

UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS
FACULTAD DE INGENIERÍA ELÉCTRICA
DEPARTAMENTO DE AUTOMÁTICA Y SISTEMAS COMPUTACIONALES



TRABAJO DE DIPLOMA

Título: Identificación de modelos difusos.

Autor: Erik Marichal Arbona.

Tutor: M.Sc. Boris Luis Martínez Jiménez.

Santa Clara

2004

UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS
FACULTAD DE INGENIERÍA ELÉCTRICA
DEPARTAMENTO DE AUTOMÁTICA Y SISTEMAS COMPUTACIONALES



TRABAJO DE DIPLOMA

Título: Identificación de modelos difusos.

Autor: Erik Marichal Arbona.

E-mail: emarichal@uclv.edu.cu

Tutor: M.Sc. Boris Luis Martínez Jiménez.

Profesor Asistente. Dpto. de Automática y Sist. Computacionales

Facultad de Ingeniería Eléctrica. UCLV.

E-mail: boris@fie.uclv.edu.cu

Santa Clara

2004



Hago constar que el presente trabajo de diploma fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de estudios de la especialidad de Ingeniería en Automática autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

Firma del Autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del Autor

Firma del Jefe de Departamento
donde se defiende el trabajo

Firma del Responsable de
Información Científico-Técnica

A mis padres.
A mi hermana.
A Mima.
A Mamita.

Quisiera agradecer a todas aquellas personas que hicieron posible la realización de este trabajo.

Principalmente a mi tutor, que sin el hubiera sido imposible lograrlo.

A la profesora María del Carmen por su ayuda incondicional durante estos cinco años.

A todos los profesores que me transmitieron sus conocimientos y experiencias profesionales

A los buenos amigos que siempre estuvieron presente hasta en los momentos más difíciles, sin dejar de mencionar, a aquellos que fueron cómplices de inolvidables momentos, a Melvin, Ariel y Héctor.

A mis compañeros de Aula con los que compartí gran parte de este tiempo.

Y un agradecimiento especial:

A mis padres y hermana por tanta consagración.

A Mima, a Mamita y Eddi por el constante apoyo que me dieron.

A mi prima, casi hermana, Eilen y a su novio Amílcar.

A mi queridísima tía Margot.

A Cachi y a Peña.

A Tomasa.

A Mauro y a Oro.

A primos, tíos y abuelos, ... en fin a mi numerosa y unida familia.

A todos gracias.

TAREA TÉCNICA

Para la realización de este trabajo, se desarrollaron las siguientes tareas:

1. Revisión bibliográfica sobre modelado difuso y técnicas de clusterización.
2. Análisis, desarrollo y programación de algoritmos de reducción de reglas y de técnicas de determinación del número de clusters.
3. Integrar técnicas implementadas anteriormente y dar como resultado un modelo difuso optimizado.
4. Simulación de los algoritmos desarrollados. Ajuste y evaluación sobre varios sistemas.
5. Análisis de los resultados obtenidos.
6. Elaboración del informe final.

Firma del Autor

Firma del tutor

RESUMEN

En este trabajo se realiza un estudio sobre la identificación de modelos difusos del tipo Takagi-Sugeno aplicando diferentes algoritmos de clusterización, tanto difusa como clásica, debido a la posibilidad que brinda la combinación de ambas técnicas de obtener modelos compactos y que, por tanto, disminuyen notablemente los gastos computacionales. Se hace una pequeña exposición de las principales características de los algoritmos de clusterización que han sido objeto de estudio en otros trabajos y que aplicaremos en este.

Se tratan también los fundamentos teóricos sobre algunos criterios de medidas de validez que permiten encontrar el número de clusters óptimo para lograr una exitosa identificación y de un algoritmo de reducción de la base de reglas necesario para eliminar las redundancias en dicha base de reglas difusas. Estos resultados son insertados dentro del procedimiento de identificación que aquí se utiliza.

Además, es obtenida una biblioteca de funciones en el *software* Matlab con la implementación de los algoritmos antes mencionados y otras funciones indispensables para lograr la modelación difusa, que finalmente fue validada aplicandola a tres ejemplos tomados de la literatura, con vista a valorar los resultados del procedimiento de identificación difusa que se propone.

TABLA DE CONTENIDOS

INTRODUCCIÓN1

 Organización de la tesis 3

CAPÍTULO 1. MODELACIÓN DIFUSA Y CLUSTERIZACIÓN.....5

 1.1 Definición de sistema de inferencia difuso (SID)..... 5

 1.2 Modelos difusos 6

 1.2.1 Etapas de la modelación difusa..... 6

 1.2.2 Modelo Takagi-Sugeno..... 7

 1.2.3 Mecanismo de inferencia 8

 1.2.4 Modelo simplificado Takagi-Sugeno..... 9

 1.3 Clusterización 9

 1.3.1 Algoritmos de clusterización difusa 10

 1.3.2 Algoritmos de clusterización clásica 12

 1.4 Conclusiones 14

CAPÍTULO 2. OBTENCIÓN DE MODELOS DIFUSOS.....15

 2.1 Descripción del proceso de identificación 15

 2.2 Medidas de validez 18

 2.3 Proyección de clusters 19

 2.3.1 Proyección en la clusterización clásica..... 19

2.3.2 Proyección en la clusterización difusa.....	20
2.4 Simplificación de la base de reglas.....	21
2.5 Programación e Implementación.....	24
2.6 Conclusiones.....	29
CAPÍTULO 3. ANÁLISIS DE RESULTADOS.....	30
3.1 Identificación de un sistema dinámico no lineal.....	30
3.2 Identificación del proceso de combustión en un horno.....	33
3.3 Identificación del proceso de neutralización del pH.....	34
3.4 Conclusiones.....	37
CONCLUSIONES.....	38
RECOMENDACIONES.....	40
REFERENCIAS BIBLIOGRÁFICAS.....	41
ANEXOS.....	44

INTRODUCCIÓN

En la identificación de sistemas complejos con un grado de alinealidad de mediano a alto, muchas veces se hace engorroso aplicar las técnicas tradicionales de modelación. Una buena solución para resolver estos problemas es recurrir a los métodos de identificación inteligente y específicamente, la modelación difusa puesto que los sistemas difusos se caracterizan por la rapidez y suavidad en la respuesta comparados con los sistemas convencionales. Esto significa más eficiencia y operaciones adecuadas para las tareas a cumplir (Martínez, 2000).

Entre diferentes tipos de sistemas difusos existentes, los modelos Takagi-Sugeno requieren de un pequeño número de reglas para describir procesos altamente no lineales, el cual puede ser significativamente más pequeño comparados con los casos en que se aplican otros tipos de modelos difusos. Además, los modelos difusos Takagi-Sugeno son sencillos y computacionalmente eficientes. Debido a estas características, este trabajo se centra en la obtención de sistema difuso de este tipo, para la modelación e identificación de sistemas no lineales.

La recogida y procesamiento de datos es un aspecto de gran importancia para llevar a cabo la identificación de sistemas. Para lograr un adecuado tratamiento de los datos y facilitar su posterior uso, es aplicada la técnica de clusterización ya que permite agrupar los datos en clusters o grupos donde sus características no varíen de forma significativa. La aplicación de estas técnicas en la modelación difusa ha sido tratado por otros autores (Abonyi, 2003; Babuska, 1998; Martínez y Herrera, 2003) y en nuestro departamento se han realizado algunas investigaciones precedentes relacionadas con los métodos de clusterización, tanto difusa (Alfonso, 2002) como clásica (Fernández, 2003), donde se implementaron algunos algoritmos de extracción de clusters. Precisamente estas investigaciones fueron la génesis

de nuestra tesis pues se necesitaba añadir a sus resultados varias funciones que completaran el proceso de obtención de modelos difusos. Por tanto, constituyen objetivos de este trabajo la integración, en una biblioteca de funciones de Matlab, de las técnicas de clusterización implementadas anteriormente con otros algoritmos de obtención del número de clusters y de reducción de la base de reglas, con vistas a dar como resultado final un modelo difuso compacto. También se deberá aplicar el procedimiento de identificación aquí asumido a varios procesos no lineales para validar los resultados.

Durante el desarrollo del trabajo, uno de los problemas a resolver lo constituyó encontrar el número de cluster óptimo para el cual se obtiene un modelo difuso sencillo y con un buen comportamiento. Para lidiar con esto se utilizan los criterios de medidas de validez, aunque debemos aclarar que solo son aplicables cuando se emplea la clusterización difusa.

Una vez que se tienen los clusters y sus prototipos es necesario realizar sus proyecciones para obtener los antecedentes de las reglas y definir los parámetros de las funciones de pertenencia difusas en el espacio n -dimensional de las entradas. Los parámetros de las funciones lineales de los consecuentes se ajustan por mínimos cuadrados de Gauss.

El proceso de proyección generalmente da lugar a redundancias. La existencia de estas redundancias en las base de reglas aumentan la complejidad y el gasto computacional de los modelos innecesariamente. Con vista a eliminarlas, sin que el modelo presente variaciones considerables en sus características, se implementa un algoritmo de reducción de base de reglas.

Se empleó Matlab, por ser una herramienta potente de cálculo y de simulación, para implementar varias funciones, con el propósito de cumplir con todos los requisitos antes mencionados del proceso de identificación y de esta manera lograr los objetivos propuestos: la generación de modelos difusos compactos y eficientes. Un resultado muy importante alcanzado fue la creación de una biblioteca de funciones que, a partir de los datos de entrada-salida de un proceso determinado, devuelve el modelo difuso sin redundancias y cumpliendo las exigencias del usuario

Para la realización de este trabajo, se realizaron las siguientes tareas:

1. Revisión bibliográfica sobre modelado difuso y técnicas de clusterización

2. Análisis, desarrollo y programación de algoritmos de reducción de reglas y de técnicas de determinación del número de clusters.
3. Integrar técnicas implementadas anteriormente y dar como resultado un modelo difuso optimizado.
4. Simulación de los algoritmos desarrollados. Ajuste y evaluación sobre varios sistemas.
5. Análisis de los resultados obtenidos.
6. Elaboración del informe final.

Organización de la tesis

Este informe está conformado por tres capítulos. El Capítulo 1 (Modelación difusa y clusterización) aborda aspectos relacionados con los modelos difusos y los principios básicos de los modelos tipo Takagi-Sugeno. Además, en el se hace un estudio general realizando comparaciones entre los principales algoritmos de clusterización difusa y clásica que serán aplicados en el procedimiento de identificación para la obtención de modelos difusos.

En el Capítulo 2 (Identificación de modelos difusos) se explican cada uno de los pasos a seguir en el procedimiento de identificación, y los fundamentos teóricos sobre algunos criterios de medidas de validez. Son analizadas las proyecciones de los clusters obtenidos, tanto por clusterización difusa como clásica. Se hace un estudio sobre el algoritmo de reducción de la base de reglas que se emplea en el trabajo y es dada una explicación general de cada una de las funciones que han sido implementadas.

En el Capítulo 3 (Análisis de resultados) se aplican a tres ejemplos clásicos tomados de literatura científica internacional, los desarrollos teóricos y las funciones implementadas en Matlab que fueron objeto de estudio en los capítulos 1 y 2. Se prueban los criterios de medidas de validez utilizando un algoritmo de clusterización difusa. Se establecen comparaciones basadas en el criterio de error cuadrático entre los modelos difusos obtenidos en este trabajo y modelos difusos obtenidos por otros autores, y por último son comparadas las características de un modelo difuso antes y después de eliminar las redundancias de la base de reglas.

