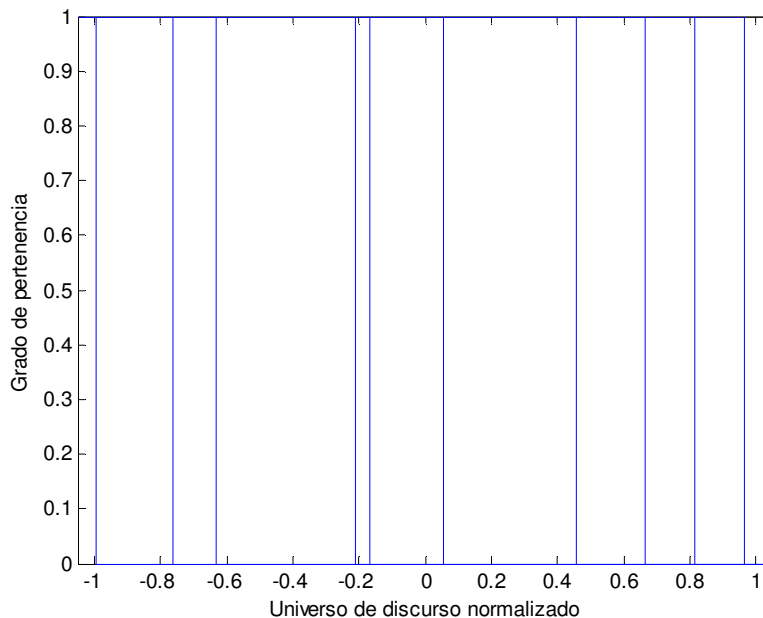


Figura 7.8 Representación de los consecuentes aplicando la razón de solape en el ajuste de los parámetros de la parte precedente

Pasamos a estudiar a continuación el resultado de aplicar el ajuste a los consecuentes de las reglas para que estos estén distribuidos de forma más uniforme en el universo de discurso. Para ello, se utilizan consecuentes de la forma TSK, se aplica el mismo número de épocas (300) y se mantiene el mismo número de reglas (10). Los valores iniciales introducidos al algoritmo son: $\eta = 0.002$, $\eta_{RS} = 0.005$, $RS^d = 0.4$, $\psi = 2$ y como factor de aprendizaje para la razón de solape de los consecuentes $\eta_{RSC} = 0.005$.

El error final en la estimación final de la función es mayor que en los dos casos anteriores (3,69), pero a cambio se consigue una correcta distribución de las FP de la parte precedente y de los consecuentes. La comparativa entre la función real y la estimada se puede apreciar en la Figura 7.9. La distribución de las FP para las variables de entrada se puede ver en la Figura 7.10 y en la Figura 7.11. El resultado de la distribución de los consecuentes se muestra en la Figura 7.12.

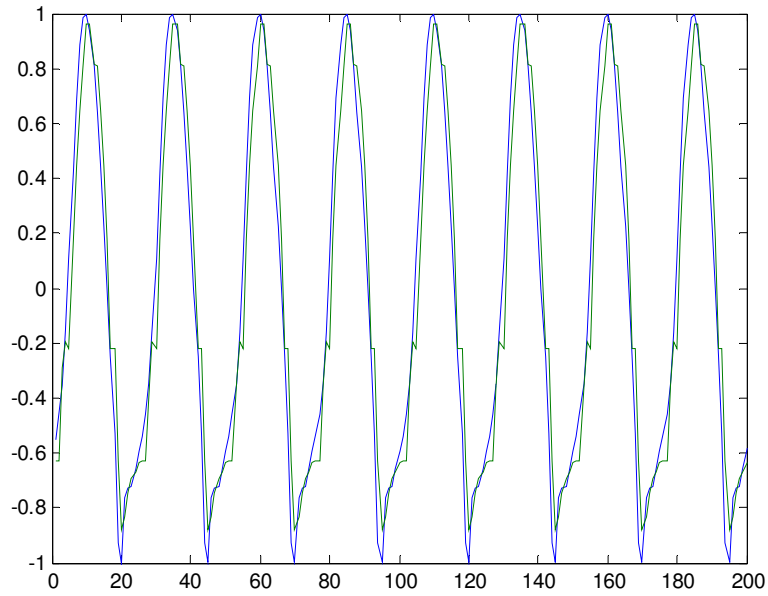


Figura 7.9 Comparación entre la función real y la función estimada para la ecuación (7.1) utilizando la razón de solape en el ajuste de todos los parámetros involucrados en la estimación

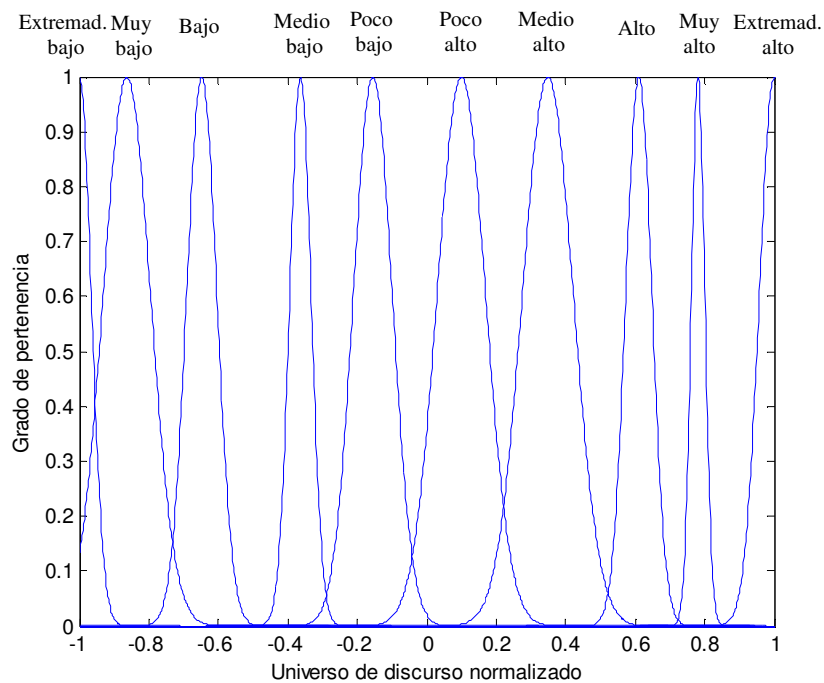


Figura 7.10 Representación de las FP para la 1ª variable de entrada aplicando la razón de solape en el ajuste de todos los parámetros involucrados en la estimación

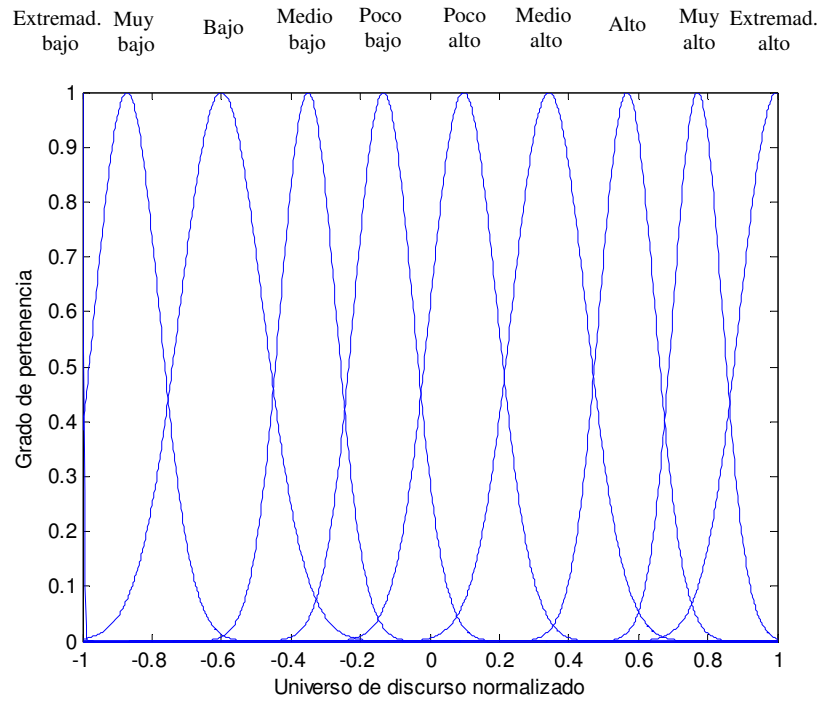


Figura 7.11 Representación de las FP para la 2ª variable de entrada aplicando la razón de solape en el ajuste de todos los parámetros involucrados en la estimación

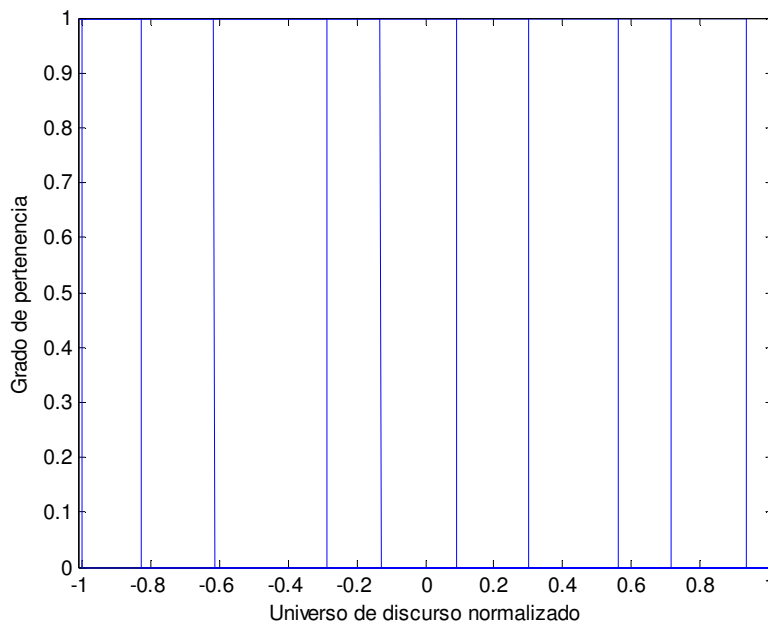


Figura 7.12 Representación de los consecuentes aplicando la razón de solape en el ajuste de los parámetros de la parte precedente

Podemos concluir que la pérdida en la calidad de la estimación de la función es muy pequeña en comparación con la mejora en la distribución de las FP de la parte precedente y de los consecuentes, por lo que el modelo final obtenido tras el ajuste de los parámetros involucrados mejora enormemente la interpretabilidad con respecto al modelo inicial que no realiza ningún tipo de ajuste de los mencionados parámetros.

A medida que tratamos de distribuir de manera más adecuada las funciones en el universo de discurso el error en la estimación de la función aumenta, pero se ve compensado con la mejora en la interpretabilidad del modelo resultante.

Hasta ahora se ha tratado como índice de ajuste la Razón de Solape. Veremos a continuación los resultados aplicando el Área de Solape.

Si el ajuste de los parámetros se produce utilizando como índice de solape el Área de Solape la estimación de la función es muy similar a la que se obtiene utilizando la Razón de solape. Los resultados, utilizando 10 reglas, $\eta = 0.001$, $\eta_{AS} = 0.002$, $\eta_{ASC} = 0.002$, $AS^d = 0.25$, $\psi = 2$ y tras 100 iteraciones del algoritmo se muestran en la Figura 7.13. El error que se produce es de 3,8.

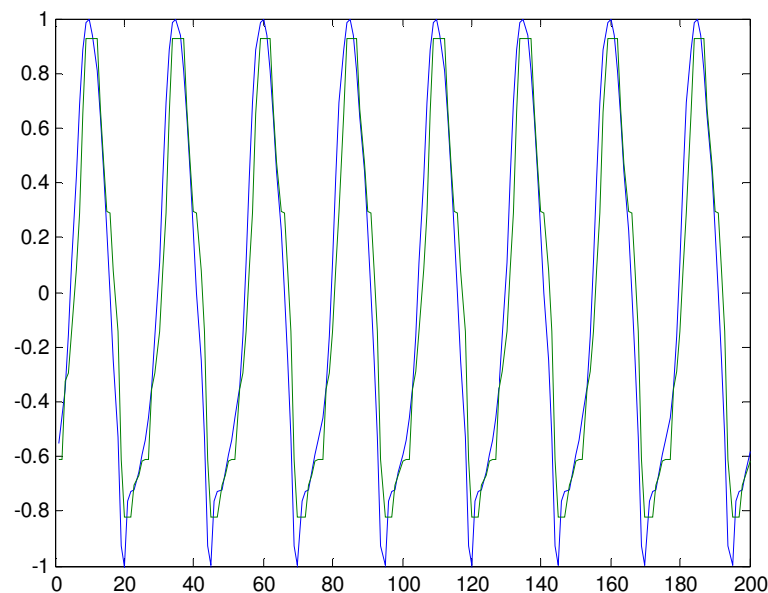


Figura 7.13 Comparación entre la función real y la función estimada para la ecuación (7.1) utilizando el área de solape en el ajuste de los parámetros involucrados en la estimación

La distribución de las FP en el universo de discurso no mejora ni para la primera variable (Figura 7.14) ni para la segunda variable (Figura 7.15), aunque se consigue que su distribución sea más adecuada que si no se utiliza ningún tipo de ajuste.

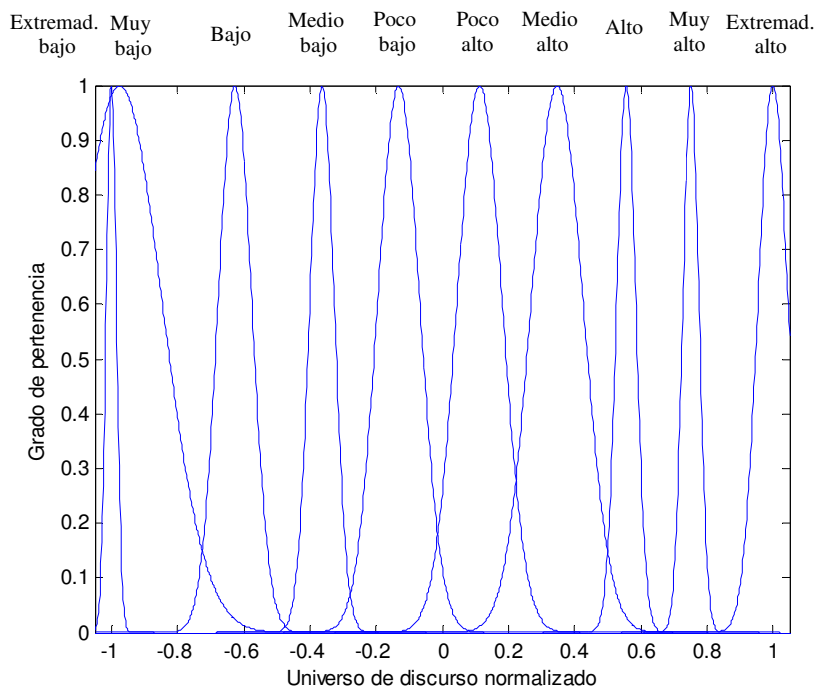


Figura 7.14 Representación de las FP para la 1ª variable de entrada aplicando el área de solape en el ajuste de los parámetros involucrados en la estimación

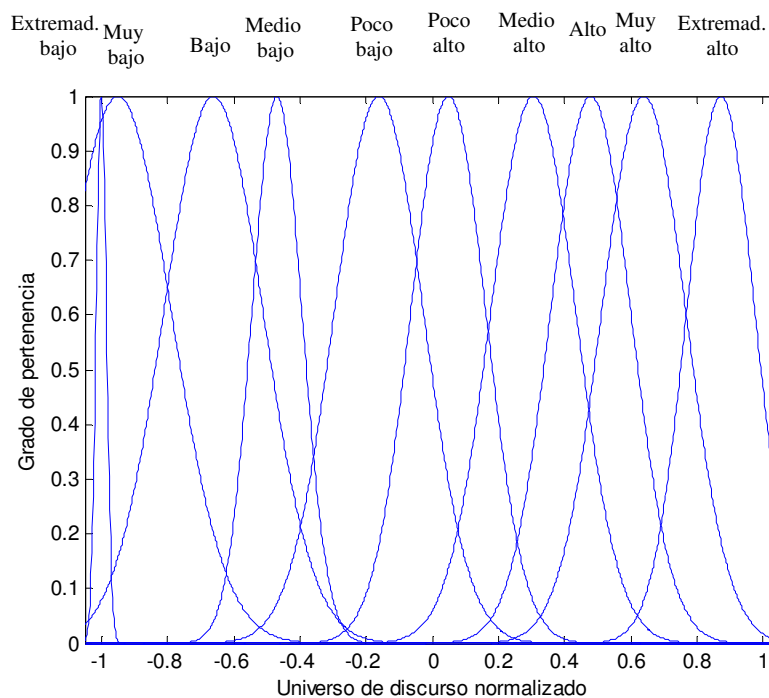


Figura 7.15 Representación de las FP para la 2ª variable de entrada aplicando el área de solape en el ajuste de los parámetros involucrados en la estimación

Los consecuentes quedan correctamente distribuidos en el universo de discurso, como se puede apreciar en la Figura 7.16.

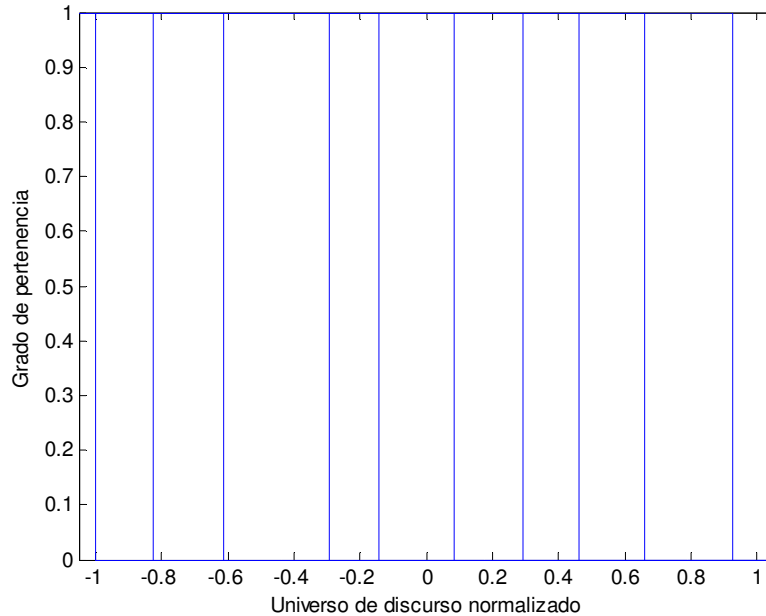


Figura 7.16 Representación de los consecuentes aplicando el área de solape en el ajuste de los parámetros involucrados en la estimación

Por lo tanto, el uso del área de solape en el ajuste genera buenos resultados si lo comparamos con la estimación de la función sin realizar ningún tipo de ajuste, ya que produce una mayor interpretabilidad de los resultados, con un error en la estimación aceptable. Sin embargo, la mejora con respecto al ajuste utilizando razón de solape no se puede apreciar, al menos en este ejemplo. Podemos concluir que los cálculos que se tienen que realizar para estimar con el área de solape son mucho mayores que los necesarios para estimar con la razón de solape y sin embargo los resultados que se obtienen no mejoran considerablemente la solución, por lo que, al menos en este caso resulta más aconsejable la razón de solape.

Veamos como se comportan estos índices de solape con una función mucho más compleja de estimar.

Los resultados de estimar esta función sin realizar ningún tipo de ajuste sobre los parámetros pueden verse en la Figura 7.18. Se puede considerar que la estimación que se consigue es muy buena y con 500 épocas y 8 reglas se obtiene un error de 0,7, si utilizamos un factor de aprendizaje de 0.002.

Las funciones de pertenencia que se generan para cada una de las variables no se encuentran correctamente distribuidas y no nos proporcionan interpretabilidad. La Figura 7.19 corresponde a la variable $u(k-4)$ y la Figura 7.20 a la variable $y(k-1)$.

Los consecuentes sufren una distribución inadecuada y se reparten por el universo de discurso de forma muy poco uniforme, por lo que su interpretabilidad es muy escasa. La Figura 7.21 nos muestra la distribución que se obtiene de los consecuentes.

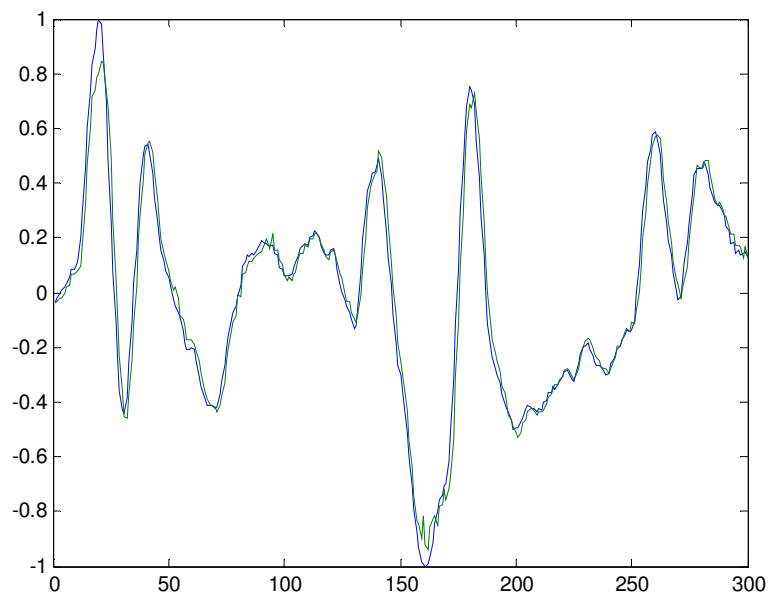


Figura 7.18 Comparación entre la función real y la función estimada para el modelo de Box y Jenkins sin aplicar ajustes sobre los parámetros

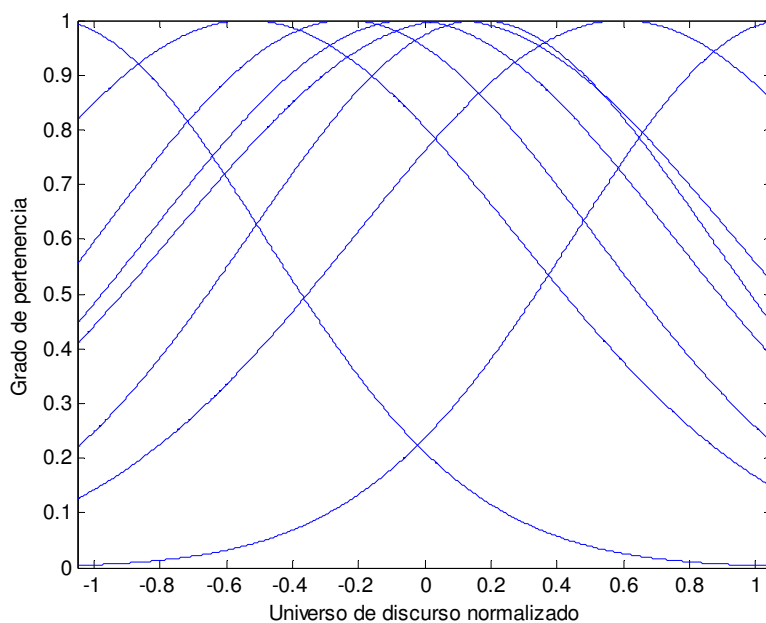


Figura 7.19 Representación de las FP para la 1ª variable de entrada sin aplicar ajustes sobre los parámetros en el modelo de Box y Jenkins

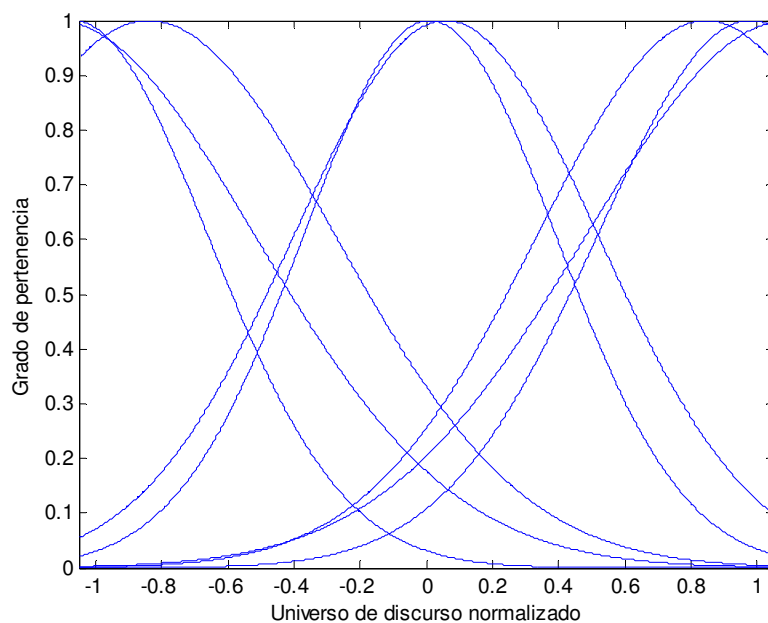


Figura 7.20 Representación de las FP para la 2ª variable de entrada sin aplicar ajustes sobre los parámetros en el modelo de Box y Jenkins

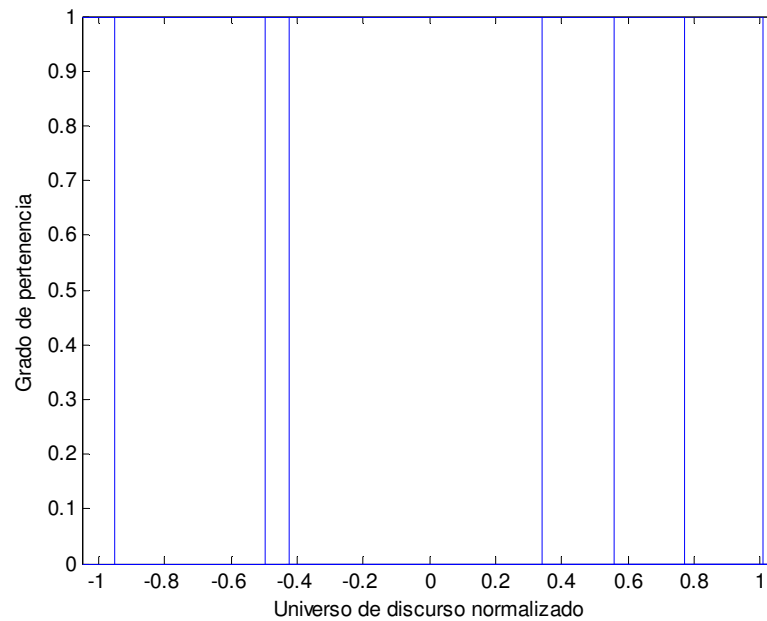


Figura 7.21 Representación de los consecuentes sin aplicar ajustes sobre los parámetros en el modelo de Box y Jenkins

Cuando aplicamos ajuste sobre los centros y anchos de las FP, pero no sobre los consecuentes se producen los resultados que se muestran a continuación. Los resultados no son los esperados cuando buscamos estimar la función lo mejor posible ajustando las funciones. Como se puede apreciar en la Figura 7.22, el error en la estimación aumenta a 1,49, pero la distribución de la primera variable (Figura 7.23) no es la adecuada, aunque la distribución de la segunda variable se comporta bastante mejor (Figura 7.24).

Estos resultados se obtienen tras ejecutar el algoritmo durante 400 épocas, con 8 reglas y con un factor de aprendizaje $\eta = 0.001$, un factor de aprendizaje para la Razón de Solape $\eta_{RS} = 0.002$, una Razón de Solape Deseada $RS^d = 0.4$, y un Factor de Ancho $\psi = 2$.

Los consecuentes se encuentran muy mal distribuidos en el universo de discurso como se aprecia en la Figura 7.25.

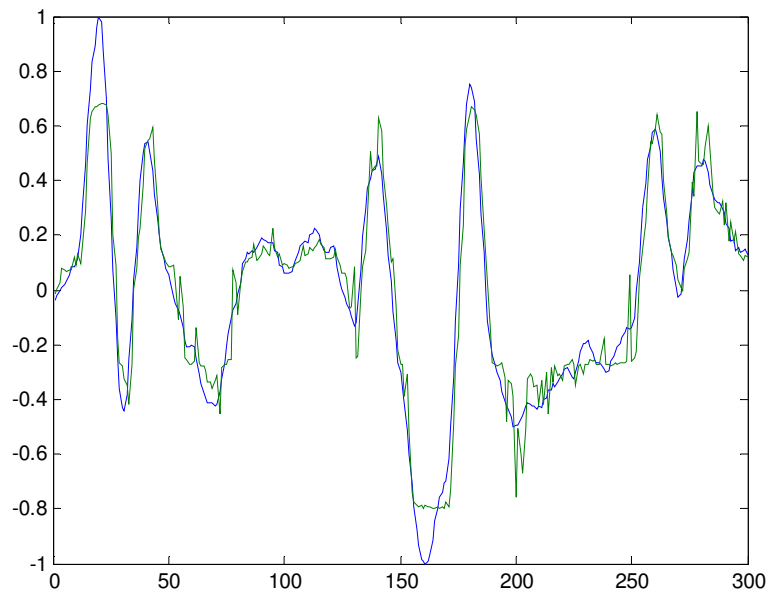


Figura 7.22 Representación de la función real y la función estimada aplicando ajuste sólo sobre los parámetros de la parte precedente en el modelo de Box y Jenkins

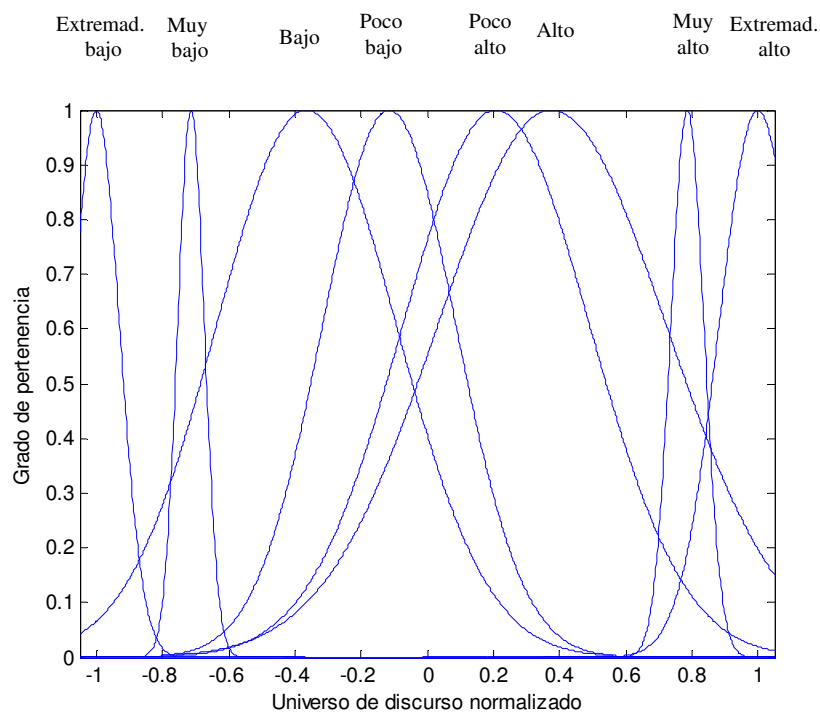


Figura 7.23 Representación de las FP para la 1ª variable de entrada aplicando ajuste sobre los parámetros de la parte precedente en el modelo de Box y Jenkins

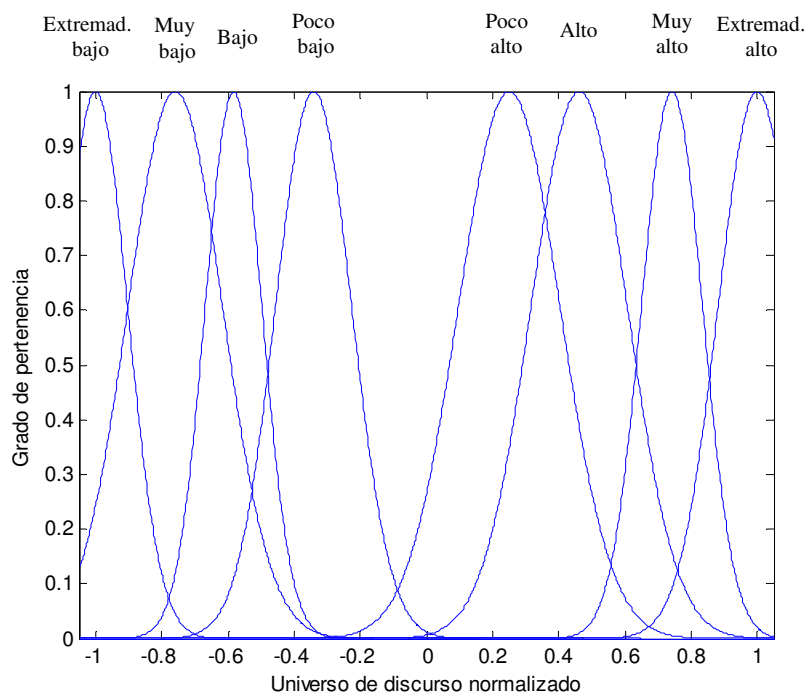


Figura 7.24 Representación de las FP para la 2ª variable de entrada aplicando ajuste sobre los parámetros de la parte precedente en el modelo de Box v Jenkins

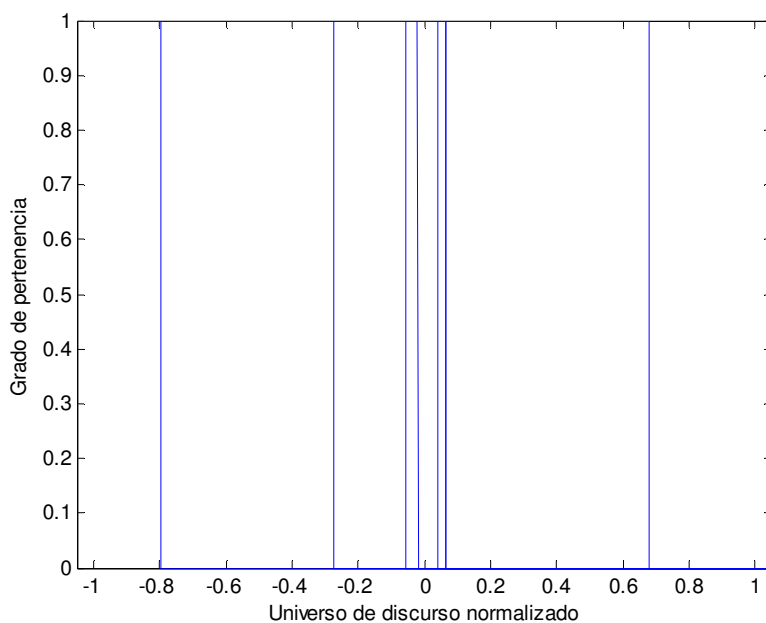


Figura 7.25 Representación de los consecuentes aplicando la razón de solape en el ajuste de los parámetros de la parte precedente en el modelo de Box y Jenkins

Si tratamos de distribuir más adecuadamente las funciones en el universo de discurso, el error que se comete al estimar la función se eleva a 3,6. La comparativa entre la función real y la estimada se muestra en la Figura 7.26.

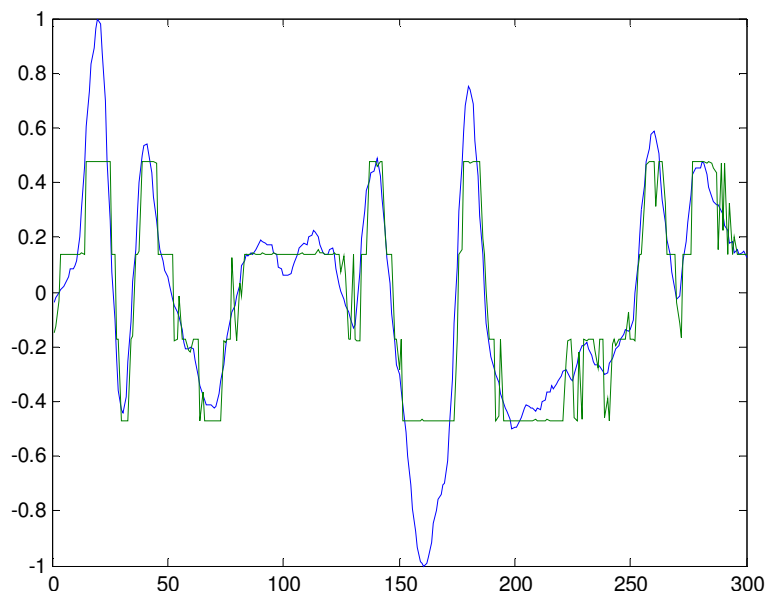


Figura 7.26 Representación de la función real y la función estimada aplicando ajuste sobre los parámetros de la parte precedente en el modelo de Box y Jenkins

Sin embargo, como se aprecia en la Figura 7.27 y en la Figura 7.28, la distribución de las funciones mejora considerablemente.

Los consecuentes, al no realizarse ningún tipo de ajuste sobre ellos quedan muy mal distribuidos en el universo de discurso. En la Figura 7.29 se puede apreciar la distribución de los mismos.