Finalmente, se expresan las conclusiones a las que se arriban y son dadas algunas recomendaciones para trabajos futuros.

CAPÍTULO 1. MODELACIÓN DIFUSA Y CLUSTERIZACIÓN

Los modelos difusos se basan en la codificación difusa de la información. Operan con conjuntos difusos en vez de con números. Permitiendo cierta imprecisión en el modelo, se pueden formular situaciones complejas incluyendo la inherente vaguedad del proceso humano al conceptualizar el mundo exterior. En muchos problemas reales, esta imprecisión es admisible e incluso útil ya que en esencia, la representación de la información en sistemas difusos imita los mecanismos de razonamiento aproximado de la mente humana.

En este capítulo se abordaran aspectos relacionados con los modelos difusos, los principios básico de los modelos del tipo Takagi-Sugeno, y se hará un estudio sobre los algoritmos de clusterización, difusos y clásicos, que serán utilizados más adelante para la obtención de dichos modelos.

1.1 Definición de sistema de inferencia difuso (SID)

Un sistema de inferencia difuso (SID) es un sistema computacional basado en los conceptos de la teoría de conjuntos difusos, reglas *If-Then* difusas y razonamiento difuso. Se conocen por diversos nombres como Sistemas difusos basados en reglas, Sistemas expertos difusos, etc.

La estructura básica de un SID consiste de tres componentes: una base de reglas difusa, una base de datos (la cual contiene las funciones de pertenencia usadas y sus parámetros) y un mecanismo de razonamiento que ejecuta el procedimiento de inferencia.

Las entradas y salidas pueden ser valores ordinarios o difusos. Cuando la salida es difusa y se necesita el valor ordinario, se emplea un método de defusificación el cual determina el valor ordinario que mejor representa un conjunto difuso.

1.2 Modelos difusos

Los modelos difusos se pueden clasificar en tres categorías: los lingüísticos (Mamdani), los de ecuaciones relacionales y los del tipo Takagi-Sugeno. En los modelos lingüísticos el antecedente y el consecuente son conjuntos difusos, mientras en el modelo Takagi-Sugeno el antecedente está formado por conjuntos difusos, pero el consecuente está compuesto de funciones lineales. Los modelos de funciones relacionales difusas tienen como objetivos formar matrices relacionales difusas según los datos de entrada-salida del proceso.

Los sistemas difusos, en general, son funciones matemáticas flexibles que pueden aproximar funciones o datos (mediciones) con la precisión deseada. Comparada con otras técnicas modernas de aproximación de funciones como las redes neuronales artificiales, los sistemas difusos brindan una representación más transparente del sistema bajo estudio, debido fundamentalmente a la posibilidad de interpretación lingüística en forma de reglas. La estructura lógica de las reglas facilita el entendimiento y el análisis del modelo de forma semi-cualitativa, similar a como los humanos razonamos.

1.2.1 Etapas de la modelación difusa

La modelación difusa se ejecuta en dos etapas. En la primera se identifica la estructura superficial del sistema y en la segunda la estructura profunda (Bello, 2004).

Primera etapa:

- Seleccionar las variables de entrada y salida relevantes.
- Seleccionar un tipo específico de SID.
- Determinar la cantidad de términos lingüísticos asociados con cada variable de entrada y de salida (para el modelo de Takagi-Sugeno determinar el grado de la función consecuente).
- Definir el conjunto de reglas difusas.

Segunda etapa:

- Seleccionar una familia apropiada de funciones de pertenencia parametrizadas.

- Entrevistar a los expertos humanos para determinar los parámetros de las funciones de pertenencia usadas (y los coeficientes de las funciones consecuentes si se usa el modelo de Takagi-Sugeno) o refinar los parámetros de las funciones de pertenencias usando técnicas de regresión y optimización (si hay un conjunto de datos de entrada-salida).

1.2.2 Modelo Takagi-Sugeno

El modelo difuso Takagi-Sugeno (Takagi y Sugeno, 1985; Sugeno y Kang, 1988) puede ser visto como una combinación de modelación lingüística y modelación matemática de regresión en el sentido que el antecedente describe regiones difusas del espacio de entrada donde el consecuente funcional es válido. Las reglas en este tipo de sistema son de la forma:

$$\text{if } x_1 \text{ is } A_{i1} \text{ and } K \text{ and } x_r \text{ is } A_{ir} \text{ then } y_i = f_i(x_1, K, x_r) \quad (1.1)$$

donde los A_{ij} son conjuntos difusos y f_i es una función real. Generalmente f_i es un polinomio de las variables de entrada x_1, K, x_r

Las funciones f_i de un modelo son generalmente de la misma estructura, solamente cambian los parámetros de cada regla. Una parametrización útil en la práctica, que mejora las capacidades de aproximación de los modelos TS (Castro, 1995), es la afín con las reglas en la forma:

$$\text{if } x_1 \text{ is } A_{i1} \text{ and } K \text{ and } x_r \text{ is } A_{ir} \text{ then } y_i = b_{i0} + b_{i1}x_1 + K + b_{ir}x_r \quad (1.2)$$

con $i = 1, K, L$. Donde L es el número de reglas, x_r son las variables de entrada, y_i son las variables de salida locales, A_{ij} son los conjuntos difusos caracterizados por las funciones de pertenencia $\mu_{A_{ij}}(x_r)$, y b_{ij} son escalares. Este modelo es llamado *modelo Takagi-Sugeno afín*, el cual es esencialmente una aproximación multi-modelo en la cual modelos lineales simples se combinan para describir el comportamiento global del sistema. Esta propiedad facilita el análisis de los modelos Takagi-Sugeno en un marco similar al de los sistemas lineales.

Por otro lado, el modelo difuso Takagi-Sugeno requiere de un pequeño número de reglas para describir procesos altamente no lineales, el cual puede ser significativamente más pequeño que cuando se aplican otros tipos de modelos difusos. Además, los modelos difusos Takagi-Sugeno son computacionalmente eficientes. Debido a estas características, nos hemos centrado en la obtención de este tipo de sistema difuso para la modelación e identificación de sistemas no lineales.

1.2.3 Mecanismo de inferencia

La salida del sistema difuso Takagi-Sugeno es un promedio ponderado de los valores de los consecuentes de las reglas. La fórmula de inferencia es:

$$y = \frac{\sum_{i=1}^L w_i y_i}{\sum_{i=1}^L w_i} = \frac{\sum_{i=1}^L w_i (b_{i0} + b_{i1}x_1 + K + b_{ir}x_r)}{\sum_{i=1}^L w_i} \quad (1.3)$$

donde y es la salida total del sistema, y w_i es el nivel de impacto de la regla i , que se define:

$$w_i = \mu_{A_{i1}}(x_1) \times \mu_{A_{i2}}(x_2) \times K \times \mu_{A_{ir}}(x_r) \quad (1.4)$$

La ecuación 1.3 puede describirse de una forma ligeramente diferente

$$y = \sum_{i=1}^L \tau_i (b_{i0} + b_{i1}x_1 + K + b_{ir}x_r), \quad \text{donde } \tau_i = \frac{w_i}{\sum_{i=1}^L w_i} \quad (1.5)$$

donde τ_i es el nivel de impacto normalizado.

Es la sencillez de estos modelos la que implica sus mayores problemas, los cuales son:

- (i) al ser sus consecuentes fórmulas matemáticas podrían no representar un marco natural para representar el conocimiento, y
- (ii) no hay mucha libertad para aplicar diferentes principios de lógica difusa por lo que se pierde versatilidad.

Sus principales ventajas radican en que son computacionalmente eficientes, trabajan bien con técnicas lineales y con técnicas de optimización y adaptativas, garantiza continuidad de la superficie de salida, y es apropiado para el análisis matemático.

1.2.4 Modelo simplificado Takagi-Sugeno

El modelo simplificado de razonamiento (también denominado modelo TS de orden cero o modelo *singleton*) es un caso particular del modelo Takagi-Sugeno. Este modelo se obtiene cuando los coeficientes b_{iL}, K, b_{iR} valen cero. En este caso, las reglas tienen la siguiente forma (renombrando a las constantes b_{i0} simplemente como b_i):

$$\begin{aligned}
 &\text{if } (x_L \text{ is } A_{L1}) \text{ and } K \text{ and } (x_R \text{ is } A_{R1}) , \text{ then } y_1 = b_1 \\
 &\text{also} \\
 &K \\
 &\text{also} \\
 &\text{if } (x_L \text{ is } A_{Ll}) \text{ and } K \text{ and } (x_R \text{ is } A_{Rl}) , \text{ then } y_L = b_L
 \end{aligned} \tag{1.6}$$

1.3 Clusterización

La clusterización devino un importante mecanismo para la modelación difusa, pues su aplicación disminuye los gastos computacionales ya que facilita una gran reducción de la base de reglas difusas. Los modelos difusos emplean este método para formar las bases de reglas, es por ello que en este epígrafe se hará un estudio general sobre los tipos de clusterización difusa y clásica, analizando el principio de funcionamiento de algunos de sus algoritmos, pero antes es importante dejar claro el concepto básico y las características esenciales de los clusters.

Un cluster es un grupo de objetos que tienen características más comunes entre ellos que con los demás miembros de otros clusters. Las características comunes deben ser interpretadas como similitudes matemáticas, los cuales son definidos por alguna norma de distancia. La distancia puede ser medida entre los propios vectores de datos o como la distancia entre el vector de datos y un prototipo del cluster.

Los prototipos son vectores de igual dimensión que los datos, usualmente no se conocen de antemano y son buscados por los algoritmos de clusterización simultáneamente con la

partición de los datos. Los prototipos se definen como objetos de alto nivel geométrico tales como subespacios lineales, no lineales o funciones.

Los datos pueden formar clusters de diferentes formas geométricas, tamaños y densidades. Pueden ser esféricos, al igual que pueden ser caracterizados como subespacios lineales y no lineales del espacio de los datos. El desempeño de la mayoría de los algoritmos de extracción de clusters se ve influenciado no solo por la forma geométrica y densidad de los clusters individualmente sino también por las relaciones espaciales y las distancias entre cada uno de ellos.

1.3.1 Algoritmos de clusterización difusa

Los algoritmos de clusterización difusa más utilizados son los que se basan en la optimización de un funcional relativo a la norma de distancia entre los diferentes datos y los prototipos. El ejemplo más evidente de este tipo de algoritmo es el FCM (*Fuzzy C-Means*) ya que constituye la base para el resto de los algoritmos, pero también son populares el GK (*Gustafson-Kessel*), el PCM (*Possibilistic C-Means*), el FCV (*Fuzzy C-Varieties*) y el FCRM (*Fuzzy C-Regression Models*).

De forma general estos algoritmos tienen como restricción que la suma de las funciones de pertenencia de un dato con respecto a todos los clusters generados por el algoritmo sea igual a la unidad, lo cual no siempre se corresponde con los grados reales que se le puede asignar a los datos. El FCM y GK se caracterizan porque sus prototipos son representados por puntos, o sea, de la misma dimensión que los datos lo que equivale a estructuras geométricas del mismo tipo que los datos.

En particular FCM (Bezdek, 1981) es un algoritmo que usa norma de la distancia fija e impone clusters de forma circular o hiperesferas. No es capaz de detectar clusters de diferentes formas y de diferente orientación en el conjunto de los datos.

El GK (Gustafson y Kessel, 1979) extiende el algoritmo Fuzzy C-Means empleando una norma de distancia adaptativa para detectar clusters de diferentes formas geométricas en el conjunto de los datos. En este algoritmo, cada cluster tiene su propia matriz de inducción de norma, la cual se toma como variable de optimización en el funcional. Esto tiene como ventaja que cada cluster pueda adaptar su norma de distancia a las estructuras topológicas

locales de los datos. El algoritmo GK induce clusters de forma hiperelipsoidales a diferencia del algoritmo FCM, debido a que utiliza una norma de distancia de tipo Mahalanobis y matriz de covarianza de los clusters. Una desventaja del algoritmo GK es que debido a sus restricciones este solo puede encontrar clusters de aproximadamente igual volumen. Para erradicar esta desventaja el algoritmo GK se combina con un procedimiento de solapamiento de clusters compatibles para detectar clusters de diferentes volúmenes.

Otra desventaja del algoritmo GK es que requiere de un mayor esfuerzo computacional debido a que la inversa y el determinante de la matriz de covarianza de los clusters necesita calcularse en cada iteración del algoritmo.

El algoritmo FCV (Bezdek, 1981) define a diferencia de los algoritmos GK y FCM los prototipos como subespacios lineales y no lineales r -dimensionales. Puede detectar clusters que están en subespacios lineales r -dimensionales de R^n . También este algoritmo se deriva de una generalización del algoritmo FCM.

Uno de los inconvenientes del algoritmo FCV es que las variedades lineales no están limitadas en tamaño y por esto el algoritmo tiende a conectar clusters que están bien separados. Para eliminar los inconvenientes de este algoritmo se utilizan otros algoritmos de clusterización.

El algoritmo FCRM (Hathaway y Bezdek, 1993), al igual que el FCV, utiliza subespacios lineales y no lineales r -dimensionales para definir sus prototipos. Puede detectar clusters que están en subespacios lineales r -dimensionales de R^n y al igual que el FCV se deriva del FCM. Estima los parámetros de los modelos de regresión simultáneamente con la partición difusa de los datos.

Este algoritmo sufre el mismo inconveniente del algoritmo FCV de no estar limitados los clusters en tamaños. La ventaja de este algoritmo es que puede ajustar localmente los modelos no lineales de los datos, tales como polinomios. Estos modelos aún son lineales en sus parámetros y por lo tanto conlleva a un problema de estimación lineal.

El algoritmo PCM (Krishnapuram y Séller, 1993) asigna grados de pertenencia entre 0 y 1, pero no cumple con la restricción de que la sumatoria de los todos los grados de pertenencia de un dato con respecto a los clusters generados sea igual a la unidad. Se puede considerar

que es un algoritmo en el que su funcional se deriva del funcional del algoritmo FCM. La matriz que contiene los grados de pertenencia de los datos con respecto a los clusters tiene sus columnas independientes lo cual hace posible la descomposición global de la función objetivo en c funciones objetivos para los c clusters. Al eliminar la restricción $\sum_{i=1}^c \mu_{ik} = 1$, las funciones de pertenencia de los c clusters se convierten en independientes las unas de las otras. Esto hace al algoritmo PCM más sensible a la inicialización puesto que nada impide que el algoritmo converja a una partición posibilística donde todos los clusters sean idénticos o muy similares entre ellos.

1.3.2 Algoritmos de clusterización clásica

Los más populares algoritmos para la realización de la clusterización clásica son los conocidos como algoritmos de un solo paso, pues estos requieren un menor esfuerzo computacional, de ahí su utilización en la identificación en línea (Fernández, 2003).

Como métodos de un solo paso, encontramos el ECM (*Evolving Clustering Method*) o Método de Clusterización Evolutivo (Song y Kasabov, 2001). Este algoritmo se basa en el cálculo de la distancia euclidiana normalizada entre los prototipos de los clusters y los datos que van procesándose. Para cada cluster, la distancia entre su prototipo y el dato más alejado tiene que ser menor o igual que el valor conocido como radio umbral, que debe ser definido por el usuario, y que deja claro el tamaño máximo que puede tener un cluster. Este parámetro afecta también la cantidad de grupos a crear. Un cluster no debe ser actualizado nuevamente cuando su radio R_u se iguala al valor umbral radio umbral ($R_u = \text{radio umbral}$).

En el proceso de clusterización, las muestras de datos provienen de una cadena de datos y se comienza con un conjunto vacío de clusters. Cuando se crea un nuevo grupo, se coloca su prototipo C_c en la misma posición del dato que lo originó, mientras que su radio R_u inicialmente se pone a 0 ($R_u = 0$).

Con las siguientes muestras, presentadas una a la vez, algunos de los grupos ya creados se actualizarán, cambiando la posición de sus prototipos, así como el valor del radio, mientras que, en otras ocasiones, serán creados nuevos clusters. En dependencia de la posición del

dato que se procesa y de los diferentes clusters que existan, se determina si se debe actualizar un cluster existente o crear uno nuevo.

El Aglomerativo es otro de los métodos de clusterización de un solo paso (Guedalia., 1998). Se considera también un método supervisado, o sea, necesita que el usuario le suministre al menos un parámetro, que influye en gran medida en su funcionamiento. En este caso se le debe proporcionar la cantidad máxima de clusters que pueden ser generados, N .

Este es un algoritmo de un solo paso, con minimización de la distorsión global y que permite seguir los nuevos datos y preservar la estructura existente al mismo tiempo. Este algoritmo se puede resumir en tres pasos: mueve el prototipo más próximo hacia la muestra, une los dos prototipos más cercanos, esto deja una capacidad de centro libre, y por último hace el centro libre igual a la muestra.

La minimización de la varianza interna de cada cluster, la maximización de la distancia entre los clusters y la adaptación a los cambios temporales en la distribución de los datos, son los principios fundamentales de su funcionamiento.

El algoritmo Modificado es un método clusterización clásica supervisado y de un solo paso, que esta concebido a partir de una modificación de los métodos EMC y el Aglomerativo por lo que poseen características comunes, para aprovechar las principales potencialidades de ambos métodos (Fernández, 2003).

Cuando comienza el proceso, el número de clusters es cero, por lo que el primer dato se toma como el primer prototipo.

Su funcionamiento se basa en el cálculo de la distancia euclidiana normalizada, esta vez, para establecer comparaciones, no solo entre los prototipos de clusters y los datos, sino también entre los prototipos de clusters.

Para cada cluster, se define su radio umbral, que es la distancia máxima que puede haber entre el prototipo y el dato que le pertenece y se encuentra mas alejado. También se define la similitud umbral como la distancia mínima posible entre dos centros de clusters. Ambos parámetros deben ser definidos con anterioridad por el usuario.

Según el resultado de las comparaciones se puede llegar a decisiones como la actualización del prototipo de un cluster ya existente, la creación de un nuevo cluster, y la unión de los dos clusters que poseen las características más semejantes. Cuando se decide actualizar el prototipo de un cluster, la nueva posición se determina teniendo en cuenta los datos anteriores indirectamente, a través del peso del cluster (*Cantidad de datos*). Mientras mayor sea el peso de un cluster menor será la modificación que se producirá sobre el cuando aparezca un nuevo dato que le pertenezca.

Cada nuevo dato, después de incluirse como miembro de uno de los clusters existentes, en caso que pertenezca a alguno, se considera un nuevo prototipo de cluster.

El método es capaz de inferir el número óptimo de clusters, y al mismo tiempo da la posibilidad de que, como cada nuevo dato es considerado un prototipo de cluster, se anticipe a la creación de nuevos grupos.

1.4 Conclusiones

Una vez establecidas en este capítulo las características de los modelos difusos, fundamentalmente de los modelos Takagi-Sugeno, y exponer las principales características de varios algoritmos de clusterización, útiles en la modelación difusa, en el próximo capítulo abordaremos la obtención de modelos difusos a partir de la clusterización del conjunto de datos de entrada-salida del proceso a modelar.

CAPÍTULO 2. OBTENCIÓN DE MODELOS DIFUSOS

En ocasiones, los investigadores lidian con sistemas para los cuales no es posible aplicar las técnicas de modelado tradicional debido a la falta de conocimiento preciso y formal del sistema, a un comportamiento altamente no lineal de este, a un alto grado de incertidumbre en los datos, etc. A estos sistemas se les conoce como sistemas complejos.

Una herramienta potente para el modelado de sistemas complejo es la obtención de modelos basado en reglas difusas o simplemente, modelos difusos, asunto que abordaremos en este capítulo.

En el epígrafe 2.2 se describen varias medidas de validez útiles para el análisis de las cualidades de los clusters. En el epígrafes 2.3 son analizadas las proyecciones de los clusters obtenidos mediante los métodos de clusterización difusa y clásicas, las que permiten determinar los parámetros de las funciones de pertenencia. En el epígrafe 2.5 se estudia detalladamente un algoritmo de simplificación de bases de reglas y en el epígrafe 2.6 son estudiadas las funciones implementadas que dan cumplimiento al procedimiento seguido para la obtención de modelos difusos descrito en el epígrafe 2.1.

2.1 Descripción del proceso de identificación

Como es de esperar, seguir un procedimiento para la identificación de modelos difusos y cumplir con cada uno de sus pasos es un requisito primordial para alcanzar los resultados esperados. A continuación, se describe en que consiste cada uno de los pasos del procedimiento para la obtención de modelos difusos, el cual es mostrado en la figura 2.1.

Paso-1: Diseño de experimentos y la colección de datos.

Este es un paso importante para cualquier método de identificación pues en él determinamos el volumen de información del conjunto de datos de identificación.

Paso-2: Selección de la estructura.

En este paso se propone determinar las variables de entradas y salidas relevantes con respecto al objetivo del proceso. Cuando identificamos un sistema dinámico la estructura y el orden del modelo pueden ser elegidos. La selección de la estructura nos permite transformar la identificación de un sistema dinámico en un problema de regresión estático o cuasi estático. La estructura puede ser seleccionada de una manera automatizada comparando estructuras con diferentes medidas de desempeño. En muchos casos la elección razonable puede hacerla el usuario, basado en su experiencia sobre el proceso.

Paso-3: Clusterización de los datos y selección del número de clusters.

La clusterización, en el contexto relativo a la modelación de sistemas, es el mecanismo mediante el cual un proceso complejo y no lineal se divide en varios subsistemas de menor complejidad más fácilmente interpretables. A cada uno de estos subsistemas se le denomina agrupamiento, grupo o cluster y está compuesto por un grupo de datos del espacio de los datos del proceso que se asemeja más al centro de este cluster que a cualquier otro centro de cluster identificado en el proceso. Al centro de cluster también se le denomina prototipo, pues representa de forma aproximada las características de todos los datos que contiene.

La localización y los parámetros de los submodelos son encontrados mediante esta forma de partición de los datos toda vez que cada uno de los cluster define una región en la cual el sistema puede aproximarse localmente a una submodelo lineal.

Al unísono con la clusterización, se encuentra el número apropiado de clusters aplicando medidas de validez. Este paso envuelve varias repeticiones de ambos procesos hasta tanto no se encuentre la partición ‘óptima’.

Paso-4: Modelo difuso Takagi-Sugeno

En este paso se obtendrán modelos difusos del tipo Takagi-Sugeno. La clusterización divide los datos en clusters, los cuales son representados por reglas que tienen un rango de validez limitado. Los antecedentes de dichas reglas son encontrados por la proyección de

los clusters en cada una de sus dimensiones. Los consecuentes son funciones lineales que dependen de las entradas y cuyos parámetros son determinadas por el método de los mínimos cuadrados de Gauss.