Estos resultados se han obtenido tras 500 iteraciones, con 8 reglas, $\eta = 0.0001$, $\eta_{RS} = 0.001$, $RS^d = 0.4$ y $\psi = 2$

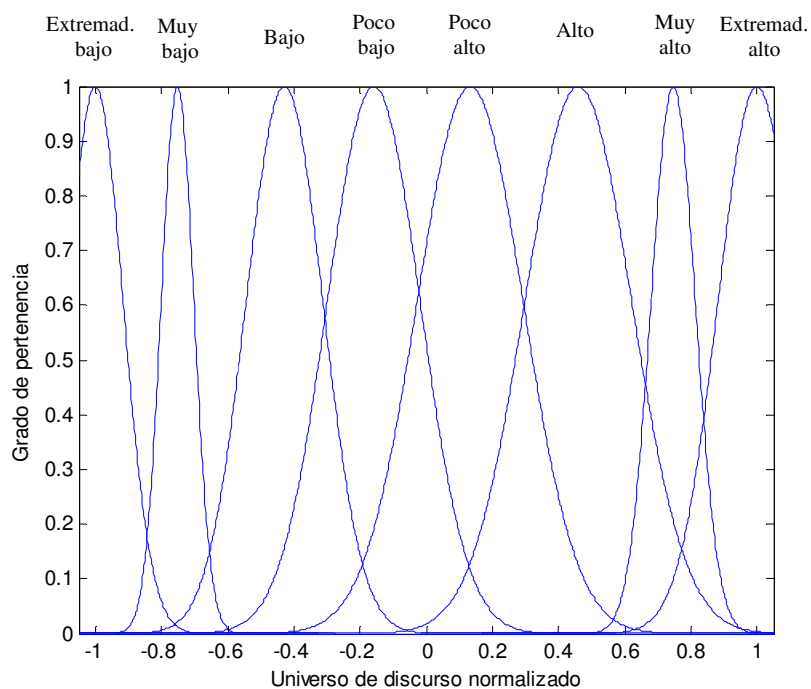


Figura 7.27 Representación de las FP para la 1ª variable de entrada aplicando ajuste sobre los parámetros de la parte precedente en el modelo de Box y Jenkins

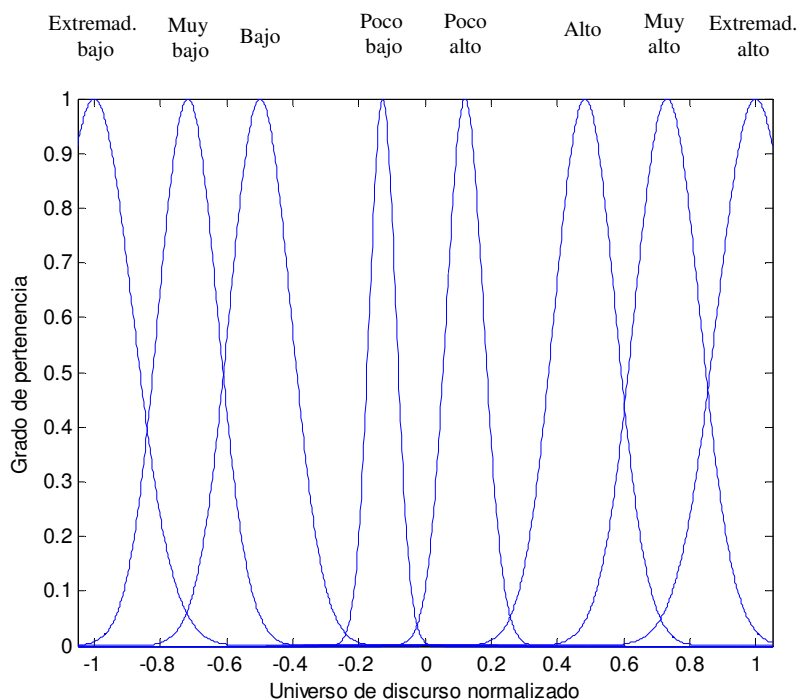


Figura 7.28 Representación de las FP para la 2ª variable de entrada aplicando ajuste sobre los parámetros de la parte precedente en el modelo de Box y Jenkins

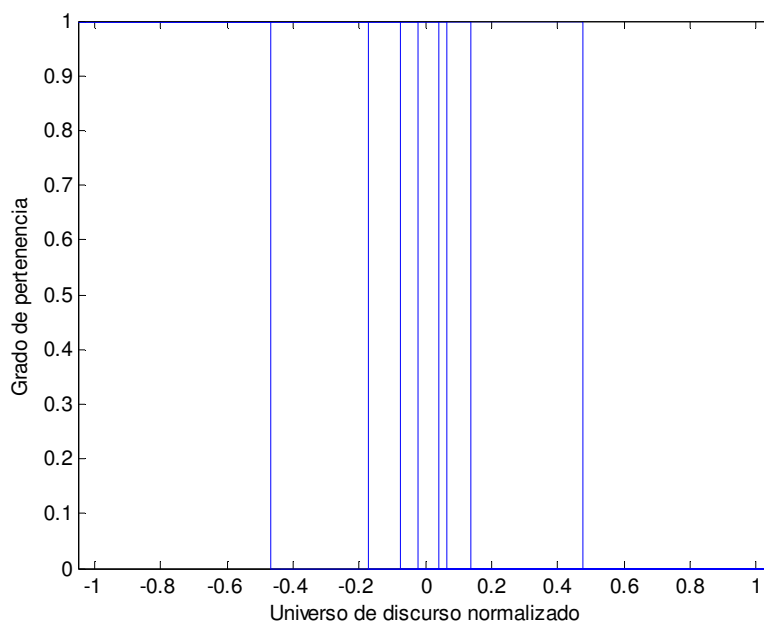


Figura 7.29 Representación de los consecuentes aplicando la razón de solape en el ajuste de los parámetros de la parte precedente en el modelo de Box y Jenkins

Los resultados mejoran considerablemente si realizamos el ajuste tanto de la parte precedente como del consecuente. En la estimación de la función se produce un error mayor (1,27) que sin realizar el ajuste de los parámetros, pero a cambio ganamos una mayor interpretabilidad.

Los valores con los que se ha ejecutado el algoritmo son: 1300 épocas, 8 reglas, $\eta = 0.0002$, $\eta_{RS} = 0.0005$, $\eta_{RSC} = 0.001$, $RS^d = 0.4$ y $\psi = 2$.

La Figura 7.30, muestra la función real y la estimada aplicando el ajuste a todos los parámetros. La distribución de la primera variable se aprecia en la Figura 7.31 y la distribución de la segunda en la Figura 7.32. Los consecuentes se pueden apreciar en la Figura 7.33.

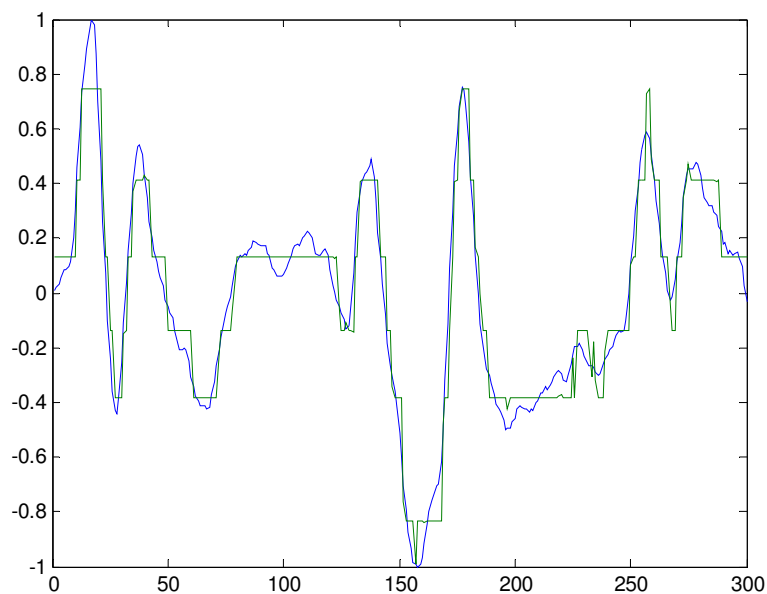


Figura 7.30 Comparación entre la función real y la función estimada para el modelo de Box y Jenkins realizando el ajuste a todos los parámetros involucrados en la estimación

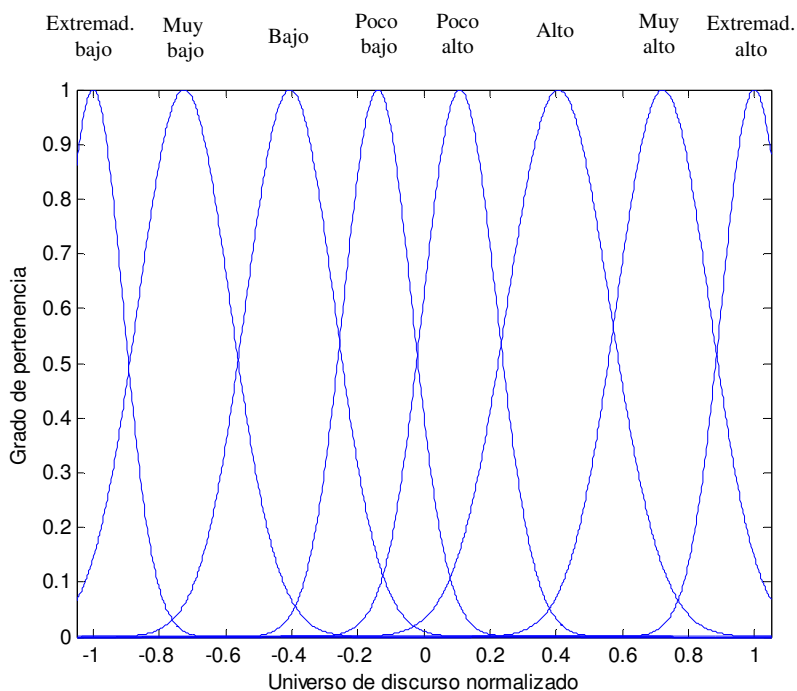


Figura 7.31 Representación de las FP para la 1ª variable de entrada realizando el ajuste a todos los parámetros involucrados en la estimación del modelo de Box y Jenkins

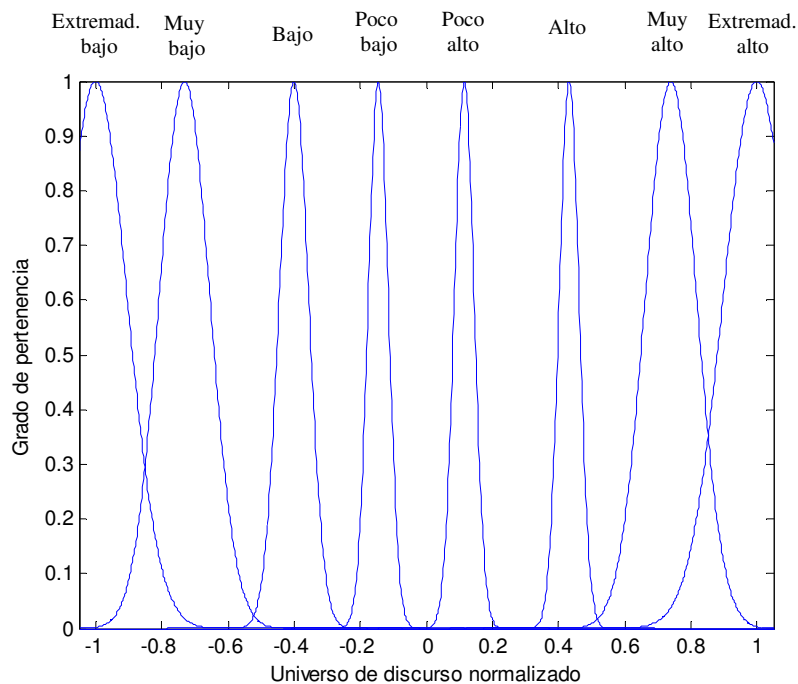


Figura 7.32 Representación de las FP para la 2ª variable de entrada realizando el ajuste a todos los parámetros involucrados en la estimación del modelo de Box y Jenkins

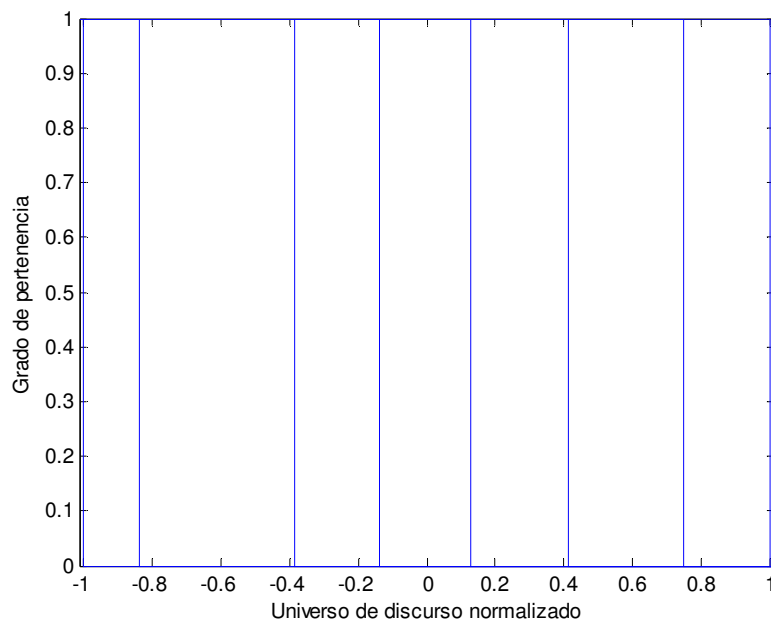


Figura 7.33 Representación de los consecuentes aplicando la razón de solape en el ajuste a todos los parámetros en el modelo de Box y Jenkins

Si utilizamos como índice para ajustar los parámetros implicados en la estimación el área de solape en lugar de la razón de solape, se obtienen nuevamente, al igual que en ejemplo mostrado en la sección 7.1, resultados similares.

Los resultados, utilizando 8 reglas, $\eta = 0.004$, $\eta_{AS} = 0.01$, $\eta_{ASC} = 0.01$, $AS^d = 0.25$, $\psi = 2$ y tras 100 iteraciones del algoritmo se muestran en la Figura 7.34. El error que se produce es de 1,3.

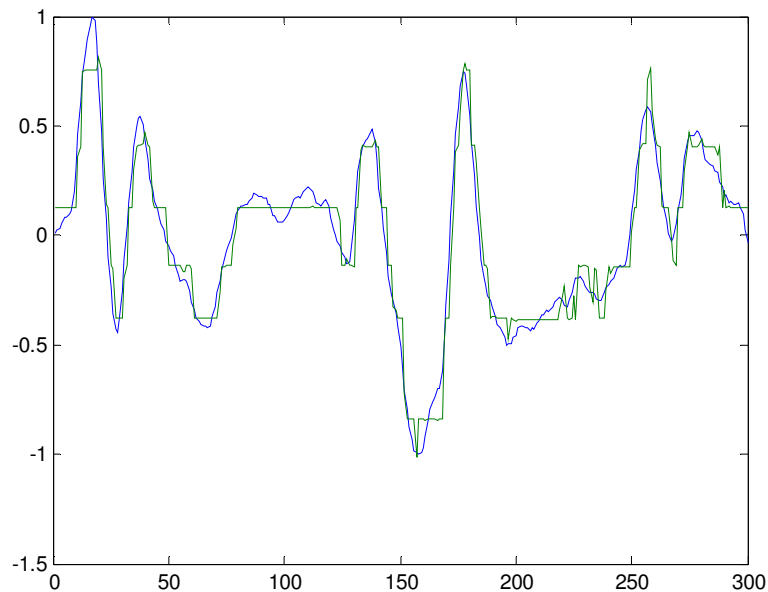


Figura 7.34 Comparación entre la función real y la función estimada para el modelo de Box y Jenkins realizando el ajuste a todos los parámetros involucrados en la estimación usando el área de solape

El error que se comete es similar al conseguido usando la razón de solape, sin embargo la distribución de las FP en el universo de discurso mejora considerablemente, por lo que la interpretabilidad del modelo en este caso es mucho mayor.

La representación de las funciones para la primera variable implicada se puede apreciar en la Figura 7.35. La segunda variable se muestra en la Figura 7.36.

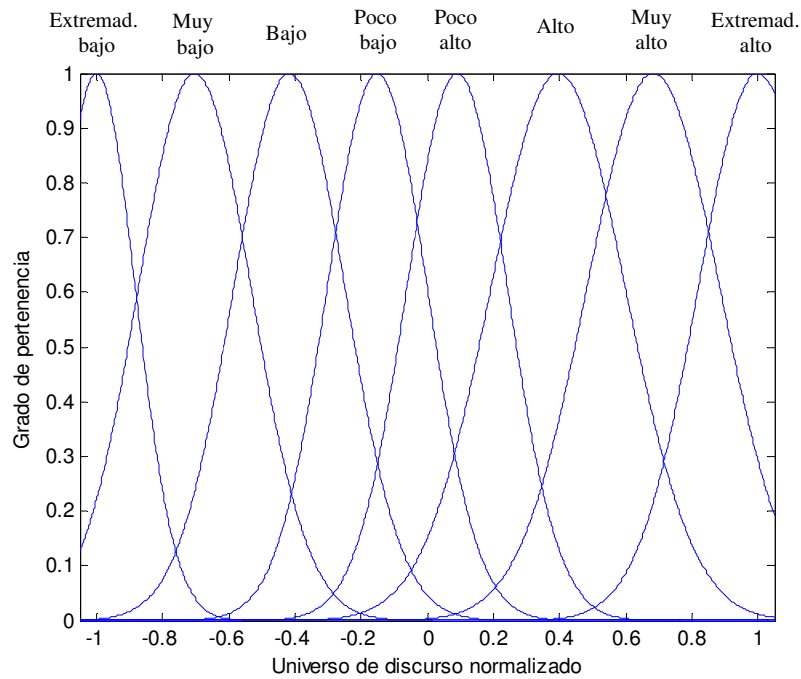


Figura 7.35 Representación de las FP para la 1ª variable de entrada realizando el ajuste a todos los parámetros involucrados en la estimación del modelo de Box y Jenkins usando el área de solape

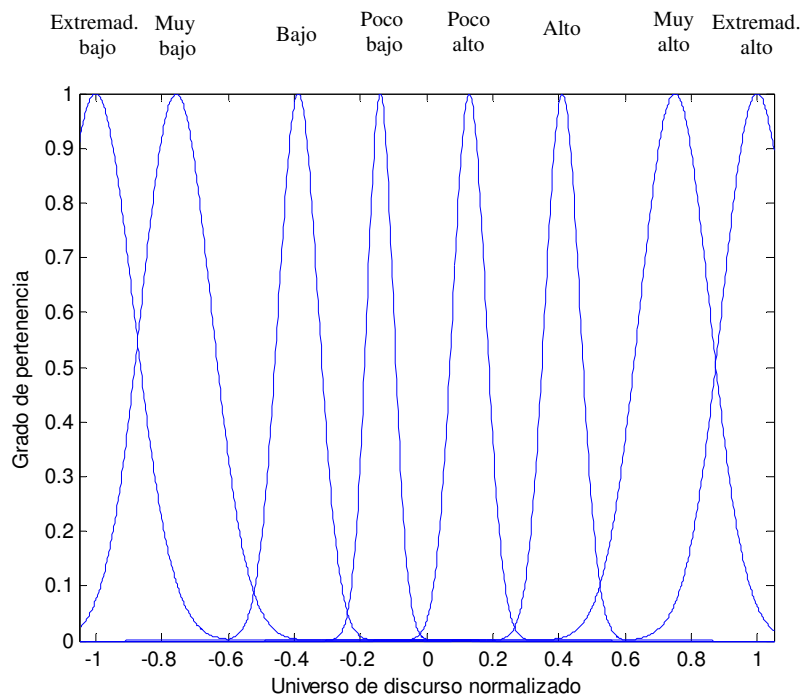


Figura 7.36 Representación de las FP para la 2ª variable de entrada realizando el ajuste a todos los parámetros involucrados en la estimación del modelo de Box y Jenkins usando el área de solape

Los consecuentes quedan distribuidos como se muestran en la Figura 7.37.