Paso-5: Simplificación de la base de reglas:

El modelo obtenido inicialmente puede ser redundante en el sentido de contener más funciones de pertenencias de las que son necesarias para describir el sistema. Las medidas de similitud difusa pueden ser aplicadas para simplificar o reducir la base de regla inicial y para obtener interpretaciones lingüísticas de las funciones de pertenencias.

Paso-6: Validación del Modelo.

Por medio de la validación el modelo final es aceptado como apropiado para el propósito dado o es rechazado. Este paso será abordado con mayor profundidad en el Capítulo 3.

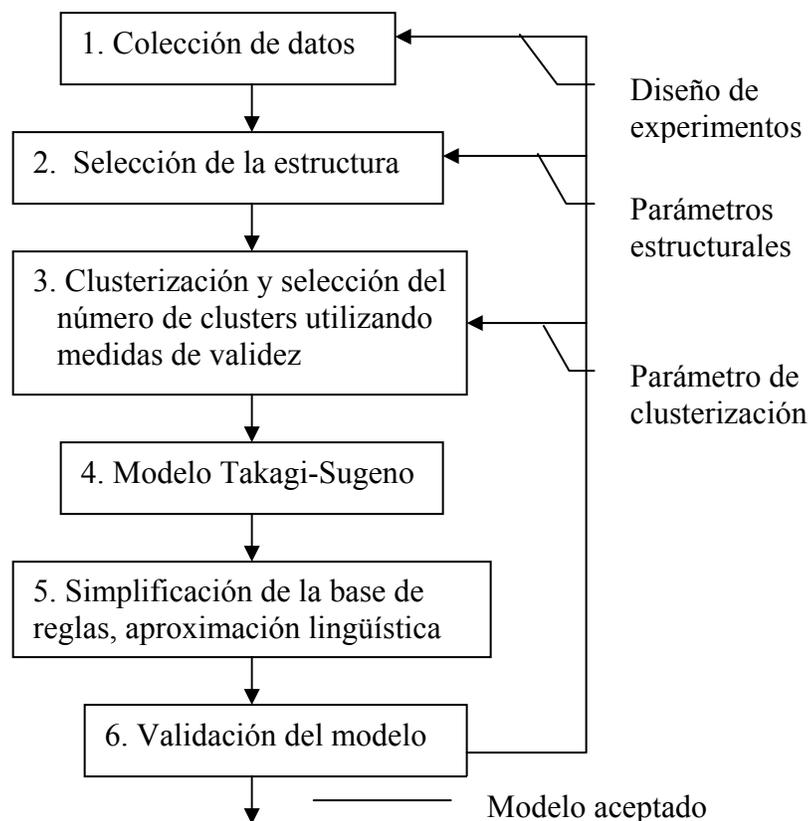


Figura 2.1. Descripción de la identificación de modelos difusos basados en la clusterización.

2.2 Medidas de validez

En el paso 3 se comentaba sobre la necesidad de encontrar el número de clusters ‘óptimo’, aplicando criterios de medidas de validez junto con la clusterización, para la obtención de modelos difusos con un buen comportamiento. Dada la importancia de este proceso, en este epígrafe se tratarán los fundamentos teóricos de algunos de estos criterios utilizados en la práctica.

Las medidas de validez de los clusters han sido extensivamente estudiadas en el contexto de reconocimiento del modelo (Babuska, 1998). Estos criterios evalúan las cualidades de los clusters, entre ellos encontramos el hipervolumen, la distancia promedio interna del cluster, la densidad de la partición difusa, etc. El número de clusters ‘óptimo’ se encuentra evaluando una medida de validez dada en un rango determinado, y seleccionando el número de clusters que minimiza o maximiza la medida de validez. En lugar de esto, si la medida de validez es monótona, el número de clusters puede encontrarse en un punto de “rodilla”.

Gath y Geva (1989) proponen el hipervolumen difuso y la densidad de partición promedio para los algoritmos de clusterización de distancia adaptable. El hipervolumen difuso (V_h) se define como:

$$V_h = \sum_{i=1}^c [\det(F_i)]^{1/2} \quad (2.1)$$

donde las F_i son las matrices de covarianza de los clusters.

La densidad de partición promedio (D_A) está dada por:

$$D_A = \frac{1}{c} \sum_{i=1}^c \frac{S_i}{[\det(F_i)]^{1/2}}, \quad (2.2)$$

donde S_i es calculado usando solo los vectores de datos z_k que están dentro del hiperelipsoide cuyos radios son las desviaciones estándares del cluster.

$$S_i = \sum_k \mu_{ik}, \forall k, \text{ tal que } (z_k - v_i)^T F_i^{-1} (z_k - v_i) < 1. \quad (2.3)$$

donde los μ_{ik} son los grados de pertenencia de los datos y los v_i prototipos de los clusters. La densidad de partición (D_p) esta definida por:

$$D_p = \frac{\sum_{i=1}^c S_i}{V_h} \quad (2.4)$$

Krishnapuram y Fred (1992) aplicaron el uso de la distancia promedio interna del cluster (D_w)

$$W_A = \frac{1}{c} \sum_{i=1}^c \frac{\sum_{k=1}^N (\mu_{ik})^m d_{ik}^2}{\sum_{k=1}^N (\mu_{ik})^m}. \quad (2.5)$$

y la distancia total interna del cluster (W_T) dada por:

$$W_T = \sum_{i=1}^c \sum_{k=1}^N \mu_{ik}^m d_{ik}^2 \quad (2.6)$$

Una buena partición es indicada por pequeños valores de V_h , D_A y D_p , y por altos valores de W_A y W_T .

2.3 Proyección de clusters

Una vez terminada la clusterización, para encontrar los parámetros de las funciones de pertenencia de los antecedentes de la base de reglas, es necesario proyectar los clusters en cada una de las dimensiones del espacio del antecedente. Para esto existen dos tipos de proyecciones, una para la clusterización clásica y otra para la difusa.

A continuación se explicará en que consiste cada una de estas proyecciones.

2.3.1 Proyección en la clusterización clásica

Los algoritmos de clusterización clásica que son utilizados en este trabajo generan clusters con formas geométricas hiperesféricas. Para realizar la proyección de estos clusters se determinan los radios de cada uno de ellos encontrando la distancia euclidiana máxima entre el prototipo y los datos que pertenecen a él. Una vez encontrado los radios se calculan

las sigmas de las funciones de pertenencia gaussianas considerando que estas son proporcionales a las proyecciones de los radios en sus dimensiones o sea:

$$\sigma = \frac{\text{radio}}{K} \quad (2.7)$$

El objetivo de esta consideración es encontrar un valor de sigma para el cual los puntos que se encuentran fuera de los límites del cluster proyectado tengan un pequeño grado de pertenencia. En este trabajo, tras varias comprobaciones, se decidió considerar a $K = \sqrt{8}$.

2.3.2 Proyección en la clusterización difusa

El principio de este método es proyectar los conjuntos difusos multidimensionales, definidos punto a punto en las filas de la matriz partición (U) que devuelven los métodos de clusterización difusa, hacia las variables individuales del antecedente de las reglas. Estas variables pueden ser las variables originales de la regresión en caso de que la proyección sea solo una proyección ortogonal del dato. Las variables antecedentes transformadas también pueden ser obtenidas por medio de la proyección del vector usando el primer vector propio de la matriz covarianza del cluster. Esta proyección del vector propio es útil para los clusters que son oblicuos a los ejes de la regresión y no pueden ser representados con precisión por proyección en ejes ortogonales (Babuska, 1998). En ambos casos las funciones de pertenencia proyectadas (definida punto a punto) son aproximadas por alguna función de pertenencia paramétrica apropiada como se muestra en la figura 2.2

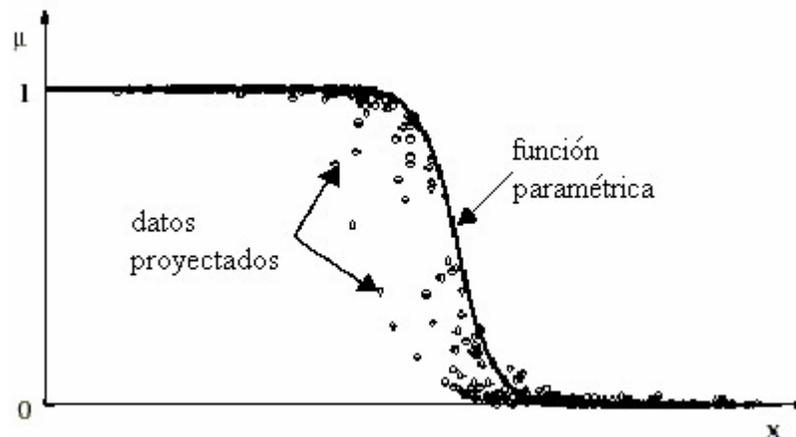


Figura 2.2. Aproximación de los datos proyectados por una función paramétrica

La función exponencial por tramo definida como:

$$\mu(x, c_l, c_r, w_l, w_r) = \begin{cases} \exp\left(-\left(\frac{x - c_l}{2w_l}\right)^2\right), & \text{si } x < c_l, \\ \exp\left(-\left(\frac{x - c_r}{2w_r}\right)^2\right), & \text{si } x > c_r, \\ 1, & \text{en otros casos,} \end{cases} \quad (2.8)$$

donde c_r y c_l son conocidos, como hombros derecho e izquierdo, y w_r y w_l , son las anchuras derechas e izquierda (ver figura 2.3), demostró ser conveniente para la representación exacta de la forma de los clusters. Esta función es ajustada, para que describa los datos proyectados, optimizando numéricamente sus parámetros. Una ventaja de este método es que las funciones de pertenencias proyectadas puedan siempre ser aproximadas de manera tal que los conjuntos difusos sean obtenidos (Babuska, 1998).

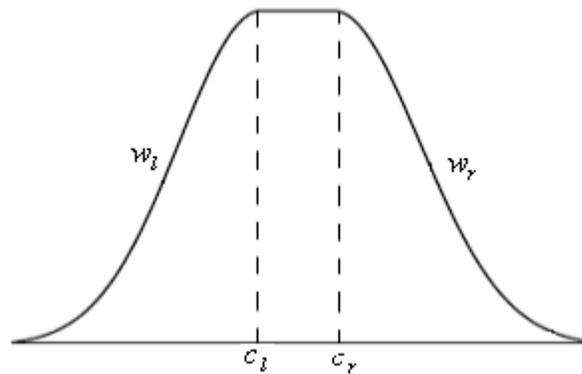


Figura 2.3. Función exponencial por tramos.

2.4 Simplificación de la base de reglas

Un buen algoritmo de reducción de reglas elimina el gasto computacional innecesario, por ese motivo y para dar cumplimiento al paso 5 del procedimiento del epígrafe 2.1, se tratan en este epígrafe las bases teóricas del algoritmo de reducción seleccionado en este trabajo.

Las aproximaciones automatizadas para la adquisición de modelos difusos a partir de los datos, como producto espacial de clusters o las técnicas de adaptación de parámetros, llevan a un cierto grado de redundancia en el modelo obtenido. Con técnicas de clusterización, la

redundancia ocurre cuando los clusters se proyectan hacia el antecedente de las variables individuales, proceso que se comentó en el epígrafe precedente.

Veamos un ejemplo donde se ilustra lo anterior. En la figura 2.4, la proyección de los clusters C_1 , C_2 y C_3 hacia variables del antecedente x_1 y x_2 muestran una similitud entre los conjuntos A_1 y A_2 , mientras que B_3 es muy similar al conjunto universal ($\mu = 1$) que cubre el dominio entero de x_2 .

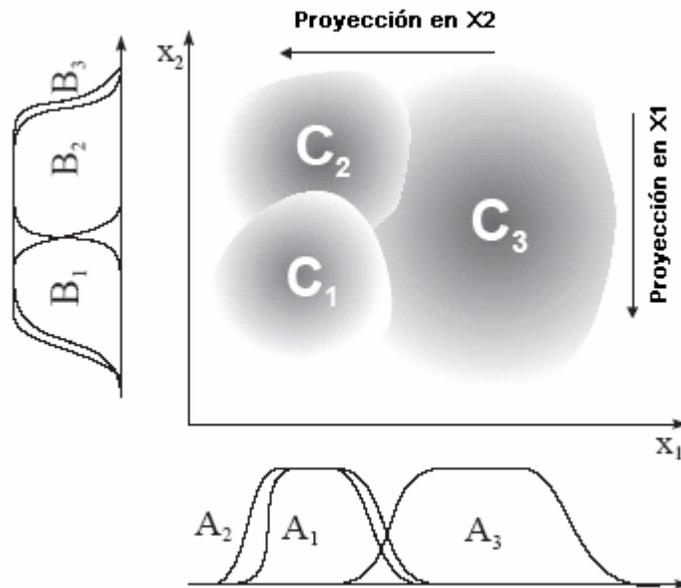


Figura 2.4. Redundancia en términos de funciones de pertenencia similares.

La simplificación de la base de reglas es lograda uniendo conjuntos difusos similares, y eliminando conjuntos difusos similares al conjunto universal (estos conjuntos difusos no contribuyen a la base de reglas). Un gran número de medidas de similitud son conocidas de la literatura. Basado en un estudio investigativo extenso (Setnes, 1995), la siguiente medida de similitud se selecciona para la simplificación de la base de reglas:

$$S(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{\sum_{q=1}^{n_d} (\mu_A(x_q) \wedge \mu_B(x_q))}{\sum_{q=1}^{n_d} (\mu_A(x_q) \vee \mu_B(x_q))}, \quad (2.9)$$

Donde $|\cdot|$ denota la cardinalidad de un conjunto y $\mu_A(x_j)$ y $\mu_B(x_j)$ son las funciones de pertenencia que describen los conjuntos difusos a comparar.

El algoritmo de simplificación une conjuntos difusos similares iterativamente usando dos umbrales: $\eta \in (0,1)$ para unir conjuntos difusos que son similares entre sí, y $\eta_r \in (0,1)$ para eliminar conjuntos difusos similares al conjunto universal. En cada iteración, la similitud entre todos los conjuntos difusos para cada variable del antecedente es considerada. El par de funciones de pertenencia con más alta similitud y que cumple que $s \geq \eta$ son unidas. La base de reglas es actualizada colocando el nuevo conjunto difuso en lugar de los que fueron fusionados. El algoritmo repetitivamente evalúa las similitudes en la base de reglas actualizadas hasta que no existan más conjuntos para los cuales $s \geq \eta$. Finalmente, conjuntos similares al conjunto universal son eliminados del antecedente de las reglas. El algoritmo solo une un par de conjuntos por iteración y en dimensiones diferentes, por lo que cada dimensión es independiente de las otras. La unión de dos conjuntos difusos $A_{l,q}$ y $A_{m,q}$ es lograda tomando el soporte¹ del nuevo conjunto difuso A como el soporte de $A_{l,q} \cup A_{m,q}$. Esto garantiza la preservación del alcance del espacio del antecedente. El centro de A es dado por el promedio de $A_{l,q}$ y $A_{m,q}$. El algoritmo puede ser resumido como sigue:

Algoritmo 2.1. Algoritmo de simplificación de base de reglas.

Dada una base de reglas obtenidas por agrupamiento, seleccionar los umbrales $\eta_r, \eta \in (0,1)$.

1. Seleccionar los dos conjuntos más similares en la base de reglas.

- Calcular $s_{ijk} = S(A_{ij}, A_{kj}), j = 1, 2, \dots, K, p, i, k = 1, 2, \dots, K$.
- Seleccionar $A_{l,q}$ y $A_{m,q}$ tal que $s_{lmq} = \max_{i,j,k,i \neq k} \{s_{ijk}\}$

2. Unir los dos conjuntos difusos más similares y actualizar la base de reglas.

¹ El soporte de un conjunto difuso A es el subconjunto X cuyos elementos tienen grado de pertenencia distinto de cero:

$$X = \text{soporte}(A) = \{x | \mu_A > 0\}$$

- Si $S(A_{ij}, A_{mq}) \geq \eta$ unir A_{iq} y A_{mq} para crear un nuevo conjunto difuso A
 - Reemplazar $A_{iq} = A$ y $A_{mq} = A$.
3. Retornar al primer paso hasta que ningún otro conjunto difuso tenga similitud $s_{ijk} \geq \eta$
 4. Eliminar los conjuntos difusos similares al conjunto universal.
 - Para cada conjunto A_{ij} calcular $S(A_{ij}, U)$, donde $\mu_U = 1, \forall x_j$.
 - Si $S(A_{ij}, U) \geq \eta_r$ eliminar A_{ij} del antecedente de la i -ésima regla.

La fusión de conjuntos difusos similares puede dar lugar a reglas con antecedentes iguales. Estas reglas pueden ser combinadas en una sola regla, permitiendo la reducción del número de regla.

2.5 Programación e Implementación

Para llevar a cabo parte del procedimiento del epígrafe 2.1, se ha formado una biblioteca de funciones en Matlab recopilando trabajos de otros investigadores e implementando algunas funciones necesarias como las que se describirán en este epígrafe. La figura 2.5 representa el orden en que son invocadas las funciones de la biblioteca según el procedimiento comentado al inicio del capítulo, pero asumiendo que los datos y la estructura del modelo son conocidos.

El análisis de las funciones de clusterización difusa y clásica puede encontrarse en los trabajos de diploma de Alfonso (2002) y Fernández (2003). Las características de la función *subclust* pueden hallarse en la ayuda de Matlab.

A continuación se explicará cada una de las funciones implementadas en este trabajo.

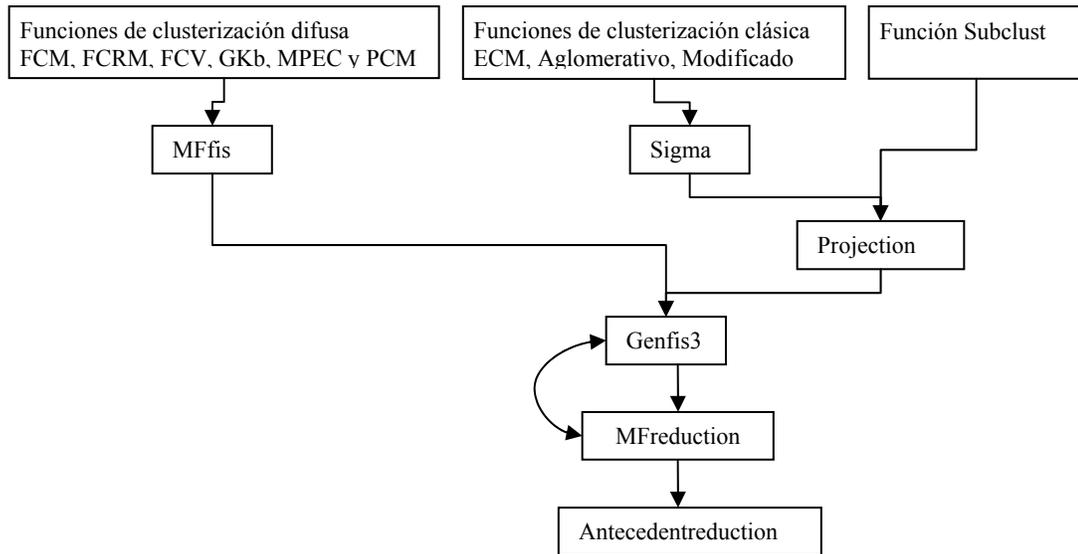


Figura 2.5. Secuencia de funciones para la obtención de modelos difusos.

Sigma

Determina las sigmas basándose en las consideraciones hechas en el sub-epígrafe 2.3.1 y selecciona los prototipos válidos de los clusters producto de los algoritmos de clusterización clásica. Si algún prototipo que representa a un solo dato, coincide con este, entonces es eliminado al igual que aquellos que no representan a ningún dato, para evitar clusters de radio cero.

La sintaxis de esta función es:

$$[sigmas, centers] = sigma(centers, Xin, Xout, flag)$$

flag es un vector fila que contiene el cluster al que pertenece cada dato.

Xin y *Xout* son matrices de entrada y salida de $M \times N$ y $M \times K$ respectivamente siendo M el número de datos, N y K las dimensiones de los datos de entrada y salida respectivamente.

centers (como parámetro de entrada) es una matriz de $C \times (N+K)$ que contiene los prototipos de los clusters, provenientes de las funciones de clusterización clásica donde C es el número de prototipos.

Los prototipos (*centers*) y las sigmas que devuelve la función tienen dimensión $C \times N$.

MFfis

Determina los Parámetros de las funciones de pertenencia. *MFfis* invoca a las funciones *smooth* y *mffit* (Babuska, 1998). *Smooth* obtiene las funciones de pertenencia de clusters difusos multidimensionales proyectados en las dimensiones individuales y *mffit* busca los parámetros óptimos de funciones de pertenencia ajustadas en conjuntos difusos, definidos de punto a punto, en el universo de discurso.

La sintaxis de *MFfis* es la siguiente:

$$[mfs] = MFfis(Xin, Xout, mu)$$

donde:

mu es la matriz que contiene los grados de pertenencia de cada dato con respecto a los prototipos.

mfs es una celda que contiene los parámetros de la función de pertenencia de cada conjunto difuso en cada uno de los antecedentes. En este caso se empleará la función exponencial por tramo comentada en el sub-epígrafe 2.3.2

Esta función esta implementada en la biblioteca de Matlab con el nombre de *gauss2mf*

$$gauss2mf(x, w_l, c_l, w_r, c_r)$$

Projection

Genera funciones de pertenencias tipo *gauss2mf* a partir de las sigmas y los prototipos de los clusters hiperesféricos generados por los algoritmos de clusterización clásica y el sustractivo, donde $w_r = w_l$ y $c_r = c_l$.

Genfis3

Es una modificación de *genfis2* y ambas generan un modelo difuso tipo Takagi-Sugeno. *Genfis2*, dentro de sí, invoca a la función *subclust* que devuelve los centros de los clusters y las sigmas de las funciones de pertenencia de cada antecedente, dicha función no es más que la implementación del algoritmo de agrupamiento sustractivo. A diferencia de *genfis2*, *genfis3* es independiente de los algoritmos de agrupamiento que se empleen, puede utilizar algoritmos de clusterización difusos y clásicos, incluyendo el sustractivo.