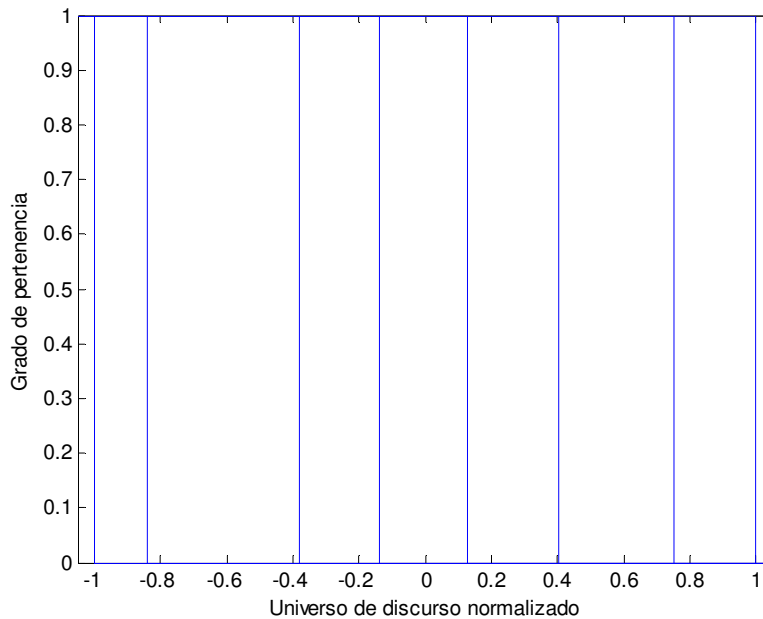


Figura 7.37 Representación de los consecuentes aplicando el área de solape en el ajuste a todos los parámetros en el modelo de Box y Jenkins

7.3. Helicóptero de aeromodelismo

El último ejemplo que se muestra es el más complejo de los que se presentan en este trabajo. Se trata del modelo matemático de un helicóptero de aeromodelismo. Para describir de manera general el sistema se realiza a continuación una traducción libre de un fragmento del artículo de Kim-Tilbury: “Mathematical Modeling and Experimental Identification of Model Helicopter” [98].

7.3.1. Aspectos técnicos previos

Aunque un helicóptero de aeromodelismo puede simplemente parecer una miniaturización de su versión a gran escala, hay diferencias notables entre los dos. La primera diferencia principal entre el de aeromodelismo y el de gran escala está relacionada con la configuración del rotor. Muchos helicópteros a gran escala tienen una correa, ya sea de aleteo libre o montada sobre muelle, en las aspas del rotor, de tal manera que el plano del rotor puede ser inclinado respecto del helicóptero.

Un sistema de articulaciones permite a las aspas del rotor el aleteo, aumentando la estabilidad. Sin embargo, este comportamiento de aleteo aumenta el tiempo que necesita el helicóptero para responder a las entradas de control. Inclinando el disco del rotor hacia delante, el helicóptero puede desplazarse hacia delante mientras se mantiene el nivel [94].

La mayoría de los helicópteros de aerodelismo poseen un diseño del rotor rígido sin articulaciones, el cual fuerza que la posición del fuselaje permanezca fija respecto al disco del rotor. Esto trae como resultado unos tiempos de respuesta más rápidos, y proporciona al piloto remoto una mejor sensación de movimiento del helicóptero. Sin embargo, para moverse hacia delante, el helicóptero deberá primero inclinarse hacia delante, lo que produce una componente hacia delante del vector de propulsión principal.

En segundo lugar, las propiedades de estabilidad son bastante diferentes. La respuesta temporal de un helicóptero de aerodelismo es mucho más rápida debido a su reducido tamaño. Sin la ayuda de ningún dispositivo extra de aumento de la estabilidad, un piloto humano tendrá dificultades para controlarlo.

En la mayoría de los helicópteros de aerodelismo, un gran giróscopo de control con un perfil aerodinámico, llamado FLYBAR, se usa para mejorar la característica de estabilidad alrededor de los ejes de *pitch* y *roll* y para minimizar el esfuerzo requerido en el actuador.

Además se usa un giróscopo electrónico para estabilizar aún más los ejes de *yaw*. En la mayoría de los helicópteros a escala completa, las inercias de un rotor y fuselaje tan grandes y la articulación que une el rotor proporcionan una estabilidad adecuada, y son innecesarios los giróscopos extra de control en los rotores.

Debido a estas diferencias considerables, un modelo matemático de un helicóptero de aerodelismo no puede ser sencillamente una versión a escala reducida del modelo a escala completa; se deberán tener en cuenta los diferentes mecanismos de actuación.

A continuación, se muestran las ecuaciones dinámicas del movimiento del helicóptero de aerodelismo incluyendo la dinámica del actuador. Las ecuaciones se basan en la dinámica del sólido rígido [144], así como en la aerodinámica básica y la teoría de helicópteros [94][152].

7.3.2. Modelo completo

En la Figura 7.38 se muestra un diagrama de bloques del modelo matemático completo del helicóptero. Las entradas δ son las cuatro posiciones de las palancas del helicóptero de aeromodelismo, y las salidas q son las posiciones, orientaciones y velocidades del sólido rígido. Hay variables internas, incluyendo la orientación del rotor ξ , el ángulo de aleteo de la flybar β , y el ángulo de pitch colectivo θ_0 .

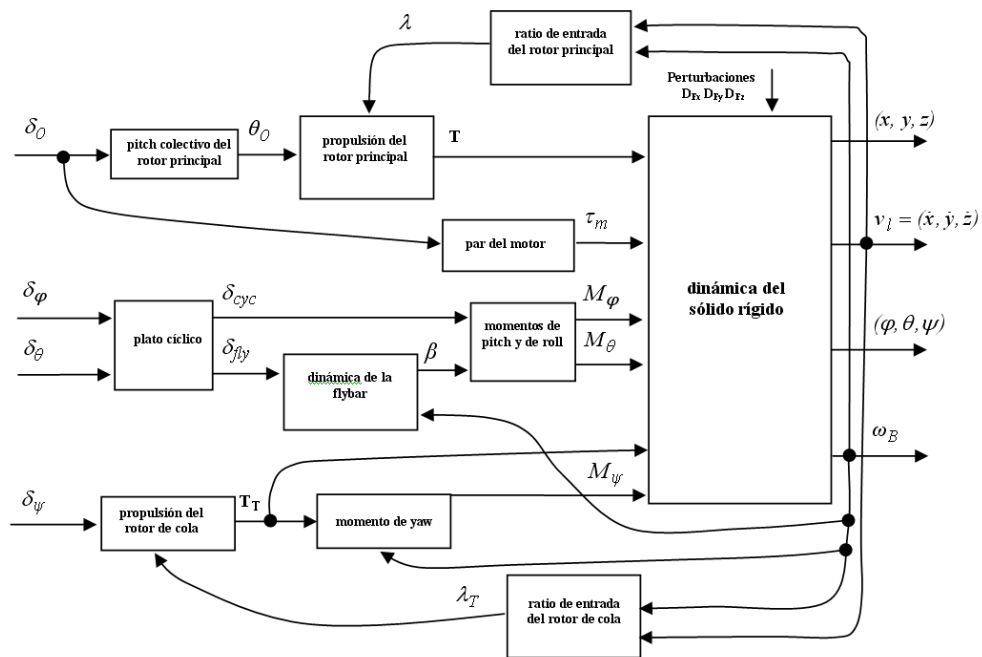


Figura 7.38 Diagrama de bloques del modelo matemático del helicóptero

El siguiente conjunto de ecuaciones diferenciales e integrales relaciona la salida con la entrada:

Pitch colectivo del rotor principal y par del motor

$$\ddot{\theta}_0 + K_5 \dot{\theta}_0 + K_6 \theta_0 = K_4 \delta_0 \tag{7.6}$$

$$\tau_m = -K_m \cdot \delta_0 \tag{7.7}$$

Propulsiones del rotor principal y de cola

$$T = \frac{n}{2} ac \rho \pi R^3 \Omega^2 \left(\frac{B^3}{3} \theta_0 - \frac{B^2}{2} \lambda \right) \quad (7.8)$$

$$T_T = \frac{n}{2} a_T c_T \rho \pi R_T^3 \Omega_T^2 \left(\frac{B^3}{3} \delta_\psi - \frac{B^2}{2} \lambda_T \right) \quad (7.9)$$

Dinámica del sólido rígido

$$q = [x_I \ y_I \ z_I \ \dot{x}_I \ \dot{y}_I \ \dot{z}_I \ \phi \ \theta \ \psi \ \omega_{B1} \ \omega_{B2} \ \omega_{B3}]^T$$

$$\dot{q} = \begin{bmatrix} v_I \\ \frac{1}{m} R_{IB} f_B \\ \Upsilon \omega_B \\ \Gamma^{-1} (\tau_B - \omega_B \times I \omega_B) \end{bmatrix} \quad (7.10)$$

$$f_B = \begin{bmatrix} -D_{F_x} \\ -D_{F_y} - T_T \\ -D_{F_z} - T_T \end{bmatrix} + R_{IB}^T \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (7.11)$$

$$\tau_B = \begin{bmatrix} M_\phi \\ M_\theta - T \ell_r \\ M_\psi + \tau_m \end{bmatrix} \quad (7.12)$$

Momentos de pitch y de roll

$$\begin{bmatrix} M_\phi \\ M_\theta \end{bmatrix} = \frac{nac\rho\Omega^2}{4\pi} \int_0^{2\pi} \int_0^{BR} \begin{bmatrix} -\sin\left(\xi + \frac{\pi}{2}\right) \\ \cos\left(\xi + \frac{\pi}{2}\right) \end{bmatrix} r^3 \theta_{cyc} dr d\xi \quad (7.13)$$

$$dL_m = \frac{1}{2} \rho (\Omega r)^2 a \theta_{cyc} c dr \quad (7.14)$$

Momento de Yaw

$$M_\psi = T_T L_T - K_g \omega_{B3} \quad (7.15)$$

Plato cíclico

$$\delta_{cyc}(\xi) = -\frac{L_8}{L_9}(\delta_\theta \sin \xi + \delta_\phi \cos \xi) \quad (7.16)$$

$$\delta_{fly}(\xi) = -\frac{L_8}{L_9}(\delta_\theta \cos \xi - \delta_\phi \sin \xi) \quad (7.17)$$

Dinámica de la flybar

$$\omega_F = \begin{bmatrix} -\omega_{B3} \sin \beta + \cos \beta (\omega_{B1} \cos \xi' + \omega_{B2} \sin \xi') \\ \omega_{B2} \cos \xi' - \omega_{B1} \sin \xi' \\ \omega_{B3} \cos \beta + \sin \beta (\omega_{B1} \cos \xi' + \omega_{B2} \sin \xi') \end{bmatrix} + \begin{bmatrix} -\xi' \sin \beta \\ \dot{\beta} \\ \xi' \cos \beta \end{bmatrix} \quad (7.18)$$

$$-I_f \omega_{F1} \omega_{F3} + I_f \dot{\omega}_{F2} = n \int_{R_1}^{BR_2} r \frac{1}{2} \rho (\Omega r)^2 a \left(\frac{\alpha_1 L_7}{L_5 L_6} \delta_{fly} - \frac{\alpha_2 \omega_{F2}}{\Omega} \right) c_2 dr \quad (7.19)$$

Ratios de entrada del rotor principal y de cola

$$\lambda = -v + \frac{v_i}{R\Omega} \quad (7.20)$$

$$\lambda_T = -v_T + \frac{v_{iT}}{R_T \Omega_T} \quad (7.21)$$

Este conjunto de ecuaciones del sistema no se puede simular fácilmente ya que el ángulo de aleteo de la flybar β es función del ángulo del rotor ξ , pues la ecuación del momento de roll (7.13) no se puede resolver directamente a menos que se exprese β de forma explícita en términos de ξ . Para poder hacer simulaciones, se aproxima el ángulo de aleteo β con una función senoidal en cada vuelta:

$$\beta \approx \beta_{\max} \cos(\xi - \xi_{\max}) \quad (7.22)$$

donde

$$\beta_{\max} = \max_{0 \leq \xi < 2\pi} \beta(\xi)$$

$$\xi_{\max} = \arg \max_{0 \leq \xi < 2\pi} \beta(\xi)$$

Usando esta aproximación, se pueden resolver analíticamente las integrales de la ecuación (7.13):

$$M_{\phi} = \frac{n\rho\Omega^2 acR^4 B^4}{16L_1(L_2 + L_3)} \left(\alpha_3 \left(\frac{L_3 L_8}{L_9} \delta_{\phi} - L_2 L_4 \beta_{\max} \cos \xi_{\max} \right) - \alpha_4 L_1 (L_2 + L_3) \frac{\omega_{B1}}{\Omega} \right)$$

$$M_{\theta} = \frac{n\rho\Omega^2 acR^4 B^4}{16L_1(L_2 + L_3)} \left(\alpha_3 \left(\frac{L_3 L_8}{L_9} \delta_{\theta} - L_2 L_4 \beta_{\max} \sin \xi_{\max} \right) - \alpha_4 L_1 (L_2 + L_3) \frac{\omega_{B2}}{\Omega} \right)$$

El modelo del sistema puede reducirse por lo tanto a un conjunto de ecuaciones puramente diferenciales. Se ha construido un diagrama de bloques de *Simulink* del modelo, y se ha usado para comparar el modelo con los datos obtenidos. En la simulación, los parámetros β_{\max} y ξ_{\max} se hallan en cada vuelta del aspa del rotor. Los términos β y ω_B en los momentos M_{ϕ} y M_{θ} de pitch y de roll sirven como términos de amortiguación en las ecuaciones del sólido rígido, aumentando la estabilidad del sistema con respecto al modelo del doble-integrador.

7.3.3. Estimación del modelo

Nos encontramos, por tanto, con un modelo MIMO, de 4 variables de entrada y 12 variables de salida. El aumento en la complejidad es considerable. Para poder aplicar las técnicas expuestas se trabajará con 12 modelos en paralelo, cada uno con 4 variables de entrada y una sola variable de salida (un modelo para cada variable de salida).

A continuación sólo se muestran algunos de los resultados obtenidos, y se comparan algunas de las distribuciones más significativas, ya que mostrar todos los gráficos comparando función real y estimada, las 4 variables de entrada y consecuentes, sin aplicar ajuste y posteriormente aplicando ajuste con razón de solape, ajuste de consecuentes y área de solape, supondría visualizar 24 gráficos sólo para un modelo, lo que generaría unos 288 gráficos si contamos los 12 modelos.

De manera representativa, el modelo 3 que estima la posición z del helicóptero en el espacio, y el modelo 4 que corresponde con la componente \dot{x} de la velocidad del mismo, se han elegido para mostrar los resultados.

Si mostramos la comparativa entre la función real y la función estimada (Figura 7.39), así como la distribución de las funciones de pertenencia para la primera variable (δ_0) en el modelo 3, sin aplicar ningún tipo de ajuste (Figura 7.40), utilizando 5 reglas, 200 épocas y

un factor de aprendizaje $\eta = 0.01$, podemos apreciar como las campanas no se distribuyen adecuadamente, aunque se produce un error muy pequeño, de tan sólo 0,42.

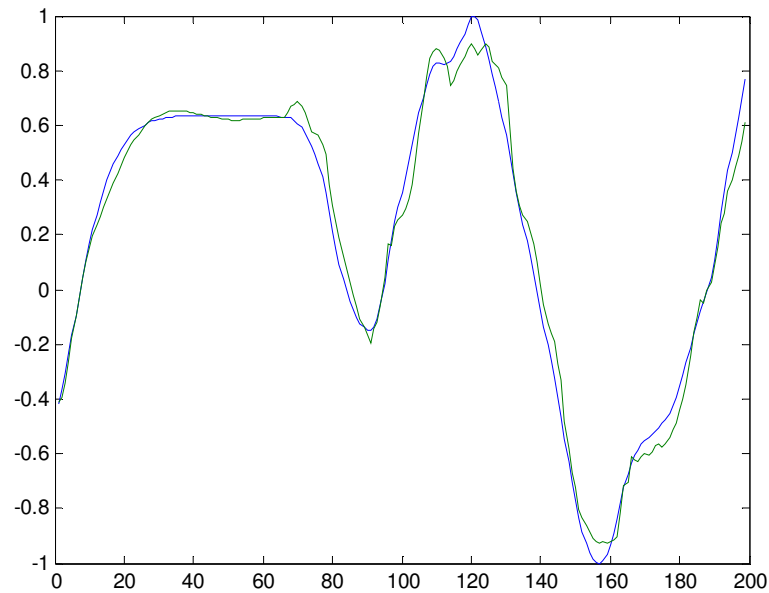


Figura 7.39 Comparación entre la función real y la función estimada (variable z) en el modelo 3 del helicóptero sin realizar el ajuste a los parámetros involucrados en la estimación

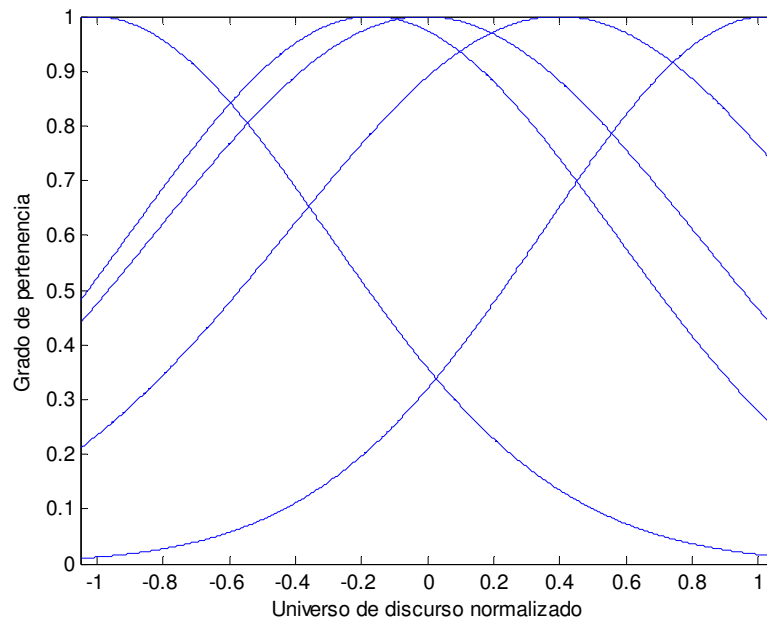


Figura 7.40 Representación de las FP para la 1ª variable de entrada (δ_0) sin realizar ajuste a los parámetros involucrados en la estimación del modelo 3

Lo mismo ocurre para la tercera variable (δ_θ) en el modelo 4. En este caso se han utilizado también 5 reglas, 200 iteraciones y $\eta = 0.01$. El error que se produce es de 0,42. Los resultados los podemos apreciar en la Figura 7.41 y en la Figura 7.42.