La sintaxis del `genfis3` es la siguiente:

$fismat = genfis3(Xin, Xout, mfs, xBounds, options)$

donde:

$xBounds$ es una matriz de $2 \times N$, que es usada para normalizar los datos de entrada y salida, donde N es el total de entradas y salidas. Cada columna contiene los valores máximo y mínimo para los correspondiente conjunto dato de entrada o salida. En caso de que $xBounds$ sea una matriz vacía o no se pase como argumento, los valores de datos, máximos y mínimos, son buscados en las matrices Xin y $Xout$.

$Options$ es un vector de cuatro elementos, que utiliza *subclust* para especificar una serie de parámetro. *Genfis3* solo emplea el último elemento ($options(4)$) para especificar si desea mostrar información durante el proceso, los demás no son necesarios.

Los parámetros de los consecuentes se encuentran por el método de los mínimos cuadrados de Gauss.

Mfreduction

Esta función es capaz de reducir el número de funciones de pertenencia de un modelo difuso, a partir de la unión de los pares de conjuntos difusos cuya similitud es mayor o igual que el coeficiente umbral η (η). Se estarán haciendo estas comparaciones mientras existan conjuntos que cumplan esta condición. Este algoritmo de comparación (ver epígrafe 2.4) se lleva a cabo en cada uno de los antecedentes del modelo. Si resulta que dos reglas llegan a tener las mismas funciones de pertenencia, en cada uno de sus antecedentes, producto de la reducción, entonces se llega a una simplificación de reglas.

La función de pertenencia resultante, después que dos conjuntos cumplen la condición de similitud, se considera que debe ser del tipo *gauss2mf*, representada en la figura 2.3, para evitar complicaciones, por ejemplo:

Suponga que los dos conjuntos A y B de la figura 2.6 cumplen la condición de similitud impuesta por el usuario y sean $[w_{1A} \ c_{1A} \ w_{rA} \ c_{rA}]$ y $[w_{1B} \ c_{1B} \ w_{rB} \ c_{rB}]$ los parámetros de sus funciones de pertenencia, la función resultante se toma de forma tal que $w_1 = w_{1A}$, $c_1 = c_{1A}$, $w_r = w_{rB}$, $c_r = c_{rB}$..

Para hallar los parámetros de los consecuentes una vez hecha las reducciones de funciones y de reglas pertinentes, *MFredution* invoca a *genfis3* pasándole los nuevos parámetros de las funciones de pertenencia para que este devuelva un nuevo modelo difuso con los consecuentes actualizados.

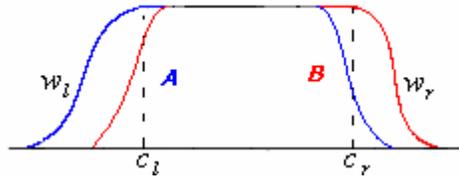


Figura 2.6. Conjuntos difusos que cumplen la condición de similitud

La sintaxis de la función es:

$fismat = MFredution(fismat, Xin, Xout, eta, xBounds, options)$

fismat, en los argumentos de entrada, es el modelo difuso obtenido por *genfis3* que se desea someter a la simplificación de base de reglas.

eta es el coeficiente umbral de similitud. En caso que *eta* sea nulo o no sea proporcionado se toma por defecto igual a 0.95.

Los demás argumentos fueron comentados en las funciones anteriores.

Antecedentreduction

Antecedentreduction reduce el número de antecedentes de cada regla mientras existan conjuntos difusos que cumplan la condición de similitud con relación al conjunto universal, si todos los antecedentes de una regla desaparecen, por cumplir la condición, se puede llegar a una reducción de reglas.

La sintaxis de esta función es la siguiente:

$fismat = Antecedentreduction(fismat, Xin, eta_r)$

donde *eta_r* (η_r) es el coeficiente umbral de similitud. Si es nulo o no es proporcionado se considera, por defecto, igual a 0.95.

2.6 Conclusiones

En este capítulo se han dado fundamentos teóricos de algunos de los pasos del procedimiento de identificación difusa visto en el epígrafe 2.1, además se hizo un pequeño estudio sobre las funciones implementadas en este trabajo, que forman parte de la biblioteca de funciones creada con el objetivo de obtener modelos difusos.

En el próximo capítulo veremos los resultados de la aplicación de lo aquí descrito en tres ejemplos reconocidos en la literatura.

CAPÍTULO 3. ANÁLISIS DE RESULTADOS

En este capítulo se aplicarán los desarrollos teóricos y las funciones implementadas en Matlab, temas que fueron explicados en el capítulo precedente. En el epígrafe 3.1 se obtiene un modelo difuso para un sistema no lineal propuesto por Narendra y Parthasarathy (1990) aplicando las medidas de validez para encontrar el número de cluster óptimo.

En el epígrafe 3.2, a partir de modelar un conjunto de datos reales tomados de Box y Jenkins (1980), se harán comparaciones entre los diferentes modelos difusos obtenidos en este trabajo y modelos difusos obtenidos por otros autores.

En el epígrafe 3.3 es modelado un proceso de neutralización del pH en un reactor de agitación continua (*CSTR*) y son comparadas las respuestas de modelos difusos, antes y después de ser sometido a una reducción de reglas.

Antes de pasar al análisis de los ejemplos, se hace notar que el comportamiento de los modelos obtenidos es evaluado mediante el criterio de error cuadrático medio (conocido por siglas en inglés como *MSE-Mean Square Errors*), el cual se define como:

$$MSE = \frac{\sum_{p=1}^N (y_p - f_p)^2}{N} \quad (3.1)$$

donde N es el número de datos, y_p es la salida observada y f_p la salida obtenida por el modelo para el p –ésimo dato de entrada.

3.1 Identificación de un sistema dinámico no lineal

El primer ejemplo que analizaremos es la identificación de un sistema dinámico no lineal descrito por la siguiente ecuación diferencial (Narendra y Parthasarathy, 1990):

$$y(k+1) = \frac{y(k)}{1+y^2(k)} + u^3(k) \quad (3.2)$$

donde $u(k)$ y $y(k)$ son la entrada y la salida del sistema respectivamente.

Para cumplir con el procedimiento planteado en el epígrafe 2.1, en el paso 1 se deben obtener los datos. Para esto se aplica la entrada:

$$u(k) = \begin{cases} \sin(2\pi k/100) & ; k \in [1,99] \\ \sin(2\pi k/100) + (0.5/(1+\rho))\sin(2\pi k/10) & ; k \in [100,250] \end{cases} \quad (3.3)$$

donde ρ es un número aleatorio uniformemente distribuido con $\rho \in [0,1]$.

Tomando en cuenta la estructura de la Ec. 3.2, se considera que la salida $y(k+1)$ está influenciada por $y(k)$ y $u(k)$ (Kukolj y Levi, 2004), i.e.:

$$y(k+1) = f(y(k), u(k)) \quad (3.4)$$

Además, para este ejemplo ha sido seleccionado como algoritmo de clusterización el FCM.

Para seguir dando cumplimiento al procedimiento de identificación, se procederá ahora a la determinación del número de clusters (paso 3) utilizando los criterios de medidas de validez descritos en el epígrafe 2.2.

Según las gráficas de medidas de validez de la figura 3.1 se puede determinar el número de clusters “óptimo” para el algoritmo de clusterización seleccionado. Debido a que no todas las medidas de validez brindan información útil en este caso, solo se consideró la distancia promedio interna del cluster (ec. 2.5) y la distancia interna del cluster (ec. 2.6).

Como se puede observar, y según lo planteado en el epígrafe 2.2, las gráficas son monótonas decrecientes, con una rodilla en cinco clusters, siendo este el número seleccionado para una buena partición.

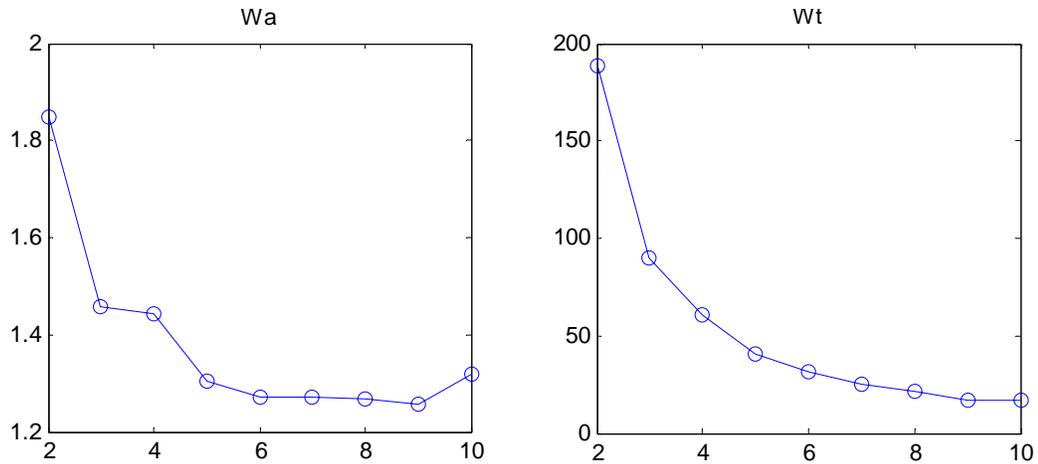


Figura 3.1 Representación de las medidas de validez seleccionadas para el conjunto de datos generados por la ecuación 3.2.

La figura 3.2 muestra el comportamiento de la salida inferida por el modelo difuso y la salida del sistema. Obsérvese que la coincidencia es casi perfecta. Además, el error cuadrático medio determinado ($MSE = 0.0022$) es una prueba de la exactitud de aproximación entre la función inferida y la real.

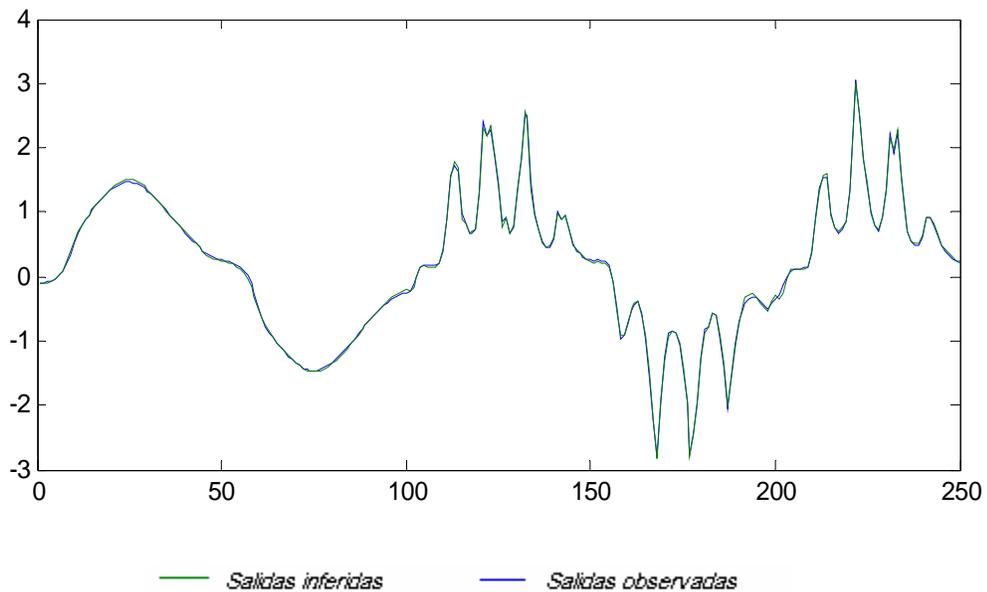


Figura 3.2. Gráfica de respuesta del modelo obtenido con el método FCM

3.2 Identificación del proceso de combustión en un horno

El ejemplo que se analizará a continuación parte de datos reales de un proceso de combustión de un gas dentro de un horno (Box y Jenkins, 1980), comúnmente usado en la validación de métodos de identificación. Consiste de 256 muestras de entrada-salida registrada con un período de muestreo de nueve segundos. Dicho proceso de combustión del gas tiene una variable de entrada, el flujo de gas $u(k)$, y una variable de salida ($y(k)$), que es la concentración de dióxido de carbono (CO_2). Siguiendo la recomendación de Kukolj y Levi (2004) con el objetivo de realizar comparaciones justas con otros métodos disponibles, se considera el valor instantáneo de la salida en $y(k)$ es influenciada por $y(k-1)$ y $u(k-4)$, las cuales se toman como entradas al modelo difuso.

En este ejemplo se han obtenido modelos con todos los algoritmos de clusterización, tanto clásica como difusa, con vista a encontrar el modelo difuso con mejores características. Para llevar a cabo las comparaciones es necesario apoyarse de la tabla 3.1 donde se muestran los errores cuadráticos medios de cada uno de los modelos difusos.

Tabla 3.1. Errores cuadráticos medios de cada uno de los modelos difusos.

Algoritmos	Variables	Reglas	MSE
Aglomerativo	2	2	0.1551
Modificado	2	2	0.1592
FCM	2	2	0.1624
FCRM	2	2	0.1694
FCV	2	2	0.1686
GKb	2	2	0.1688
MPEC	2	2	0.1590
PCM	2	2	0.1548
Sustractivo	2	2	0.1585

Como se puede ver, para un mismo número de reglas, el algoritmo de clusterización que da lugar al modelo difuso con menor MSE es el PCM. El comportamiento de la salida inferida de dicho modelo y la observada se ilustra en la figura 3.3.

Si se comparan los resultados obtenidos con los de otros autores, se puede concluir que la respuesta de los modelos difusos obtenidos en este trabajo son tan buenos o mejores que las presentadas por la mayoría de los modelos difusos mostrados en el Anexo 1 y con resultados bastante parecidos al modelo difuso tipo Takagi-Sugeno encontrado por Kukulj y Levi (2004).

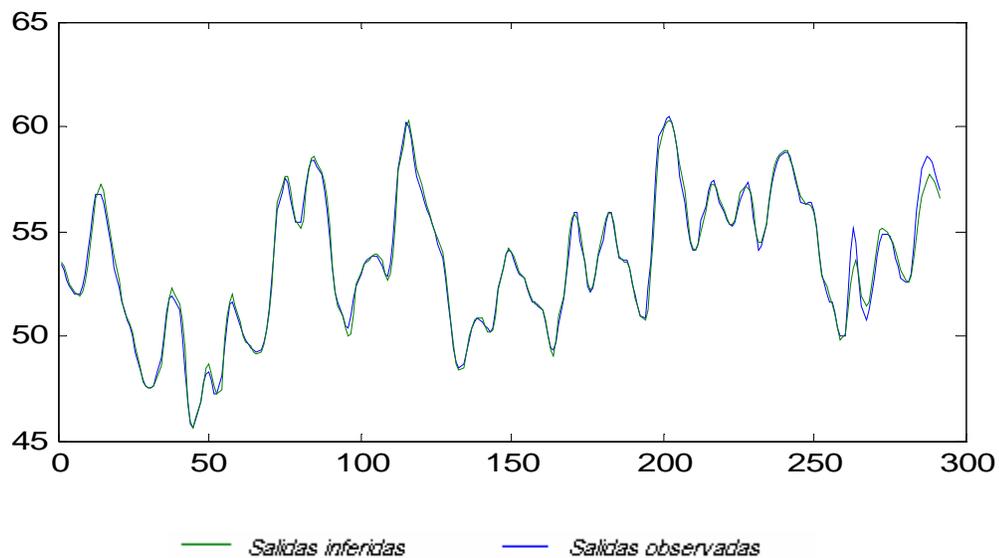


Figura 3.3. Aproximación de $y(k)$ a partir de $u(k-4), y(k-1)$

3.3 Identificación del proceso de neutralización del pH

Como último ejemplo se tiene un proceso de neutralización de pH en un CSTR (*Continuous Stirred Tank Reactor*), el cual es un proceso altamente no lineal que presenta complejidades debido a las grandes variaciones de la dinámica.

La variable de entrada del proceso, $u(k)$, es el flujo de hidróxido de sodio ($NaOH$) y la salida, $y(k)$, es el pH de la mezcla. Como este proceso es de primer orden, la salida en instantánea, $y(k+1)$, está influenciada por $y(k)$ y $u(k)$. Utilizando el algoritmo de clusterización FCM y para una partición de seis clusters, se obtiene la salida del modelo que muestra la figura 3.4. Nótese que la aproximación es muy buena.

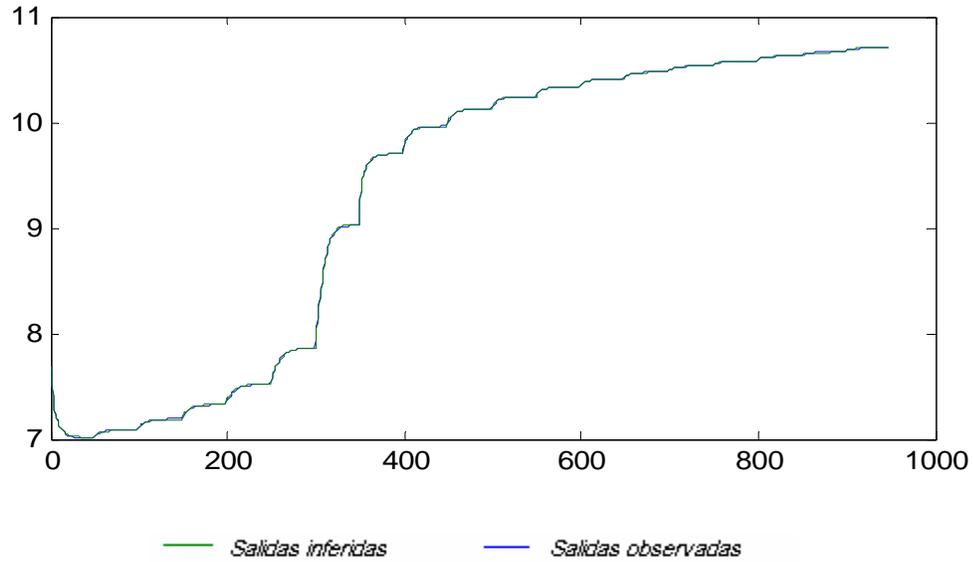


Figura 3.4. Aproximación de $y(k+1)$ a partir de $y(k), u(k)$

En aras de comprobar la efectividad del algoritmo de reducción de la base de reglas que se implementó en este trabajo, nos dimos a la tarea de aplicarlo al modelo obtenido.

El procedimiento se repite para varios valores del coeficiente umbral η , con $\eta_r = 0.95$. Los resultados alcanzados se muestran en la tabla 3.2, 3.3, 3.4 y 3.5.

Tabla 3.2. Errores cuadráticos medios antes y después de la reducción de la base de reglas*.

Base de reglas	η η_r		Antes de la reducción		Después de la reducción	
			No de reglas	MSE	No de reglas	MSE
I	0.7	0.95	6	$5.3009 \cdot 10^{-5}$	5	$5.8649 \cdot 10^{-5}$
II	0.6	0.95	6	$5.3086 \cdot 10^{-5}$	5	$4.4156 \cdot 10^{-5}$
III	0.5	0.95	6	$5.3085 \cdot 10^{-5}$	4	$5.1371 \cdot 10^{-5}$

*En todos los casos se trabaja con $\eta_r = 0.95$

Tabla 3.3. Base de reglas reducida I.

Regla	Antecedente	Consecuente
1	if in1 is in1mf1 and in2 is in2mf1	then out1 is out1mf1
2	if in1 is in1mf2 and in2 is in2mf2	then out1 is out1mf2
3	if and in2 is in2mf3	then out1 is out1mf3
4	if and in2 is in2mf2	then out1 is out1mf5
5	if in1 is in1mf6 and in2 is in2mf6	then out1 is out1mf6

Tabla 3.4. Base de reglas reducida II.

Regla	Antecedente	Consecuente
1	if and in2 is in2mf1	then out1 is out1mf1
2	if in1 is in1mf2 and in2 is in2mf1	then out1 is out1mf2
3	if and in2 is in2mf3	then out1 is out1mf3
4	if in1 is in1mf5 and in2 is in2mf5	then out1 is out1mf5
5	if and in2 is in2mf6	then out1 is out1mf6

Tabla 3.5. Base de reglas reducida III.

Regla	Antecedente	Consecuente
1	if and in2 is in2mf1	then out1 is out1mf1
2	if in1 is in1mf2 and in2 is in2mf2	then out1 is out1mf2
3	if and in2 is in2mf4	then out1 is out1mf4
4	if in1 is in1mf5 and in2 is in2mf1	then out1 is out1mf5

Como se puede observar en ocasiones el modelo con la base de reglas reducida tiene menor error cuadrático medio que el modelo antes de ser sometido a la reducción, esto demuestra que la reducción no afecta el comportamiento del modelo, en cambio, disminuye el gasto

computacional e incluso puede llegar a obtenerse modelos difusos con mejor comportamiento que el original.

3.4 Conclusiones

A modo de conclusión podemos decir que para validar las funciones desarrolladas, se han utilizado ejemplos de la literatura científica internacionalmente reconocidos, con el fin de obtener modelos difusos siguiendo el procedimiento del epígrafe 2.1, siempre partiendo de que los datos y las estructuras de los procesos estudiados fueron previamente conocidos. Se aplicaron las medidas de validez implementadas en este trabajo para encontrar el número de cluster ‘óptimo’, comprobando el buen comportamiento del modelo difuso obtenido. Se establecieron comparaciones entre modelos difusos obtenidos en este trabajo y los obtenidos por otros autores, resultando que los algoritmos de clusterización utilizados dan lugar a modelos difusos con características similares o mejores que los de varios autores representados en el Anexo 1. Finalmente, se probaron las funciones de reducción de reglas, comentadas en el epígrafe 2.5, con diferentes coeficientes umbrales, y se pudo constatar el buen comportamiento de los nuevos modelos difusos con bases de reglas reducidas.

CONCLUSIONES

Concluida esta investigación, después de un estudio teórico-práctico sobre distintos temas como la modelación difusa, los modelos Takagi-Sugeno como método aplicar en este trabajo, los algoritmos de clusterización difusa y clásica implementados en estudios precedentes, los criterios de medidas de validez para encontrar el número de clusters óptimo, los algoritmos de reducción de la base de reglas con vistas a simplificar el modelo sin provocar serias afectaciones en el comportamiento de este, entre otros aspectos, se puede destacar que:

- La modelación difusa, y específicamente la obtención de modelos difusos Takagi-Sugeno, ha constituido en los últimos años un área de estudio muy atractiva y activa. La combinación de esta con los métodos de clusterización ha centrado gran atención debido a la posibilidad de obtener modelos compactos, óptimos y eficientes. Es de suponer que en el futuro se logren nuevos desarrollos teóricos que permitan optimizar aun más estos modelos.
- La identificación difusa de sistemas basada en la clusterización implica la formación de los clusters en el espacio de los datos y la posterior traducción de estos en reglas tales que representen con exactitud el sistema a identificar. En este trabajo se utilizan las técnicas de proyección para lograr dicha “traducción”, las que han dado muy buenos resultados como se ha podido comprobar.
- La aplicación del algoritmo de reducción de la base de reglas implementado para eliminar las redundancias, da prueba de la efectividad que tienen en los modelos difusos, debido a que disminuye ostensiblemente el gasto computacional sin provocar considerables variaciones en los modelos obtenidos.