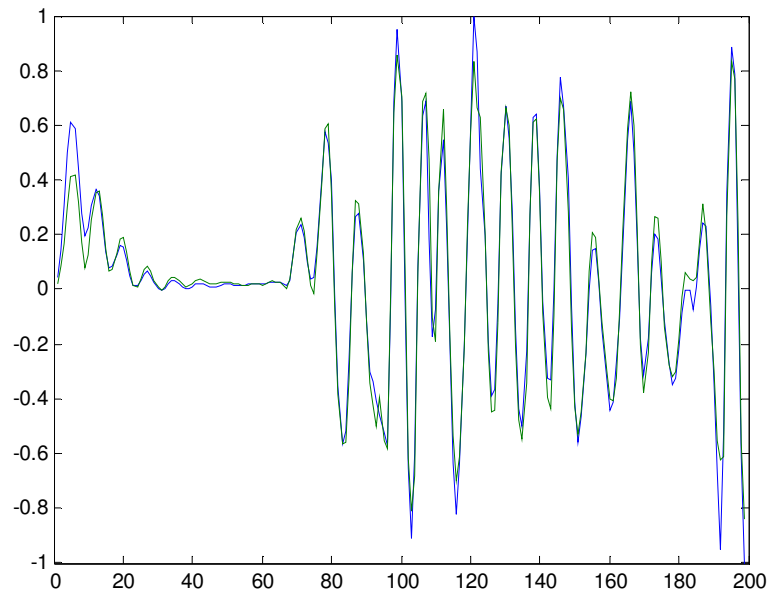


Figura 7.41 Comparación entre la función real y la estimada (variable \dot{x}) en el modelo 4 del helicóptero sin realizar el ajuste a los parámetros involucrados en la estimación

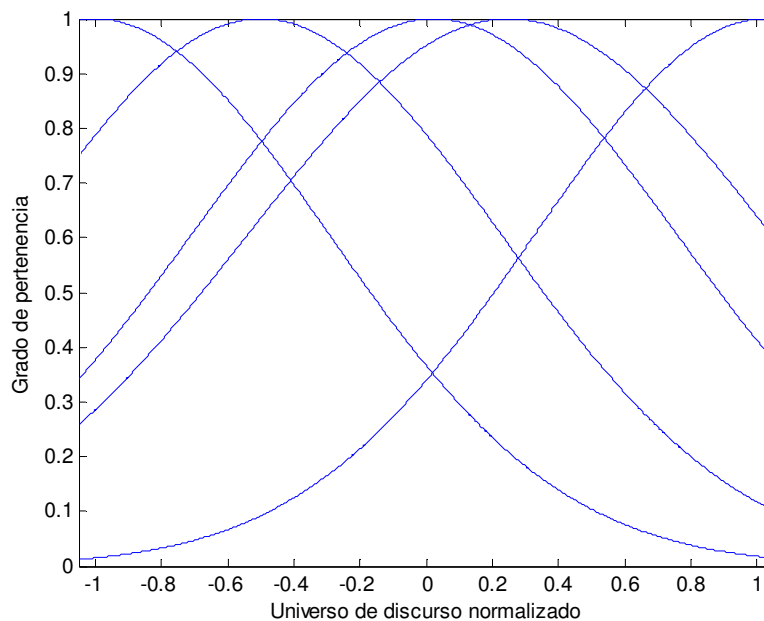


Figura 7.42 Representación de las FP para la 3ª variable de entrada (δ_θ) sin realizar ajuste a los parámetros involucrados en la estimación del modelo 4

Si realizamos el ajuste de los parámetros aplicando la razón de solape como índice para medir el solapamiento de las funciones de pertenencia en el universo de discurso, la distribución de las campanas mejora considerablemente, sin perder demasiada precisión en la estimación de las funciones. Para mostrar esta apreciable mejora en la distribución se presenta en la Figura 7.43, el resultado para la variable δ_0 en el modelo 3 (variable z en el helicóptero), tras realizar 400 iteraciones y con 5 reglas. Los parámetros utilizados en la estimación fueron: factor de aprendizaje $\eta = 0.002$, factor de aprendizaje para la razón de solape $\eta_{RS} = 0.0005$, factor de aprendizaje para los consecuentes $\eta_{RSC} = 0.001$, razón de solape deseada $RS^d = 0.4$ y factor de ancho $\psi = 2$.

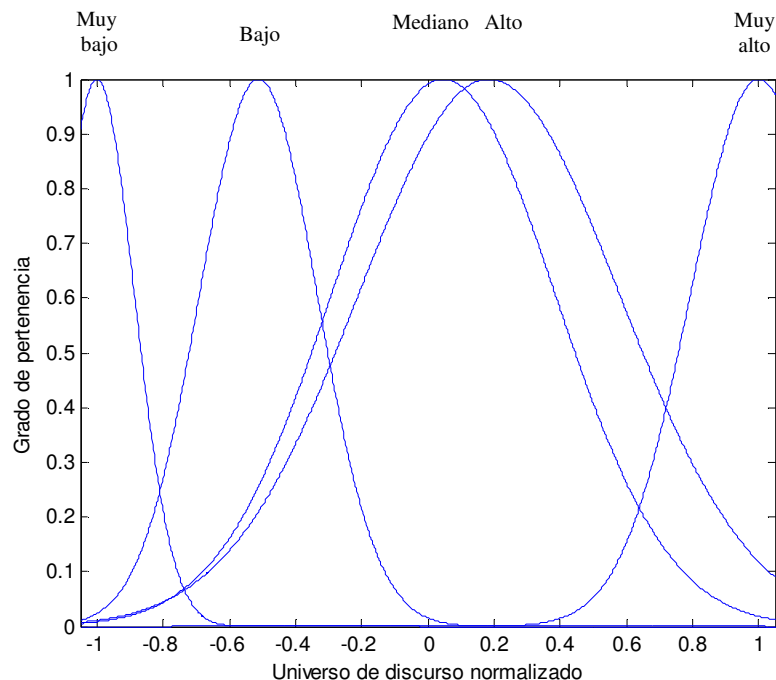


Figura 7.43 Representación de las FP para la 1ª variable de entrada (δ_0) realizando ajuste a los parámetros involucrados en la estimación del modelo 3 con la Razón de Solape

El error que se comete es de 1,12 y la comparativa entre la función real y la función estimada la podemos apreciar en la Figura 7.44.

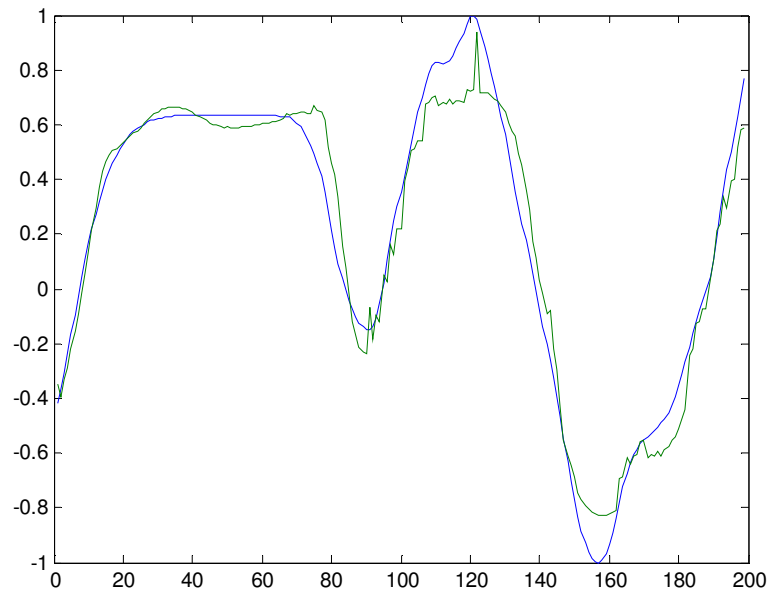


Figura 7.44 Comparación entre la función real y la función estimada (variable z) en el modelo 3 del helicóptero realizando ajuste a los parámetros involucrados en la estimación con la Razón de Solape

La mejora también es apreciable para la variable δ_θ en el modelo 4 (variable \hat{x} en el helicóptero). Con tan sólo 200 iteraciones, 5 reglas y ejecutando el algoritmo con los siguientes parámetros: factor de aprendizaje $\eta = 0.01$, factor de aprendizaje para la razón de solape $\eta_{RS} = 0.01$, factor de aprendizaje para los consecuentes $\eta_{RSC} = 0.01$, razón de solape deseada $RS^d = 0.4$ y factor de ancho $\psi = 2$, se consigue que las funciones de pertenencia queden distribuidas como podemos apreciar en la Figura 7.45.

El error cometido en esta estimación es de tan sólo 0.69, y la comparativa entre la función real y la estimada queda mostrada en la Figura 7.46.

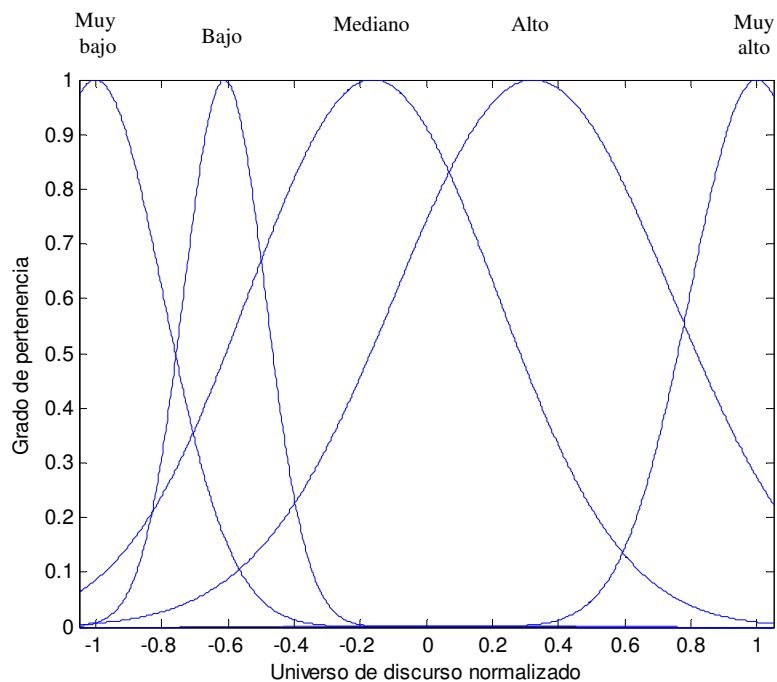


Figura 7.45 Representación de las FP para la 3ª variable de entrada (δ_θ) realizando ajuste a los parámetros involucrados en la estimación del modelo 4 con la Razón de Solape

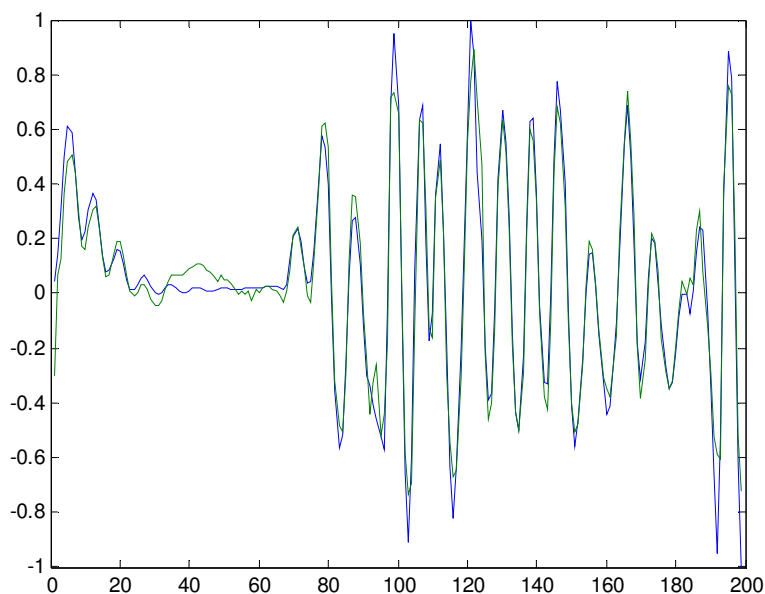


Figura 7.46 Comparación entre la función real y la función estimada (variable \dot{x}) en el modelo 4 del helicóptero realizando ajuste a los parámetros involucrados en la estimación con la Razón de Solape

Por último, se verán los resultados al aplicar como índice de solape entre las funciones de pertenencia el Área de Solape.

En primer lugar se muestra la comparativa entre las funciones real y estimada (Figura 7.47), tras 970 épocas y manteniendo las 5 reglas iniciales, para la posición z del helicóptero en el espacio (modelo 3). Se consigue un menor error (0.70) que aplicando la Razón de Solape, aunque se pierde precisión si la comparamos con la estimación realizada sin ningún tipo de ajuste. Los parámetros utilizados para realizar la estimación han sido: $\eta = 0.005$, $\eta_{AS} = 0.002$, $\eta_{ASC} = 0.002$, $AS^d = 0.25$ y $\psi = 2$.

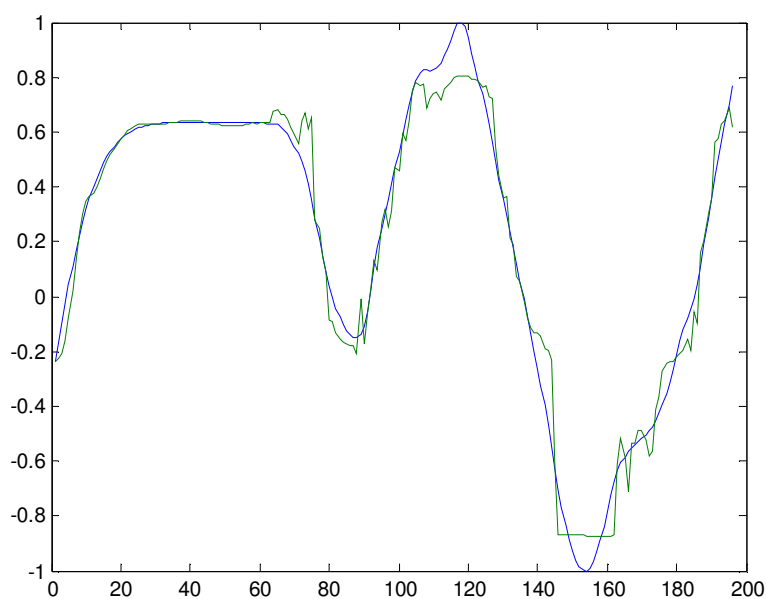


Figura 7.47 Comparación entre la función real y la función estimada (variable z) en el modelo 3 del helicóptero realizando ajuste a los parámetros involucrados en la estimación con el Área de Solape

La distribución de las campanas en el universo de discurso es muy correcta, por lo que aunque se ha perdido precisión en la estimación de la función se ha ganado una mayor interpretabilidad en el mismo. Esta distribución se puede apreciar en la Figura 7.48.

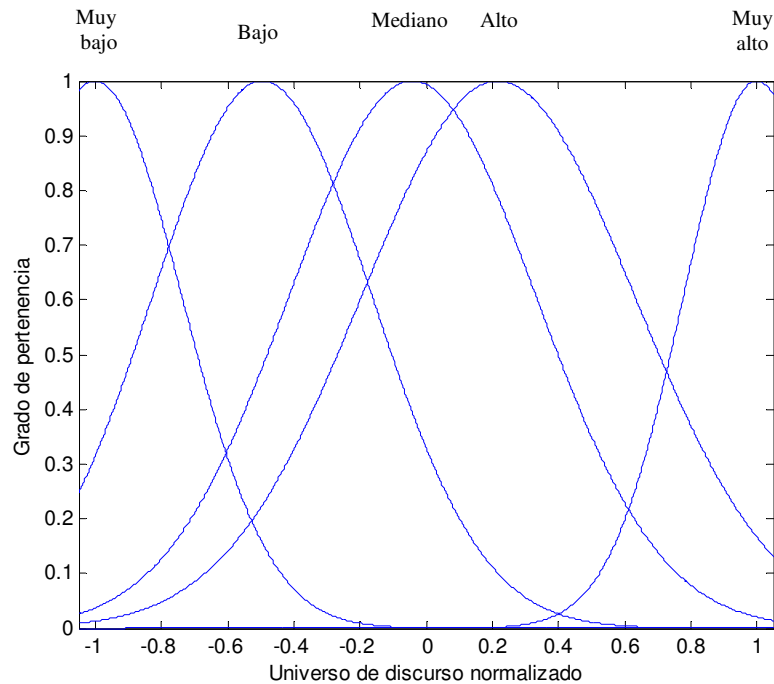


Figura 7.48 Representación de las FP para la 1ª variable de entrada (δ_0) realizando ajuste a los parámetros involucrados en la estimación del modelo 3 con el Área de Solape

A continuación se muestra la comparativa entre las funciones real y estimada (Figura 7.49), tras 825 épocas y manteniendo las 5 reglas iniciales, para la componente \dot{x} de la velocidad del helicóptero (modelo 4). En este caso se consigue un mayor error (0.74) que aplicando la Razón de Solape, y por supuesto, se pierde precisión si la comparamos con la estimación realizada sin ningún tipo de ajuste, sin embargo, la distribución de las campanas mejora de forma significativa (Figura 7.50) para la tercera variable de entrada δ_θ , lo que se traduce en una mayor interpretabilidad del modelo. El aumento del error que se produce puede ser asumido si consideramos la ganancia en la interpretabilidad. Los parámetros utilizados para realizar la estimación han sido: $\eta = 0.005$, $\eta_{AS} = 0.002$, $\eta_{ASC} = 0.002$, $AS^d = 0.25$ y $\psi = 2$.

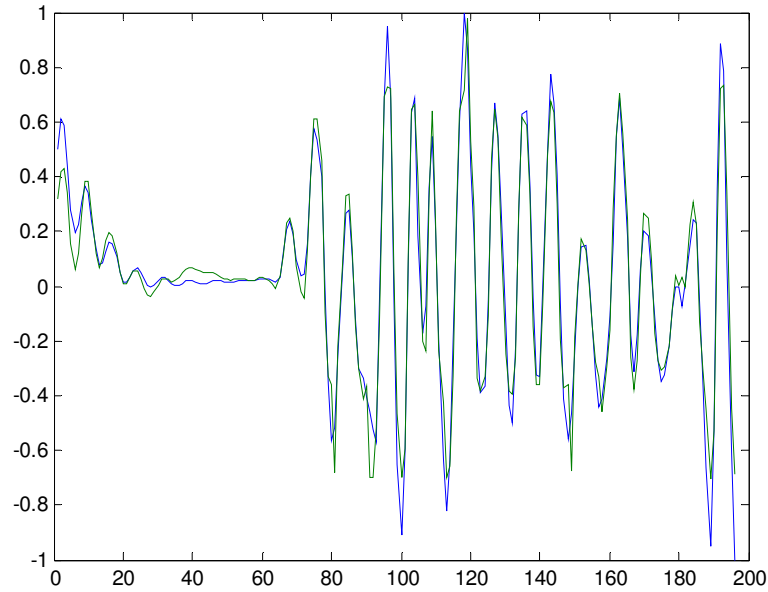


Figura 7.49 Comparación entre la función real y la función estimada (variable \dot{x}) en el modelo 4 del helicóptero realizando ajuste a los parámetros involucrados en la estimación con el Área de Solape

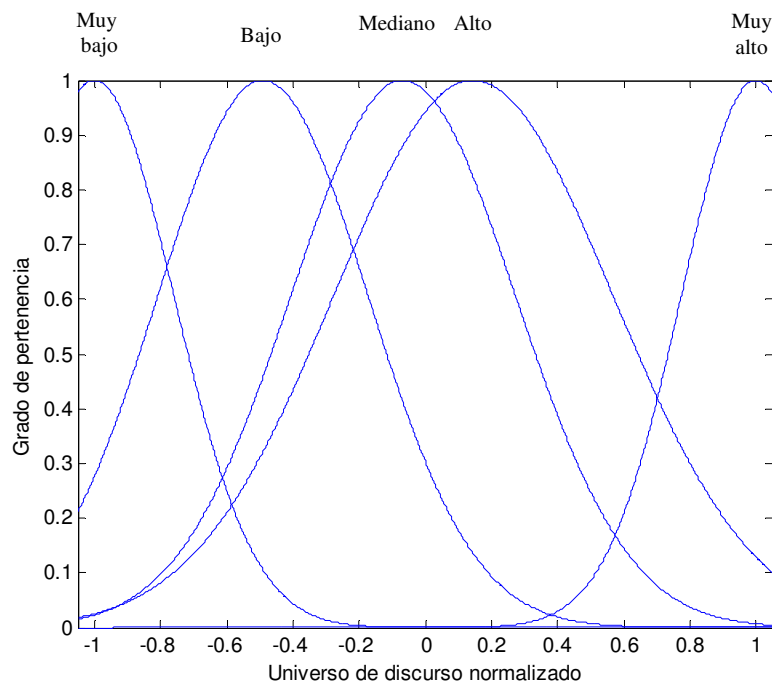


Figura 7.50 Representación de las FP para la 3ª variable de entrada (δ_θ) realizando ajuste a los parámetros involucrados en la estimación del modelo 4 con el Área de Solape

7.4. Conclusiones

En este capítulo se han abordado varios ejemplos que tratan de mostrar la validez de la metodología propuesta en esta Tesis. Se comienza con dos ejemplos sencillos de dos variables de entrada y una variable de salida. En ambos se puede apreciar como se consigue un modelo mucho más interpretable sin una considerable pérdida de precisión. Las funciones de pertenencia se distribuyen adecuadamente en el universo de discurso, al igual que los consecuentes sin un aumento significativo del error. Se consiguen, por tanto, estimar adecuadamente las funciones con una ganancia importante en la interpretabilidad del modelo.