-
- La utilización de los criterios de medidas de validez implementados son útiles en la determinación del número de clusters “óptimo”, y por tanto, en la obtención de modelos difusos sencillos con buenas capacidades de aproximación.
 - El empleo de modelos Takagi-Sugeno con consecuentes lineales es una vía factible en la identificación de sistemas complejos no lineales, dado que permiten obtener modelos simples, eficientes y con un número de reglas reducido.
 - Una contribución muy importante de este trabajo lo constituye la integración de varios algoritmos de clusterización implementados en otras tesis y métodos de simplificación de sistemas difusos en una biblioteca de funciones en Matlab. Esta herramienta puede ser utilizada en instituciones docentes y científicas para la aplicación de estas técnicas en la identificación de sistemas
 - Este módulo de funciones tiene como objetivo generar un modelo difuso suficientemente exacto y reducido a la vez. Su efectividad se probó en varios ejemplos no lineales de gran complejidad donde se pudo apreciar la efectividad del procedimiento. El funcionamiento y las características de los modelos difusos resultantes en esta tesis son similares o mejores, si se comparan con modelos obtenidos por otros investigadores.

RECOMENDACIONES

Con vistas a trabajos de investigación futuros en esta temática, se recomienda:

- Que se continúe el estudio e implementación de otros algoritmos de reducción de la base de reglas y otros métodos modelación difusa.
- Que se investigue en la obtención del número de clusters de manera automática, lo cual ampliaría el rango de aplicación de nuestra biblioteca de funciones.
- Que se incluya en el programa de la carrera, asignaturas relacionadas con la inteligencia artificial
- Crear grupos investigativos con vista a incentivar el interés sobre el tema de control inteligente.

REFERENCIAS BIBLIOGRÁFICAS

- Abonyi, J. (2003) *Fuzzy Model Identification for Control*, Birkhauser Boston.
- Alfonso, M. A (2002). *Algoritmos de clasificación difusa de datos*. Trabajo de diploma, Dpto. de Automática y Sist. Computacionales, Fac. Ing. Eléctrica, Universidad Central "Marta Abreu" de Las Villas, Santa Clara, Cuba.
- Babuska R. y H. Verbruggen (1997). Constructing fuzzy models by product space clustering. En: *Fuzzy Model Identification: Selected Approaches*, pp 53-90, H. Hellendoorn y D. Driankov (eds), Springer, Berlin, Germany.
- Babuska, R. (1998) *Fuzzy modeling for control*, Kluwer Academic Publishers, Boston
- Baraldi, A. y P. Blonda, (1999). "A Survey of Fuzzy Clustering Algorithms for Pattern Recognition", *IEEE Trans. Systems., Man and Cybernetics-Part B: Cybernetics*, **29**. (6), Diciembre.
- Bello, R. (2004). Conferencia #12: Sistemas de inferencia borrosos. Disponible en http://www.mfc.uclv.edu.cu/ia/Inteligencia_Artificial/Conferencia/Conferencia_No_12_Sistemas_de_inferencia_borrosa.htm, accedido el 10/4/2004.
- Bezdek, J. (1981). *Pattern recognition with Fuzzy Objective Function*. Plenum Press, New York.
- Box, G. y G. Jenkins (1970). *Time Series Análisis, Forecasting and Control*. Holden Day, San Francisco, CA.
- Bruno, N (1999), *Sistemas de inferencia difusos basados en redes neuroadaptativas*. Technical Report. Serie C. Facultad de Matemática, Astronomía y Física, Universidad Nacional de Córdoba, Argentina.

- Carbonell, P., J. Díez y L. Navarro (2000). “Aplicaciones de técnicas de modelos en sistemas complejos”, *Revista Iberoamericana de Inteligencia Artificial*, **9** (January)
- Castro, J. (1995). Fuzzy logic controllers are universal approximators. *IEEE Transactions on Systems, Man and Cybernetics*: 629-635.
- Díez, J., J. Navarro y A. Sala (2002). Algoritmos de clustering en la identificación de modelos borrosos. XXIII Jornadas de Automática, La Laguna, Santa Cruz de Tenerife. Disponible en http://www.cea-ifac.es/actividades/jornadas/XXIII/documentos/ja02_002.pdf, accedido el 25/03/2004
- Fernández, J. A (2003). *Algoritmos de clusterización en línea*. Trabajo de Diploma, Dpto. de Automática y Sist. Computacionales, Fac. Ing. Eléctrica, Universidad Central "Marta Abreu" de Las Villas, Santa Clara, Cuba.
- Gath, I. y A. Geva (1989). “Unsupervised optimal fuzzy clustering”. *IEEE Trans. Pattern Analysis and Machine Intelligence* **7**: 773–781.
- Guedalia, I., M. London y M. Werman. (1998). “An on line agglomerative clustering method for non-stationary data”, *Neural Computation*, **11**(2): 521-540
- Gustafson, D. y W. Kessel (1979). Fuzzy clustering with a fuzzy covariance matrix. En: *Proc. IEEE CDC*, San Diego, CA, USA: 761–766.
- Halkidi, M., Y. Batistakis y M. Vazirgiannis (2002). “Clustering Validity Checking Methods: Part II”. *SIGMOD Record*, **31**(3): 19-27
- Hathaway, R. y J. Bezdek (1993). “Switching regression models and fuzzy clustering”. *IEEE Trans. Fuzzy Systems*, **1**(3): 195-204.
- Krishnapuram, R. y C-P Freg (1992). “Fitting an unknown number of lines and planes to image data through compatible cluster merging”. *Pattern Recognition*, **25**(4): 385–400.
- Krishnapuram, R. y J. Keller (1993). A possibilistic approach to clustering. *IEEE Trans. Fuzzy Systems*, **1**(2): 98-110.
- Kukolj D.y E. Levi (2004). “Identification of Complex Systems Base don Neural and Takagi-Sugeno Fuzzy Model”, *IEEE, Trans. Systems., Man and Cybernetics*, **34** (1). February.

- Martínez, B. (2000) *Identificación difusa de un biorreactor*. Tesis de Maestría en Automática, Dpto. de Automática y Sist. Computacionales, Fac. de Ing. Eléctrica, Universidad Central "Marta Abreu" de Las Villas, Santa Clara, Cuba.
- Martínez, B.L. y F. Herrera (2003). "Estrategia para la Construcción de Modelos Difusos utilizando Clustering y Transformación Ortogonal". En: *Taller Internacional sobre Control por Computadora de Sistemas Industriales: Teoría y Aplicación, X Convención Internacional INFORMATICA 2003*. Ciudad de La Habana
- Narendra S.K. y K.Parthasarathy, (1990) "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, **1**(Jan): 4-27.
- Salehfar, H (2000). A systematic approach to linguistic fuzzy modeling based on input-output data. *Proceedings of the 2000 Winter Simulation Conference*. USA
- Setnes, M. (1995). *Fuzzy Rule Base Simplification Using Similarity Measures*. M.Sc. thesis, Delft University of Technology, Delft, The Netherlands.
- Song, Q. y N. Kasabov (2001). ECM- A Novel On-Line Evolving Clustering, Method and its Applications. Proceedings of the Fifth Biannual Conference on Artificial Neural Networks and Expert Systems (ANNES2001), Dunedin, New Zealand: 87-92.
- Sugeno, M., Kang, G.T. (1988). "Structure identification of fuzzy model". *Fuzzy Sets and Systems*, **28** (1): 15-33.
- Takagi, T. y M. Sugeno. (1985). "Fuzzy identification of systems and its applications to modelling and control", *IEEE Transaction. On Systems, Man, and Cybern.*, **15** (1): 116-132.

ANEXOS

Comparación de estructura y exactitud para el problema del horno de gas de Box y Jenkins

Model - Author	Number of inputs	Number of rules	Number of parameters	MSE
Box and Jenkins (70) [33]	6	-	10	0.202
Tong (80) [36]	2	19	-	0.469
Pedrycz (84) [37]	2	81	-	0.320
Xu and Lu (87) [38]	2	25	-	0.328
Peng and Wang (88) [39]	2	49	-	0.300
Sugeno and Tanaka (91) [23]	6	2	17	0.068
Yoshinari et al. (93) [40]	2	6	-	0.298
Sugeno and Yasukawa (93) [21]	3	6	96	0.190
Linear model [21]	5	-	-	0.193
Lin and Cunningham (95) [41]	5	4	354	0.071
Nie (95) [42]	2	35	-	0.197
Pedrycz et al. (95) [43]	3	8	-	0.395
Wang and Langari (95) [44]	6	2	-	0.066
Wang and Langari (96) [45]	2	5	-	0.172
Delgado et al. (97) [34]	2	4	-	0.168
Kim et al. (SI fuzzy) (97) [46]	6	2	38	0.055
Joo et al. (97) [47]	2	6	-	0.166
Lin et al. (97) [48]	7	12	684	0.157
Kim et al. (with uncorrelation) (98) [49]	6	2	110	0.048
Chen et al (98) [30]	2	3	15	0.268
Wang and Rong (99) [50]	2	29	116	0.137
Lo and Yang (99) [35]	6	2	26	0.062
Kang et al. (00) [51]	2	5	45	0.161
<i>This model</i>	2	2	12	0.129

TABLE II
 BOX-JENKINS' GAS FURNACE PROBLEM: COMPARISON OF STRUCTURE AND ACCURACY OF AVAILABLE MODELS AND THE PROPOSED
 ADAPTIVE NEURO-FUZZY MODEL.

Model - Author	Number of inputs	Number of rules	Number of parameters	MSE
Box and Jenkins (70) [33]	6	-	10	0.202
Tong (80) [36]	2	19	-	0.469
Pedrycz (84) [37]	2	81	-	0.320
Xu and Lu (87) [38]	2	25	-	0.328
Peng and Wang (88) [39]	2	49	-	0.300
Sugeno and Tanaka (91) [23]	6	2	17	0.068
Yoshinari et al. (93) [40]	2	6	-	0.298
Sugeno and Yasukawa (93) [21]	3	6	96	0.190
Linear model [21]	5	-	-	0.193
Lin and Cunningham (95) [41]	5	4	354	0.071
Nie (95) [42]	2	35	-	0.197
Pedrycz et al. (95) [43]	3	8	-	0.395
Wang and Langari (95) [44]	6	2	-	0.066
Wang and Langari (96) [45]	2	5	-	0.172
Delgado et al. (97) [34]	2	4	-	0.168
Kim et al. (SI fuzzy) (97) [46]	6	2	38	0.055
Joo et al. (97) [47]	2	6	-	0.166
Lin et al. (97) [48]	7	12	684	0.157
Kim et al. (with uncorrelation) (98) [49]	6	2	110	0.048
Chen et al (98) [30]	2	3	15	0.268
Wang and Rong (99) [50]	2	29	116	0.137
Lo and Yang (99) [35]	6	2	26	0.062
Kang et al. (00) [51]	2	5	45	0.161
<i>This model</i>	2	2	12	0.129