El tercer ejemplo muestra un modelo complejo con 4 variables de entrada y 12 variables de salida. En este caso, no se consigue una distribución de las funciones tan buena pero se gana bastante en interpretabilidad, sin que la precisión se vea considerablemente afectada.

Hay que considerar que con la metodología propuesta siempre se puede apostar por una mayor precisión con la consiguiente pérdida de interpretabilidad o viceversa. Será el propio diseñador, basado en la aplicación, el que podrá elegir la combinación más adecuada de ambas.

8. Conclusiones y desarrollos futuros

Este último capítulo completa la memoria de la Tesis extrayendo las principales conclusiones derivadas de los trabajos de investigación presentados. Posteriormente se detallan las principales aportaciones y líneas de trabajo futuras que apoyándose en las propuestas presentadas se presuponen puedan surgir a partir de la Tesis.

8.1. Conclusiones

Esta Tesis presenta una metodología para obtener la estimación de un modelo mediante sistemas borrosos a partir de datos de entrada/salida, de forma que no se prime exclusivamente la precisión en la estimación del mismo sino que la interpretabilidad del resultado sea tenida muy en cuenta.

Se ha implementado un algoritmo que a medida que se van adaptando los parámetros del sistema borroso, éstos sufren un ajuste que hace que se mantengan dentro de los márgenes adecuados para mantener la interpretabilidad del resultado. Como consecuencia se obtiene la estimación del modelo con un grado de precisión adecuado sin la pérdida de su componente heurístico, interpretable por un experto.

Para mantener los parámetros dentro de estos márgenes adecuados se han utilizado dos índices que nos miden si entre funciones de pertenencia adyacentes se produce el solape deseado. Estos índices son la Razón de Solape y el Área de Solape.

El estudio se ha centrado en dos tipos de funciones de pertenencia, las triangulares (en concreto las que generan triángulos isósceles y/o equiláteros) y las gaussianas. Estas últimas presentan un inconveniente adicional: se extienden en el universo de discurso teóricamente hasta el infinito, por lo que tanto la Razón de Solape como el Área de Solape no podrían calcularse. Para solucionar este problema se hace uso del Factor de Ancho, parámetro que permite al diseñador decidir cuando considera que el grado de pertenencia de la función gaussiana está suficientemente cercano a cero. Es por este motivo que los ejemplos que se muestran están centrados en este tipo de funciones.

La metodología descrita permite no sólo distribuir adecuadamente los parámetros de la parte precedente de las reglas sino que se aplica también a los consecuentes. Cuando estos consecuentes son un singleton o de tipo TSK, el cálculo de la Razón de Solape y del Área de Solape también supone un inconveniente, por lo que se utiliza una técnica

novedosa, que permite calcular el solape entre dos valores singleton o TSK adyacentes. Para ello se convierten los consecuentes en funciones de pertenencia triangulares equivalentes, cuyos centros se corresponden con los valores de los consecuentes y cuyos anchos son los apropiados para obtener la Razón de Solape Deseada por el diseñador.

En cualquier aplicación en la que se utilice adaptación de parámetros con aprendizaje supervisado se pueden aplicar las técnicas descritas en esta Tesis, por lo que habrá muchas aplicaciones de la misma.

8.2. Desarrollos futuros

Un trabajo de investigación no debe ser algo cerrado, sino todo lo contrario, una puerta abierta a nuevas situaciones que se van encontrando durante el desarrollo de la propia investigación. En una Tesis doctoral este hecho es, si cabe, aún más notorio, ya que la extensión del trabajo realizado tanto en el tiempo como en los contenidos abordados va generando constantemente nuevas ideas que necesariamente deben ser aparcadas para su posterior estudio.

Esta sección pretende, de forma resumida, presentar algunas posibles líneas futuras de investigación que puedan surgir de este trabajo:

1. Ampliar el tipo de funciones de pertenencia que se pueden utilizar en la estimación del modelo, para no limitar el uso sólo a funciones de pertenencia triangulares y gaussianas.
2. Etiquetado automático de las funciones de pertenencia resultantes, de forma que el diseñador obtenga un modelo del sistema con la máxima interpretabilidad.
3. Mejora del algoritmo de adaptación para permitir alternancia en el orden de las funciones de pertenencia, lo que generaría reglas finales de mayor precisión e interpretabilidad.
4. Optimización del código del propio algoritmo, lo que podría incidir en un aumento considerable de la velocidad de cómputo, y por lo tanto, en una reducción del tiempo de convergencia. Esta optimización puede realizarse sustituyendo todo el proceso programado en MATLAB[®] por un código C/C++, mucho más rápido que el primero.

-
5. Limitar los solapes máximos y mínimos que pueden producirse al desplazar las FP en el ajuste. Con ello evitaríamos soluciones en las que hay un excesivo solapamiento o un solapamiento nulo.
 6. Permitir que en la estimación de un modelo puedan darse simultáneamente diferentes tipos de FP. Por ejemplo, utilizar FP diferentes en los extremos del universo de discurso.

Apéndice I.

Definición del Área de Solape en cada caso

1. La FP j tiene un área menor (por lo tanto, $\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_{j+1}}^{X_l} \mu_{F_{j+1}} + \int_{X_l}^{U_j} \mu_{F_j}}{\int_{L_j}^{U_j} \mu_{F_j}} \quad (\text{I.1})$$

2. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_{j+1}}^{X_l} \mu_{F_{j+1}} + \int_{X_l}^{U_j} \mu_{F_j}}{\int_{L_j}^{U_j} \mu_{F_{j+1}}} \quad (\text{I.2})$$

3. La FP j tiene un área menor ($\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_j}^{X_l} \mu_{F_j} + \int_{X_l}^{U_{j+1}} \mu_{F_{j+1}}}{\int_{L_j}^{U_j} \mu_{F_j}} \quad (\text{I.3})$$

4. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_j}^{X_l} \mu_{F_j} + \int_{X_l}^{U_{j+1}} \mu_{F_{j+1}}}{\int_{L_j}^{U_j} \mu_{F_{j+1}}} \quad (\text{I.4})$$

Apéndice II.

Definición del Área de Solape para funciones de pertenencia Gaussianas en cada caso

1. La FP j tiene un área menor (por lo tanto, $\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_{j+1}}^{x_l} \exp\left[-\left(\frac{x - \gamma^{j+1}}{\beta^{j+1}}\right)^2\right] + \int_{x_l}^{U_j} \exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right]}{\int_{L_j}^{U_j} \exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right]} \quad (\text{II.1})$$

2. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_j}^{x_l} \exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right] + \int_{x_l}^{U_{j+1}} \exp\left[-\left(\frac{x - \gamma^{j+1}}{\beta^{j+1}}\right)^2\right]}{\int_{L_j}^{U_j} \exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right]} \quad (\text{II.2})$$

3. La FP j tiene un área menor ($\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_{j+1}}^{x_l} \exp\left[-\left(\frac{x - \gamma^{j+1}}{\beta^{j+1}}\right)^2\right] + \int_{x_l}^{U_j} \exp\left[-\left(\frac{x - \gamma^j}{\beta^j}\right)^2\right]}{\int_{L_{j+1}}^{U_{j+1}} \exp\left[-\left(\frac{x - \gamma^{j+1}}{\beta^{j+1}}\right)^2\right]} \quad (\text{II.3})$$

4. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).

$$AS(j) = \frac{\int_{L_j}^{x_l} \exp\left[-\left(\frac{x-\gamma^j}{\beta^j}\right)^2\right] + \int_{x_l}^{U_{j+1}} \exp\left[-\left(\frac{x-\gamma^{j+1}}{\beta^{j+1}}\right)^2\right]}{\int_{L_{j+1}}^{U_{j+1}} \exp\left[-\left(\frac{x-\gamma^{j+1}}{\beta^{j+1}}\right)^2\right]} \quad (\text{II.4})$$

Apéndice III.

Adaptación de cada parámetro implicado en el ajuste para funciones de pertenencia Gaussianas en cada caso

1. La FP j tiene un área menor (por lo tanto, $\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).

$$\gamma_i^j(k) = \gamma_i^j(k) - \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_j} \quad (\text{III.1})$$

$$\gamma_i^{j+1}(k+1) = \gamma_i^{j+1}(k) + \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_j} \quad (\text{III.2})$$

$$\beta_i^j(k+1) = \beta_i^j(k) + \eta_{AS} (AS - AS^d) \left(\frac{\beta_{j+1} \text{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_j^2 \text{erf}(\psi)} + \frac{\beta_{j+1}}{2\beta_j^2} + \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \text{erf}(\psi) \sqrt{\pi} \beta_j} \right) \quad (\text{III.3})$$

$$\beta_i^{j+1}(k+1) = \beta_i^{j+1}(k) - \eta_{AS} (AS - AS^d) \left(\frac{\text{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_j \text{erf}(\psi)} + \frac{1}{2\beta_j} - \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \text{erf}(\psi) \sqrt{\pi} \beta_j} \right) \quad (\text{III.4})$$

donde $\text{erf}(x)$ es un número cuyo valor se calcula según (6.14)

2. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la izquierda en el universo de discurso ($\gamma^j \leq \gamma^{j+1}$).

$$\gamma_i^j(k+1) = \gamma_i^j(k) + \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_j} \quad (\text{III.5})$$

$$\gamma_i^{j+1}(k+1) = \gamma_i^{j+1}(k) - \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_j} \quad (\text{III.6})$$

$$\beta_i^j(k+1) = \beta_i^j(k) - \eta_{AS} (AS - AS^d) \left(\frac{\beta_{j+1} \text{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_j^2 \text{erf}(\psi)} - \frac{\beta_{j+1}}{2\beta_j^2} + \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \text{erf}(\psi) \sqrt{\pi} \beta_j} \right) \quad (\text{III.7})$$

$$\beta_i^{j+1}(k+1) = \beta_i^{j+1}(k) + \eta_{AS} (AS - AS^d) \left(\frac{\text{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_j \text{erf}(\psi)} - \frac{1}{2\beta_j} + \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \text{erf}(\psi) \sqrt{\pi} \beta_j} \right) \quad (\text{III.8})$$

donde $\text{erf}(x)$ es un número cuyo valor se calcula según (6.14)

3. La FP j tiene un área menor ($\beta^j \leq \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).

$$\gamma_i^j(k+1) = \gamma_i^j(k) - \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \quad (\text{III.9})$$

$$\gamma_i^{j+1}(k+1) = \gamma_i^{j+1}(k) + \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\text{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \quad (\text{III.10})$$

$$\beta_i^j(k+1) = \beta_i^j(k) - \eta_{AS} (AS - AS^d) \left(\frac{\operatorname{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_{j+1}\operatorname{erf}(\psi)} + \frac{1}{2\beta_{j+1}} - \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \operatorname{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \right) \quad (\text{III.11})$$

$$\beta_i^{j+1}(k+1) = \beta_i^{j+1}(k) + \eta_{AS} (AS - AS^d) \left(\frac{\beta_j \operatorname{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_{j+1}^2 \operatorname{erf}(\psi)} + \frac{\beta_j}{2\beta_{j+1}^2} + \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \operatorname{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \right) \quad (\text{III.12})$$

donde $\operatorname{erf}(x)$ es un número cuyo valor se calcula según (6.14)

4. La FP $j+1$ tiene un área menor ($\beta^j > \beta^{j+1}$) y la FP j se encuentra más a la derecha en el universo de discurso ($\gamma^j > \gamma^{j+1}$).

$$\gamma_i^j(k+1) = \gamma_i^j(k) + \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\operatorname{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \quad (\text{III.13})$$

$$\gamma_i^{j+1}(k+1) = \gamma_i^{j+1}(k) - \eta_{AS} (AS - AS^d) \frac{\exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{\operatorname{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \quad (\text{III.14})$$

$$\beta_i^j(k+1) = \beta_i^j(k) + \eta_{AS} (AS - AS^d) \left(\frac{\operatorname{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_{j+1}\operatorname{erf}(\psi)} - \frac{1}{2\beta_{j+1}} + \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \operatorname{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \right) \quad (\text{III.15})$$

$$\beta_i^{j+1}(k+1) = \beta_i^{j+1}(k) - \eta_{AS} (AS - AS^d) \left(\frac{\beta_j \operatorname{erf}\left(\frac{\gamma_j - \gamma_{j+1}}{\beta_j + \beta_{j+1}}\right)}{2\beta_{j+1}^2 \operatorname{erf}(\psi)} - \frac{\beta_j}{2\beta_{j+1}^2} + \frac{(\gamma_j - \gamma_{j+1}) \exp\left(-\frac{(\gamma_j - \gamma_{j+1})^2}{(\beta_j + \beta_{j+1})^2}\right)}{(\beta_j + \beta_{j+1}) \operatorname{erf}(\psi) \sqrt{\pi} \beta_{j+1}} \right) \quad (\text{III.16})$$

donde $\operatorname{erf}(x)$ es un número cuyo valor se calcula según (6.14)

Bibliografía

- [1] Acosta, J., Nebot, A., Villar, P. and Fuertes, J.M. *Automatic learning of fuzzy partitions in human central nervous system modeling using genetic algorithms*. International Conference on Health Science Simulation, pp. 90-95. SCS International, 2004.
- [2] Al-Hadithi, B.M., Matía, F. and Jiménez, A. *Slide-mode fuzzy controller for highly non linear systems*. The 10th IEEE International Conference on Fuzzy Systems, pp. 1515-1518, Melbourne, Australia, 2001.
- [3] Al-Hadithi, B.M., Matía, F. and Jiménez, A. *Análisis de estabilidad de sistemas borrosos*. Revista Iberoamericana de Automática e Informática Industrial (RIAI), 4(2):7-25, 2007.
- [4] Al-Hadithi, B.M., Matía, F. and Jiménez, A. *Fuzzy controller for robot manipulators*. Proceedings of the 12th international Fuzzy Systems Association world congress on Foundations of Fuzzy Logic and Soft Computing, pp. 688-697, Cancún, México, 2007.
- [5] Albertos, P. and Sala, A. *Perspectives of fuzzy control: lights and shadows*. 2nd International IEEE Conference 'Intelligent Systems', Vol. 1, pp. 25-32, 2004.
- [6] Alsina, C., Trillas, E. and Valverde, L.J. *On some logical connectives for fuzzy sets theory*. Journal of Mathematical Analysis and Applications, 93(1):15-26, 1983.
- [7] Alsina, C. and Trillas, E. *On uniformly close fuzzy preorders*. Fuzzy Sets and Systems, 53(3):343-346, 1993.
- [8] Alsina, C. and Trillas, E. *On the law $S(S(x,y),T(x,y)) = S(x,y)$ of fuzzy logic*. Fuzzy Optimization and Decision Making, 6(2):99-107, 2007.
- [9] Alsinet, T., Chesñevar, C., Godo, L., Sandri, S. and Simari, G. *Formalizing argumentative reasoning in a possibilistic logic programming setting with fuzzy unification*. International Journal of Approximate Reasoning, 2008.
- [10] Alsinet, T., Chesñevar, C., Godo, L. and Simari, G. *A logic programming framework for possibilistic argumentation: Formalization and logical properties*. Fuzzy Sets and Systems, 159(10):1208-1228, 2008.
- [11] Andújar, J.M. and Barragán, A.J. *A methodology to design stable nonlinear fuzzy control systems*. Fuzzy Sets and Systems, 154(2):157-181, 2005.
- [12] Andújar, J.M., Barragán, A.J., Córdoba, J.M. and Fernández, I. *Diseño de sistemas de control borroso: Modelado de la planta*. Revista Iberoamericana de Automática e Informática Industrial (RIAI), 3(1):75-81, 2006.
- [13] Andújar, J.M., Barragán, A.J. and Gegúndez, M.E. *A general and formal methodology for designing stable nonlinear fuzzy control systems*. IEEE Transactions on Fuzzy Systems, 17(5):1081-1091, 2009.

-
- [14] Andújar, J.M., Barragán, A.J., Gegúndez, M.E. and Maestre, M. *Control borroso multivariable basado en heurística. Un caso práctico: grúa porta contenedores*. Revista Iberoamericana de Automática e Informática Industrial (RIAI), 4(2):81-89, 2007.
- [15] Aracil, J., Ollero, A. and García-Cerezo, A. *Stability indices for the global analysis of expert control systems*. IEEE Transactions on Systems, Man, and Cybernetics, 19(5):998-1007, 1989.
- [16] Aracil, J. and Gordillo, F. *Describing function method for stability analysis of PD and PI fuzzy controllers*. Fuzzy Sets and Systems, 143(2):233-249, 2004.
- [17] Aroba, J., Grande, J.A., Andújar, J.M., De la Torre, M.L. and Riquelme, J.C. *Application of fuzzy logic and data mining techniques as tools for qualitative interpretation of acid mine drainage processes*. Environmental Geology, 53(1):135-145, 2007.
- [18] Arrue, B.C., Ollero, A. and Martínez de Dios, J.R. *An intelligent system for false alarm reduction in infrared forest-fire detection*. IEEE Intelligent Systems and Their Applications, 15(3):64-73, 2000.
- [19] Bai, R., Chai, T. and Ma, E. *A Novel approach for extraction of fuzzy rules using the neuro-fuzzy network and its application in the blending process of raw slurry*. Lecture Notes In Computer Science, 4492:362-370, 2007.
- [20] Barcelo, A., Montseny, E. and Sobrevilla, P. *On fuzzy texture spectrum for natural microtextures characterization*. Conference EUSFLAT'05, Barcelona, Spain, 2005.
- [21] Barro, S., Marín, R., Palacios, F. and Ruíz, R. *Fuzzy logic in a patient supervision system*. Artificial Intelligence in Medicine, 21(1-3):193-199, 2001.
- [22] Baumann, B.M., Washington, G., Glenn, B.C. and Rizzoni, G. *Mechatronic design and control of hybrid electric vehicles*. IEEE/ASME Transactions on Mechatronics, 5(1):58-72, 2000.
- [23] Birkhoff, G. *Lattice Theory*. American Mathematical Society, 2nd edition, 1948.
- [24] Bondia, J. and Picó, J. *Analysis of linear systems with fuzzy parametric uncertainty*. Fuzzy Sets and Systems, 135(1):81-121, 2003.
- [25] Bondia, J. and Picó, J. *Application of functional intervals to the response evaluation of linear time-invariant systems with fuzzy input*. Reliable Computing, 10(5):369-387, 2004.
- [26] Bondia, J., Sala, A., Picó, J. and Sainz, M.A. *Controller design under fuzzy pole-placement specifications: An interval arithmetic approach*. IEEE Transactions on Fuzzy Systems, 14(6):822-836, 2006.

-
- [27] Bou, F., Esteva, F. and Godo, L. *Modal systems based on many-valued logics*. New dimensions in fuzzy logic and related technologies. EUSFLAT, Vol. 1, pp. 177-182, 2007.
- [28] Box, G.E.P., Jenkins, G.M. *Time Series Analysis: Forecasting and Control*. Holden Day, 1989
- [29] Brown, M. and Harris, C. J. *Neurofuzzy Adaptive Modelling and Control*. Prentice Hall, 1994.
- [30] Buckles, B.P. and Petry, F.E. *Genetic Algorithms*. IEEE Computer Society Press, 1992.
- [31] Cárdenas, J.M., Marín, M.A. and Navarrete, I. *Fuzzy temporal constraint logic: a valid resolution principle*. Fuzzy Sets and Systems, 117(2):231-250, 2001.
- [32] Carmona, P., Castro, J.L. and Zurita, J.M. *Strategies to identify fuzzy rules directly from certainty degrees: A comparison and a proposal*. IEEE Transactions on Fuzzy Systems, 12(5):631-640, 2004.
- [33] Castro, J.L. *Fuzzy logics as families of bivaluated logics*. Fuzzy Sets and Systems, 64(3):321-332, 1994.
- [34] Castro, J.L. *Fuzzy logic controllers are universal approximators*. IEEE Transactions on Systems, Man, and Cybernetics, 25(4):629-635, 1995.
- [35] Castro, J.L., Castro-Schez, J.J. and Zurita, J.M. *Learning maximal structure rules in fuzzy logic for knowledge acquisition in expert systems*. Fuzzy Sets and Systems, 101(3):331-342, 1999.
- [36] Castro, J.L. and Delgado, M. *Fuzzy systems with defuzzification are universal approximators*. IEEE Transactions on Systems, Man, and Cybernetics–Part B: Cybernetics, 26(1):149-152, 1996.
- [37] Castro, J.L., Mantas, C.J. and Benítez, J.M. *Interpretation of artificial neural networks by means of fuzzy rules*. IEEE Transactions on Neural Networks, 13(1):101-116, 2002.
- [38] Castro, J.L. and Trillas, E. *The management of the inconsistency in expert systems*. Fuzzy Sets and Systems, 58(1):51-57, 1993.
- [39] Castro, J.L., Trillas, E. and Zurita, J.M. *Non-monotonic fuzzy reasoning*. Fuzzy Sets and Systems, 94(2):217-225, 1998.
- [40] Castro, J.L. and Zurita, J.M. *And inductive learning algorithm in fuzzy systems*. Fuzzy Sets and Systems, 89(2):193-203, 1997.
- [41] Cherry, A.S. and Jones, R.P. *Fuzzy logic control of an automotive suspension system*. IEE Proceedings: Control Theory and Applications, pp. 149-160. IEE, Stevenage, United Kingdom, 1995.

-
- [42] Chiclana, F., Herrera, F. and Herrera-Viedma, E. *Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations*. Fuzzy Sets and Systems, 97(1):33-48, 1998.
- [43] Contreras, J., Misa, R. y Murillo, L.F. *Obtención de modelos borrosos interpretables de Procesos Dinámicos*. Revista Iberoamericana de Automática e Informática Industrial (RIAI), 5(3):70-77, 2008.
- [44] Cerdón, O. *An evolutionary Methodology for Designing Fuzzy Rule-Based Systems*. PhD thesis, University of Granada, 1997.
- [45] Cerdón, O., Gomide, F., Herrera, F., Hoffmann, F. and Magdalena, L. *Ten years of genetic fuzzy Systems: Current Framework and new trends*. Fuzzy Sets and Systems, 141(1):5-31, 2004.
- [46] Cuesta, F., Gordillo, F., Aracil, J. and Ollero, A. *Stability analysis of nonlinear multivariable Takagi-Sugeno fuzzy control systems*. IEEE Transactions on Fuzzy Systems, 7(5):508-520, 1999.
- [47] Cuesta, F., Ollero, A., Arrue, B.C. and Braunstingl, R. *Intelligent control of nonholonomic mobile robots with fuzzy perception*. Fuzzy Sets and Systems, 134(1):47-64, 2003.
- [48] Cuesta, F., Ponce, E. and Aracil, J. *Local and global bifurcations in simple Takagi-Sugeno fuzzy systems*. IEEE Transactions on Fuzzy Systems, 9(2):355-368, 2001.
- [49] De la Torre, M.L., Grande, J.A., Aroba, J. and Andújar, J.M. *Optimization of fertirrigation efficiency in strawberry crops by application of fuzzy logic techniques*. Journal of Environmental Monitoring, 7(11):1085-1092, 2005.
- [50] De Miguel, L.J., Mediavilla, M. and Perán, J.R. *Fault diagnosis system based on sensitivity analysis and fuzzy logic*. 26th International Symposium on Multiple-Valued Logic (ISMVL'96), pp. 50, Santiago de Compostela, Spain, 1996. IEEE Computer Society.
- [51] Delgado, M., Herrera, F., Herrera-Viedma, E. and Martínez, L. *Combining numerical and linguistic information in group decision making*. Information Sciences, 107(1-4):177-194, 1998.
- [52] Delgado, M., Marín, N., Sánchez, D. and Vila, M.A. *Fuzzy association rules: General model and applications*. IEEE Transactions on Fuzzy Systems, 11(2):214-225, 2003.
- [53] Delgado, M., Sánchez, D., Martín-Bautista, M.J. and Vila, M.A. *A probabilistic definition of a nonconvex fuzzy cardinality*. Fuzzy Sets and Systems, 126(2):177-190, 2002.
- [54] Delgado, M., Sánchez, D. and Vila, M.A. *Fuzzy cardinality based evaluation of quantified sentences*. International Journal of Approximate Reasoning, 23(1):23-66, 2000.

-
- [55] Delgado, M., Verdegay, J.L. and Vila, M.A. *Linguistic decision-making models*. International Journal of Intelligent Systems, 7(5):479-492, 1992.
- [56] Delgado, M., Verdegay, J.L. and Vila, M.A. *On aggregation operations of linguistic labels*. International Journal of Intelligent Systems, 8(3):351-370, 1993.
- [57] Delgado, M., Verdegay, J.L. and Vila, M.A. *Model for linguistic partial information in decision-making problems*. International Journal of Intelligent Systems, 9(4):365-378, 1994.
- [58] Díez, J.L., Navarro, J.L. and Sala, A. *A fuzzy clustering algorithm enhancing local model interpretability*. Soft Computing, 11(10):973-983, 2007.
- [59] Ding, Y., Ying, H. and Shao, S.H. *Necessary conditions on minimal system configuration for general MISO Mamdani fuzzy systems as universal approximators*. IEEE Transactions on Systems, Man, and Cybernetics–Part B: Cybernetics, 30(6):857-864, 2000.
- [60] Dormido, S., Santos, M., de Madrid, A.P. and Morilla, F. *Autosintonía de controladores borrosos utilizando técnicas clásicas basadas en reguladores PID*. Comunicaciones del III FLAT, pp. 217-225, Universidad de Santiago de Compostela, España, 1993.
- [61] Dubois, D., Esteva, F., Godo, L. and Prade, H. *Fuzzy-set Based Logics. An History-oriented Presentation of their Main Developments*, Vol. 8, pp.325-449. Elsevier, 2007.
- [62] Esteva, F., Gispert, J., Godo, L. and Noguera, C. *Adding truth-constants to logics of continuous T-norms: axiomatization and completeness results*. Fuzzy Sets and Systems, 158(6):597-618, 2007.
- [63] Farrall, S.D. and Jones, R.P. *Energy management in a automotive electric/heat engine hybrid powertrain using fuzzy decision making*. Proceedings of the 1993 IEEE International Symposium on Intelligent Control, pp. 463-468, Chicago, IL, USA, 1993.
- [64] Félix, P., Barro, S. and Marín, R. *Fuzzy constraint networks for signal pattern recognition*. Artificial Intelligence, 148(1-2):103-140, 2003.
- [65] Félix, P., Fraga, S., Barro, S. and Marín, R. *Linguistic representation of fuzzy temporal profiles*. International Journal Uncertainly Fuzziness Knowledge-Based Systems, 7(3):243-256, 1999.
- [66] Félix, P., Fraga, S., Marín, R. and Barro, S. *Trend detection based on a fuzzy temporal profile model*. Artificial Intelligence in Engineering, 13(4):341-349, 1999.
- [67] Filev, D.P. and Yager, R.R. *An adaptive approach to defuzzification based on level sets*. Fuzzy Sets and Systems, 54(3):355-360, 1993.
- [68] Flaminio, T. and Godo, L. *A logic for reasoning about the probability of fuzzy events*. Fuzzy Sets and Systems, 158(6):625-638, 2007.

-
- [69] Freeman, J.A. and Skapura, D.M. *Neural Networks: Algorithms, Applications and Programming Techniques*. Addison Wesley Longman Publishing Co., CA, USA, 1991.
- [70] García-Cerezo, A., Mandow, A. and López-Baldán, M.J. *Fuzzy modelling operator navigation behaviors*. IEEE International Conference on Fuzzy Systems, Vol. 3, pp. 1339-1345, Piscataway, NJ, USA, 1997.
- [71] García-Cerezo, A. and Ollero, A. *Design of stable fuzzy control systems from experimental data*. EUFIT94, pp. 1175-1182, 1994.
- [72] García-Cerezo, A., Ollero, A. and Aracil, J. *Dynamic analysis of weighed-output fuzzy control Systems*. Annual Review in Automatic Programming, 19(C):43-48, 1994.
- [73] Godo, L. and Rodriguez, R.O. *Logical approaches to fuzzy similarity-based reasoning: an overview*. Springer-Verlag, 2008.
- [74] Godo, L. and Torra, V. *On aggregation operators for ordinal qualitative information*. IEEE Transactions of Fuzzy Systems, 8(2):143-154, 2000.
- [75] Gómez-Bravo, F., Cuesta, F. and Ollero, A. *Parallel and diagonal parking in nonholonomic autonomous vehicles*. Engineering Applications of Artificial Intelligence, 14(4):419-434, 2001.
- [76] Gómez-Skarmeta, A.F., Valdés, M., Jiménez, F. and Marín-Blázquez, J.G. *Approximative fuzzy rules approaches for classification with hybrid-GA techniques*. Information Sciences, 136(1-4):193-214, 2001.
- [77] Gordillo, F., Aracil, J. and Alamo, T. *Determining limit cycles in fuzzy control systems*. IEEE International Conference on Fuzzy Systems, Vol. 1, pp. 193-198, 1997.
- [78] Grande, J.A., Andújar, J.M., Aroba, J., De la Torre, M.L. and Beltrán, R. *Precipitation, pH and metal load in AMD river basins: An application of fuzzy clustering algorithms to the process characterization*. Journal of Environmental Monitoring, 7(4):325-334, 2005.
- [79] Guillaume, S. *Designing Fuzzy Inference Systems from Data: An Interpretability-Oriented Review*. IEEE Transactions on Fuzzy Systems, 9(3):426-442, 2001.
- [80] Herrera, F. and Herrera-Viedma, E. *Aggregation operators for linguistic weighted information*. IEEE Transactions on Systems, Man, and Cybernetics–Part B: Cybernetics, 27(5):646-656, 1997.
- [81] Herrera, F., Herrera-Viedma, E. and Chiclana, F. *Multiperson decision-making based on multiplicative preference relations*. European Journal of Operational Research, 129(2):372-385, 2001.
- [82] Herrera, F., Herrera-Viedma, E. and Martínez, L. *A fusion approach for managing multi-granularity linguistic term sets in decision making*. Fuzzy Sets and Systems, 114(1):43-58, 2000.

-
- [83] Herrera, F., Herrera-Viedma, E. and Verdegay, J.L. *Sequential selection process in group decision making with a linguistic assessment approach*. Information Sciences, 85(4):223-239, 1995.
- [84] Herrera, F., Herrera-Viedma, E. and Verdegay, J.L. *Direct approach processes in group decision making using linguistic OWA operators*. Fuzzy Sets and Systems, 79(2):175-190, 1996.
- [85] Herrera, F., Herrera-Viedma, E. and Verdegay, J.L. *A model of consensus in group decision making under linguistic assessments*. Fuzzy Sets and Systems, 78(1):73-87, 1996.
- [86] Herrera, F., Herrera-Viedma, E. and Verdegay, J.L. *A rational consensus in group decision making using linguistic assessments*. Fuzzy Sets and Systems, 88(1):31-49, 1997.
- [87] Herrera, F., Lozano, M. and Verdegay, J.L. *A learning process for fuzzy control rules using genetic algorithms*. Fuzzy Sets and Systems, 100(1-3):143-158, 1998.
- [88] Herrera, F. and Martínez, L. *A 2-tuple fuzzy linguistic representation model for computing with words*. IEEE Transactions on Fuzzy Systems, 8(6):746-752, 2000.
- [89] Hirota, K. *Industrial education on fuzzy systems in Japan*. Proceedings of the Industrial Electronics, Control, and Instrumentation, Vol. 1, pp. 176-181, Maui, HI, USA, 1993.
- [90] Holland, J.H. *Adaptation in natural and artificial systems*. University of Michigan Press, 1975.
- [91] Jang, J.S.R. *ANFIS: Adaptive-Network-Based Fuzzy Inference Systems*. IEEE Transactions on Systems, Man, and Cybernetics, 23(3): 665-685, 1993.
- [92] Jiménez, F., Cadenas, J.M., Verdegay, J.L. and Sánchez, G. *Solving fuzzy optimization problems by evolutionary algorithms*. Information Sciences, 152(1):303-311, 2003.
- [93] Jiménez, F., Cadenas, J.M., Sánchez, G., Gómez-Skarmeta, A.F. and Verdegay, J.L. *Multi-objective evolutionary computation and fuzzy optimization*. International Journal of Approximate Reasoning, 43(1):59-75, 2006.
- [94] Johnson, W. *Helicopter theory*. Dover Publications, Inc., 1980.
- [95] Juez, J., Sainz, G.I., Moya, E.J. and Perán, J.R. *Early detection and diagnosis of faults in an AC motor using neurofuzzy techniques: FasArt + fuzzy K nearest neighbors*. Lecture Notes in Computer Science, 2085:571-578, 2001.
- [96] Ketata, R., Bellaaj, H., Chtourou, M. and Ben Amer, M. *Adjustment of membership functions, generation and reduction of fuzzy rule base from numerical data*. Malaysian Journal of Computer Science, 20(2):147-169, 2007.

-
- [97] Kim, H.M., Dickerson, J. and Kosko, B. *Fuzzy throttle and brake control for platoons of smart cars*. Fuzzy Sets and Systems, 84(3):209-234, 1996.
- [98] Kim, S.K., Tilbury, D.M.: *Mathematical Modeling and Experimental Identification of an Unmanned Helicopter Robot with Flybar*. Dynamics. Journal of Robotics Systems. Vol 21, no. 3, pp. 95-116, 2004.
- [99] King, P.J. and Mamdani, E.H. *The application of fuzzy control systems to industrial processes*. Automatica, 13(3):235-242, 1977.
- [100] Klement, E., Mesiar, R. and Pap, E. *Triangular norms. Position paper I: basic analytical and algebraic properties*. Fuzzy Sets and Systems, 143(1):5-26, 2004.
- [101] Klement, E., Mesiar, R. and Pap, E. *Triangular norms. Position paper II: general constructions and parameterized families*. Fuzzy Sets and Systems, 145(3):411-438, 2004.
- [102] Klement, E., Mesiar, R. and Pap, E. *Triangular norms. Position paper III: continuous T-norms*. Fuzzy Sets and Systems, 145(3):439-454, 2004.
- [103] Klir, G.J. and Yuan, B. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall International, New Jersey, 1995.
- [104] Kosko, B. *Fuzzy engineering*. Prentice-Hall, Inc., 1997.
- [105] Kosko, B. *Fuzzy systems as universal approximators*. IEEE Transactions on Computers, 43(11):1329-1333, 1994.
- [106] Kosko, B. *Neural networks and fuzzy systems: a dynamical systems approach to machine intelligence*. Prentice-Hall, Inc., 1992.
- [107] Kouatli, I. and Jones, B. *An Improved Design Procedure for Fuzzy Control Systems*. International Journal of Machine Tools and Manufacture. Vol. 31, no. 1, pp. 107-122, 1991.
- [108] Koza, J.R. *Genetic Programming. On the Programming of Computers by Means of Natural Selection*. The MIT Press, 1972.
- [109] Kreinovich, V., Mouzouris, C.G. and Nguyen, H.T. *Fuzzy rule based modeling as a universal approximation tool*. Fuzzy Systems: Modeling and Control. Kluwer Academic, 1998.
- [110] Kreinovich, V., Nguyen, H.T. and Yam, Y. *Fuzzy systems are universal approximators for a smooth function and its derivatives*. International Journal of Intelligent Systems, 15(6):565-574, 2000.
- [111] Layne, J.R., Passino, K.M. and Yurkovich, S. *Fuzzy learning control for antiskid braking systems*. IEEE Transactions on Control Systems Technology, 1(2):122-129, 1993.

-
- [112] Lee, H.D. and Sul, S.K. *Fuzzy-logic-based torque control strategy for parallel-type hybrid electric vehicle*. IEEE Transactions on Industrial Electronics, 45(4):625-632, 1998.
- [113] Lee, Y. and Zak, S.H. *Designing a genetic neural fuzzy antilock-brake-system controller*. IEEE Transactions on Evolutionary Computation, 6(2):198-211, 2002.
- [114] Lennon, W.K. and Passino, K.M. *Intelligent control for brake systems*. IEEE Transactions on Control Systems Technology, 7(2):188-202, 1999.
- [115] Li, H.X. and Chen, C.P. *The equivalence between fuzzy logic systems and feedforward neural networks*. IEEE Transaction on Neural Networks. 11(2):356-365, 2000
- [116] Liu, P. *Mamdani fuzzy system: universal approximator to a class of random processes*. IEEE Transactions on Fuzzy Systems, 10(6):756-766, 2002.
- [117] Liu, X. *Parameterized defuzzification with maximum entropy weighting function-another view of the weighting function expectation method*. Mathematical and Computer Modelling, 45(1-2):177-188, 2007.
- [118] Ljung, L. and Glad, T. *On global identifiability for arbitrary model parametrizations*. Automatica, 30(2):265-276, 1994.
- [119] López, M.J. *Contribuciones al modelado y control de sistemas dinámicos mediante lógica borrosa*. Tesis doctoral, Universidad de Málaga, 1996.
- [120] López-Baldán, M.J., García-Cerezo, A., Cejudo, J.M. and Romero, A. *Fuzzy modeling of a termal solar plant*. International Journal of Intelligent Systems, 17(4):369-379, 2002.
- [121] López-Baldán, M.J., Ruiz-Gómez, J., Fernández, R. and García-Cerezo, A. *Input-output fuzzy modelling applied to a mobile robot*. Computational Intelligence and Applications, pp. 283-288, 1999.
- [122] Lowe, D. *Adaptive radial basis function nonlinearities, and the problem of generalisation*. Institution of electrical engineers. International conference on artificial neural networks, 1989.
- [123] Lukasiewicz, J. *O logice trojwartosciowej*. Ruch Filozoficzny, 5: 170-171, 1920.
- [124] Mabuchi, S. *A proposal for a defuzzification strategy by the concept of sensitivity analysis*. Fuzzy Sets and Systems, 55(1):1-14, 1993.
- [125] Madau, D.P., Yuan, F., Davis, L.I. and Jr. Feldkamp, L.A. *Fuzzy logic anti-lock brake system for a limited range coefficient of friction surface*. Second IEEE International Conference on Fuzzy Systems, pp. 883-888. IEEE, Piscataway, NJ, USA, 1993.
- [126] Mamdani, E.H. *Application of fuzzy algorithms for control of a simple dynamic plant*. Proceedings of the IEEE, 121(12):1585-1588, 1974.

-
- [127] Mamdani, E.H. *Application of fuzzy logic to approximate reasoning using linguistic synthesis*. IEEE Transactions on Computers, C-26(12):1182-1191, 1977.
- [128] Mamdani, E.H. and Assilian, S. *An experimental in linguistic synthesis with a fuzzy logic controller*. International Journal of Man-Machine Studies, 7(1):1-13, 1975.
- [129] Marsh, S., et. al.: *Fuzzy Logic Education Program*. Center for Emerging Computer Technologies. Motorola, 1994.
- [130] Martínez-De Dios, J.R. and Ollero, A. *A multiresolution-fuzzy method for robust threshold selection in image segmentation*. Intelligent Automation and Soft Computing, 12(4):419-430, 2006.
- [131] Matía, F., Al-Hadithi, B.M. and Jiménez, A. *On the global stability of Takagi-Sugeno general model*. Mathware and Soft Computing, 6(2-3):293-304, 1999.
- [132] Matía, F., Al-Hadithi, B.M. and Jiménez, A. *Generalization of stability criterion for Takagi-Sugeno continuous fuzzy model*. Fuzzy Sets and Systems, 129(3):295-309, 2002.
- [133] Matía, F., Jiménez, A., Al-Hadithi, B.M., Rodríguez-Losada, D. and Galán, R. *The fuzzy Kalman filter: State estimation using possibilistic techniques*. Fuzzy Sets and Systems, 157(16):2145-2170, 2006.
- [134] Mauer, G.F. *Fuzzy logic controller for an ABS braking system*. IEEE Transactions on Fuzzy Systems, 3(4):381-388, 1995.
- [135] Mencattini, A. Salmeri, M. and Salsano, A. *Sufficient conditions to impose derivative constraints on MISO Takagi-Sugeno fuzzy logic systems*. IEEE Transactions on Fuzzy Systems, 13(4):454-467, 2005.
- [136] Mendel, J.M. *Fuzzy logic systems for engineering: A tutorial*. Proceedings of the IEEE, 83(3):345-377, 1995.
- [137] Menger K. *Statistical metric spaces*. National Academy of Sciences, Vol. 37, pp. 535-537, USA, 1942.
- [138] Miller, G.A. *The magical number seven or minus two: some limits on our capacity of processing information*. Psychological Rev., 63:81-97, 1956.
- [139] Mitra, S. and Pal, S. *Fuzzy sets in pattern recognition and machine intelligence*. Fuzzy Sets and Systems, 156(3): 381-386, 2005.
- [140] Mizumoto, M. *Fuzzy controls under various fuzzy reasoning methods*. Information Sciences, 45(2):129-151, 1988.
- [141] Mizumoto, M. and Zimmermann, H.J. *Comparison of fuzzy reasoning methods*. Fuzzy Sets and Systems, 8(3):253-283, 1982.

-
- [142] Montseny, E., Sobrevilla, P. and Mares, P. *Edge orientation-based fuzzy hough transform (EOFHT)*. 3rd International Conference on Fuzzy Logic and Technology. Prof. Rainer Hampel, pp. 683-688, 2003.
- [143] Moraga, C., Trillas, E. and Guadarrama, S. *Multiple-valued logic and artificial intelligence fundamentals of fuzzy control revisited*. Artificial Intelligence Review, 20(3-4):169-197, 2003.
- [144] Murray, R., Li, Z. and Sastry, S. *A mathematical introduction to robotic manipulation*. CRC Press, 1994.
- [145] Narendra, K.S. and Parthasarathy, K. *Identification and Control of Dynamical Systems Using Neural Networks*. IEEE Trans. Neural Networks. Vol. 1, no. 1, pp. 4-27, 1990.
- [146] Nguyen, D. and Widrow, B. *Neural Network for self learning control systems*. IEEE Control Systems Magazine. Pp. 18-23, 1990.
- [147] Nomura, H., Hayashi, I. and Wakami, N. *A self-tuning method of fuzzy inference rules by descent method*. In: Lowen, R. and Roubens, M (Eds.), Fuzzy Logic, Kluwer Academic Publishers, Dordrecht, pp. 465-475, 1993.
- [148] Norgaard, M., Ravn, O., Poulsen, N. and Hansen, L. *Neural Networks for Modelling and Control of Dynamic Systems*. Springer-Verlag, 2000.
- [149] Ollero, A., Aracil, J. and García-Cerezo, A. *Robust desing of rule-based fuzzy controllers*. Fuzzy Sets and Systems, 70(2-3):249-273, 1995.
- [150] Ollero, A., Aracil, J. and Gordillo, F. *Stability analysis of MIMO fuzzy control systems in the frequency domain*. IEEE International Conference on Fuzzy Systems, Vol. 1, pp. 49-54, Anchorage, AK, USA, 1998.
- [151] Ollero, A., García-Cerezo, A. and Martinez, J. *Fuzzy supervisory path tracking of mobile robots*. Control Engineering Practice, 2(2):313-319, 1994.
- [152] Padfield, G. *Helicopter flight dynamics*. Blackwell Science Ltd, 1996.
- [153] Paiva, R.P., and Dourado, A. *Interpretability and learning in neuro-fuzzy systems*. Fuzzy Sets and Systems. 147(1):17-38, 2004.
- [154] Palm, R., Driankov, D. and Hellendoorn, H. *Model based fuzzy control*. Springer, 1997.
- [155] Palma, J., Juarez, J.M., Campos, M. and Marín, R. *Fuzzy theory approach for temporal model-based diagnosis: An application to medical domains*. Artificial Intelligence in Medicine, 38(2):197-218, 2006.
- [156] Perfilieva, I. and Kreinovich, V. *A new universal approximation result for fuzzy systems, which reflects CNF-DNF duality*. International Journal of Intelligent Systems, 17(12):1121-1130, 2002.

-
- [157] Pradera, A., Trillas, E. and Calvo, T. *A general class of triangular norm-based aggregation operators: Quasi-linear T-S operators*. International Journal of Approximate Reasoning, 30(1):57-72, 2002.
- [158] Rao, M.V.C. and Prahlad, V. *A tunable fuzzy logic controller for vehicle-active suspension systems*. Fuzzy Sets and Systems, 85(1):11-21, 1997.
- [159] Rovatti, R. *Fuzzy piecewise multilinear and piecewise linear systems as universal approximators in Sobolev norms*. IEEE Transactions on Fuzzy Systems, 6(2):235-249, 1998.
- [160] Ruiz-Gómez, J., López-Baldán, M.J. and García-Cerezo, A. *Fuzzy modelling of a ternary batch distillation column*. System and Control: Theory and Applications, pp. 320-324, 2000.
- [161] Sainz, G.I., Fuente, M.J. and Vega, P. *Recurrent neuro-fuzzy modeling of a wastewater treatment plant*. European Journal of Control, 10(1):84-96, 2004.
- [162] Sainz, G.I., Santamaria, J.J. and Dimitriadis, Y.A. *Performance of a recurrent neuro-fuzzy ART based system for pattern recognition and modelling of dynamical systems: RFasArt*. IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th, 1(1):534-539, 2001.
- [163] Sainz, G.I., Dimitriadis, Y.A., Cano, J.M., Gómez, E. and Parrado, E. *ART based model set for pattern recognition: FasArt family*. Neuro-Fuzzy pattern recognition, World Scientific Co., pp. 147-177, 2000.
- [164] Sala, A. and Albertos, P. *Fuzzy systems evaluation: the inference error approach*. IEEE Transactions on Systems, Man and Cybernetics–Part B: Cybernetics, 28(2):268-275, 1998.
- [165] Sala, A. and Albertos, P. *Inference error minimisation: fuzzy modelling of ambiguous functions*. Fuzzy Sets and Systems, 121(1):95-111, 2001.
- [166] Sala, A., Díez, J.L., Navarro, J.L. and Albertos, P. *Fuzzy model usage and readability in identification for control*. Intelligent Automation and Control Trends, Principles, and Applications – Preceedings of the Sixth Biannual World Automation Congress, WAC, pp. 113-118, Seville, Spain, 2004.
- [167] Salcedo, J.V. and Martínez, M.A. *BIBO stabilisation of Takagi-Sugeno fuzzy systems under persistent perturbations using fuzzy output-feedback controllers*. IET Control Theory and Applications, 2(6):513-523, 2008.
- [168] Salcedo, J.V. and Martínez, M.A. *Design of PDC fuzzy controllers under persistent disturbances and application in mechanical systems*. Advances in Engineering Software, 39(11):937-946, 2008.
- [169] Salcedo, J.V., Martínez, M.A. and García-Nieto, S. *Stabilization conditions of fuzzy systems under persistent perturbations and their application in nonlinear systems*. Engineering Applications of Artificial Intelligence, 21(8):1264-1276, 2008.

-
- [170] Sánchez, O., Ollero, A. and Heredia, G. *Adaptive Fuzzy Control for Automatic Path Tracking of Outdoor Mobile Robots. Application to ROMEO 3R*. IEEE International Conference on Fuzzy Systems. Barcelona, 1997.
- [171] Santos, M., de la Cruz, J.M. and Dormido, S. *Self-tuning of fuzzy logic controllers in cascade loops*. Intelligent Systems and Soft Computing for Nuclear Science and Industry, pp. 258-264, 1996.
- [172] Santos, M., Dormido, S. and de la Cruz, J.M. *Evaluation of the fuzzy controllers performance*. 4th European Congress on Intelligent Techniques and Soft Computing EUFIT'96, Vol. 2, pp. 1056-1060, Verlag Mainz, Aachen, 1996.
- [173] Santos, M., Dormido, S. and de la Cruz, J.M. *Fuzzy-PID controllers vs. Fuzzy-PI controllers*. 5th IEEE International Conference on Fuzzy Systems FUZZ-IEEE'96, pp. 1598-1604, 1996.
- [174] Santos, M., Dormido, S. and de la Cruz, J.M. *Influence of the information processing in fuzzy logic controllers*. 6th International Conference Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU'96, Vol. 1, pp. 81-86, 1996. Proyecto Sur de ediciones.
- [175] Santos, M., Dormido, S., de la Cruz, J.M. and López, J.A. *Tuning of fuzzy controllers: Applications of the relay method*. EUROSIM, Vol. 3, pp. 1131-1136, New Orleans, 1995.
- [176] Santos, M., Dormido, S., de Madrid, A.P. and de la Cruz, J.M. *Between fuzzy-PID and PID-conventional controllers: a good choice*. Biennial Conference of the North American Fuzzy Information Processing Society (NAFIPS), pp. 123-127, 1996.
- [177] Santos, M., Dormido, S., de Madrid, A.P., Morilla, F. and de la Cruz, J.M. *Tuning fuzzy logic controllers by classical techniques*. CASR'94: Selected papers from the 4th International Workshop on Computer Aided System Theory, pp. 214-224, London, UK, 1996. Springer-Verlag.
- [178] Schouten, N.J., Salman, M.A. and Kheir, N.A. *Fuzzy logic control for parallel hybrid vehicles*. IEEE Transactions on Control Systems Technology, 10(3):460-468, 2002.
- [179] Schouten, N.J., Salman, M.A. and Kheir, N.A. *Energy management strategies for parallel hybrid vehicles using fuzzy logic*. Control Engineering Practice, 11(2):171-177, 2003.
- [180] Schweizer B. and Skalar A. *Probabilistic metric spaces*. North Holland, Amsterdam, 1983.
- [181] Shao, J.Q., Lee, Y.C. and Jones, R.D. *Generation of fuzzy rules by discretizing and pruning the weights of a radial basis function network*. LANL Unclassified Report, LA-UR-93-3379, 1993.
- [182] Shi, Y. and Mizumoto, M. *A new approach of neuro-fuzzy learning algorithm for tuning fuzzy rules*. Fuzzy Sets and Systems, 112(1):99-116, 2000.

-
- [183] Sugeno, M. and Kang, G.T. *Structure identification of fuzzy model*. Fuzzy Sets and Systems, 28(1):15-33, 1988.
- [184] Takagi, T. and Sugeno, M. *Fuzzy identification of systems and its applications to modeling and control*. IEEE Transactions on Systems, Man, and Cybernetics, 15(1):116-132, 1985.
- [185] Taskin, Y., Hacıoglu, Y. and Yagiz, N. *The use of fuzzy-logic control to improve the ride comfort of vehicles*. Journal of Mechanical Engineering, 53(4):233-240, 2007.
- [186] Togai, M. and Watanabe, H. *Expert system on a chip: an engine for real-time approximate reasoning*. Proceedings of the ACM SIGART International symposium on Methodologies for intelligent systems, pp. 147-154, New York, NY, USA, 1986. ACM Press.
- [187] Torra, V. and Narukawa, Y. *Fuzzy measures and integrals in evaluation of strategies*. Information Sciences, 177(21):4686-4695, 2007.
- [188] Trillas, E. *Sobre funciones de negación en la teoría de conjuntos difusos*. Stochastica, 3:47-60, 1979.
- [189] Trillas, E. *Conjuntos borrosos*. Vicens-Vives, 1980.
- [190] Trillas, E. and Alsina, C. *Some remarks on approximate entailment*. International Journal of Approximate Reasoning, 6(4):525-533, 1992.
- [191] Trillas, E., Delgado, M., Vila, M.A., Castro, J.L., Verdegay, J.L., Moraga, C., Cuenca, J., Gutierrez, J. and Ruiz, A. *Fundamentos e introducción a la ingeniería "Fuzzy"*. Omron Electronics S.A., 1994.
- [192] Trillas, E., Alsina, C. and Terricabras, J.M. *Introducción a la lógica borrosa*. Ariel, Barcelona, 1995.
- [193] Trillas, E. and Cubillo, S. *Primeras lecciones sobre lógica borrosa*. Facultad de Informática, Servicio de Publicaciones, 1999.
- [194] Tsukamoto, Y. *An approach to fuzzy reasoning model*. Advances in Fuzzy Set Theory and Applications. Amsterdam, 1979.
- [195] Turksen, I.B. and Tian, Y. *Combination of rules or their consequences in fuzzy expert systems*. Fuzzy Sets and Systems, 58(1):3-40, 1993.
- [196] Vachtsevanos, G.J., Farinwata, S.S. and Pirovolou, D.K. *Fuzzy logic control of an automotive engine*. IEEE Control Systems Magazine, 13(3):62-68, 1993.
- [197] Vélez, M.A., Sánchez, O. and Ollero, A. *Neuro-Fuzzy modelling with parameters adjustment based on overlap ratio*. 10th Mediterranean Conference on Control and Automation. Lisboa, 2002.

-
- [198] Vetterlein, T. and Navara, M. *Defuzzification using Steiner points*. Fuzzy Sets and Systems, 157(11):1455-1462, 2006.
- [199] Von Altrock, C. *Fuzzy logic technologies in automotive engineering*. Embedded systems conference, Vol. 2, pp. 407-422, 1994.
- [200] Von Altrock, C., Krause, B. and Zimmermann, H.J. *Advanced fuzzy logic control of a model car in extreme situations*. Fuzzy Sets and Systems, 48(1):41-52, 1992.
- [201] Von Altrock, C., Krause, B. and Zimmermann, H.J. *Advanced fuzzy logic control technologies in automotive applications*. IEEE Conference on Fuzzy Systems, pp. 831-842, 1992.
- [202] Wang, L.X. *Adaptive fuzzy system and control*. Prentice Hall, 1994.
- [203] Wang, L.X. *A course in fuzzy systems and control*. Prentice Hall, 1996.
- [204] Wang, L.X. *Fuzzy systems are universal approximators*. IEEE International Conference on Fuzzy Systems, pp. 1163-1170, San Diego, CA, USA, 1992.
- [205] Wang, L., Langari, R. and Yen, J. *Identifying fuzzy rule-based models using orthogonal transformation and backpropagation*. Fuzzy Theory Systems: Techniques and applications. Vol. 1, pp. 188-202, 1999.
- [206] Won, J.S. and Langari, R. *Fuzzy torque distribution control for a parallel hybrid vehicle*. Expert Systems, 19(1):4-10, 2002.
- [207] Yager, R.R. and Filev, D.P. *On the issue of defuzzification and selection based on a fuzzy set*. Fuzzy Sets and Systems, 55(3):255-271, 1993.
- [208] Yager, R.R. and Filev, D.P. *Essential of Fuzzy Modeling and Control*. John Wiley & Sons. 1994.
- [209] Yasunobu, S. and Miyamoto, S. *Automatic train operation by predictive fuzzy control*. Industrial Applications of Fuzzy Control, pp. 1-18, Amsterdam, 1985.
- [210] Yen, J. and Langari, R. *Fuzzy Logic: Intelligence, control, and information*. Prentice Hall, 1999.
- [211] Ying, H. *Sufficient conditions on uniform approximation of multivariate functions by general Takagi-Sugeno fuzzy systems with linear rule consequent*. IEEE Transactions on Systems, Man, and Cybernetics–Part A: Systems and Humans, 28(4):515-520, 1998.
- [212] Ying, H., Ding, Y., Li, S. and Shao, S. *Typical Takagi-Sugeno and Mamdani fuzzy systems as universal approximators: necessary conditions and comparison*. IEEE International Conference on Fuzzy Systems, Vol. 1, pp. 824-828, 1998.
- [213] Zadeh, Lofti A. *Fuzzy sets*. Information and Control, 8(3):338-353, 1965.

-
- [214] Zadeh, Lofti A. *Outline of a new approach to the analysis of complex systems and decision processes*. IEEE Transactions on Systems, Man, and Cybernetics, 3:28-44, 1973.
- [215] Zadeh, Lofti A. *Semantic inference from fuzzy premises*. In Proceedings of the 6th International Symposium on Multiple-valued logic, pp. 217-218, Los Alamitos, CA, USA, 1976. IEEE Computer Society Press.
- [216] Zadeh, Lofti A. *Similarity relations and fuzzy orderings*. Information Sciences, 3(2):177-200, 1971.
- [217] Zadeh, Lofti A. *The concept of a linguistic variable and its application to approximate reasoning – I*. Information Sciences, 8(3):199-249, 1975.
- [218] Zadeh, Lofti A. *The concept of a linguistic variable and its application to approximate reasoning – II*. Information Sciences, 8(4):301-357, 1975.
- [219] Zadeh, Lofti A. *The concept of a linguistic variable and its application to approximate reasoning – III*. Information Sciences, 9(1):43-80, 1975.
- [220] Zeng, K., Xu, W. and Zhang N. *Universal approximation of special Mamdani fuzzy systems*. Control and Decision, 15(4):435-438, 2000.
- [221] Zeng, K. Zhang, N. and Xu, W. *A comparative study on sufficient conditions for Takagi-Sugeno fuzzy systems as universal approximators*. IEEE Transactions on Fuzzy Systems, 8(6):773-780, 2000.