# Actas del

# VII CONGRESO ESPAÑOL SOBRE TECNOLOGÍAS Y LÓGICA FUZZY

Tarragona, 15-18 de Septiembre de 1997

Organizado por:
Dep. d'Enginyeria Informàtica (Univ. Rovira i Virgili)

Promovido por:
La Asociación Española sobre Tecnologías y Lógica Fuzzy

**Luis Magdalena**

Estas actas han sido impresas mediante el proceso de photo-offset, utilizando los manuscritos suministrados por los autores de los diferentes artículos. Estos manuscritos han sido elaborados mediantes diferentes editores de texto, por lo que la reproducción no es completamente uniforme. A fin de asegurar la rápida publicación, esta disparidad no ha podido ser corregida. Es por ello que se ruega a los lectores excusen las deficiencias de la publicación.

Impreso en España

Publicado por:
Departament d'Enginyeria Informàtica
Universitat Rovira i Virgili
Ctra. Salou s/n
43006 Tarragona

Editado por:
Eduard Montseny

ISBN: 84-88693-98-2

Depósito legal: T - 1220/97

El Congreso Español sobre Tecnología y Lógica Fuzzy, con las seis ediciones anteriores, se ha consolidado como el foro de reunión y discusión más importante, a nivel nacional, de los investigadores que trabajan o utilizan la teoría y las técnicas fuzzy. Este volumen recopila las comunicaciones presentadas al VII Congreso celebrado en la Universidad Rovira i Virgili (Tarragona) del 15 al 18 de septiembre de 1997.

Hace más de 30 años que el Prof. Zadeh publicó el "histórico" trabajo en el que introducía los conceptos básicos de la teoría fuzzy. Si bien el Prof. Zadeh, cuando introdujo la teoría de los Fuzzy Sets, pertenecía a la Universidad de Berkeley (EUA), ha sido Japón el país que más iniciativas, tanto a nivel teórico como aplicado, ha desarrollado en este campo. También es el país que cuenta con mayor número de aplicaciones industriales, basadas en dicha teoría, y donde la etiqueta "basado en teoría fuzzy" es signo de alta calidad.

España es uno de los países pioneros en la investigación en Fuzzy Sets. Hoy en día se puede afirmar que estamos en el quinto lugar del mundo, atendiendo al número de trabajos presentados, por investigadores españoles, en congresos y publicados en revistas especializadas. No obstante, se ha de reconocer que las aplicaciones comerciales todavía son pocas, debiéndose hacer un esfuerzo en este sentido, a fin de no caer en el eufemismo de que aquí creamos teorías interesantes y otros países son los que las aplican y obtienen los beneficios comerciales de las mismas.

El VII Congreso ESTYLF'97, promovido por la Asociación Española de Tecnología y Lógica Fuzzy, tiene dos objetivos básicos. El primero consolidar e incrementar la colaboración entre investigadores teóricos y aplicados, los cuales emplean la teoría y las técnicas difusas en su investigación, como una herramienta útil, flexible y robusta para el logro de sus objetivos. El segundo, tan importante como el primero, abrir nuevas vías a la transferencia de dicha tecnología y a su aplicación, no sólo en la investigación en las empresas, sinó en equipamiento industrial.

Desde aquí queremos agradecer a los conferenciantes su esfuerzo e interés por colaborar en nuestro congreso, así como su contribución a la calidad del mismo.

También queremos expresar nuestro agradecimiento a los miembros del comité de programa, por haber realizado las revisiones de los trabajos con total prontitud y eficacia.

Así mismo quiero agradecer todos los consejos y la colaboración recibidos durante la organización del congreso, y muy especialmente a los miembros del comité organizador todos los esfuerzos realizados.

También queremos extender nuestra gratitud a la Asociación Española de Tecnología y Lógica Fuzzy, por la confianza demostrada al concedernos la organización de este séptimo congreso. Agradecer, así mismo, la colaboración de: Vicerectorado de Investigación de la Universidad Rovira i Virgili, Ayuntamiento de Tarragona, Cámara de Comercio de Tarragona, Caixa de Tarragona, Rotring, DGYCIT y CIRIT.

Eduard Montseny

# Presidente

Eduard Montseny

# Comité de Programa

Josep Aguilar
Pedro Albertos
Claudi Alsina
Jaime Gil Aluja
Senen Barro
Juan Luís Castro
Miguel Delgado
Francesc Esteva
José Mª Fuertes
Pere García
Mª Ángeles Gil
Pedro Gil
Lluís Godo
Antonio González
Julio Gutiérrez
Joan Jacas
Ramón L. Mántaras
Luís Magdalena
Roque Marín
Gaspar Mayor
Aníbal Ollero
Enric Plaza
Alberto Sanfeliu
Pilar Sobrevilla
Alejandro Sobrino
Antonio Terceño
Vicenç Torra
Enric Trillas
Llorenç Valverde
José Luís Verdegay
Amparo Vila
Juan José Villanueva

# Comité Organizador

Susana Álvarez
Carlos García-Barroso
Albert Larré
Pere Millán
Antonio Moreno
Domènec Puig
David Riaño
Pilar Sobrevilla

# Entidades Colaboradoras

Universitat Rovira i Virgili
Ayuntamiento de Tarragona
Cámara de Comercio de Tarragona
Caixa de Tarragona
Rotring España
DGYCIT
CIRIT

# Índice


## SESIÓN: TEORÍA I

Aproximaciones superior e inferior de conjuntos borrosos ..... 1
*D.Boixader, J.Jacas, J.Recasens*

Antónimos y modificadores lingüísticos ..... 7
*A.Rodríguez, E.Trillas, M.J.Herrero*

Un enfoque lógico para calcular el grado de cumplimiento de sentencias con cuantificadores lingüísticos ..... 15
*M.Delgado, D.Sánchez, M.A.Vila*

On the Semantics of Fuzzy Statements Based on Possibilistic Constraints ..... 21
*F.Esteva, P.García, L.Godo*

## SESIÓN: TEORÍA II

Un modelo matemático del "y" en la lógica borrosa ..... 27
*E.Trillas, A.Pradera, S.Cubillo*

Medias casi lineales ponderadas extendidas ..... 33
*T.Calvo, P.Canet, M.Carbonell*

Generación de funciones de agregación extendidas ..... 39
*M.Mas, G.Mayor, J.Suñer, J.Torrens*

Toma de decisiones colectivas con preferencias difusas recíprocas ..... 45
*J.L.García, B.Llamazares*

## SESIÓN: APLICACIONES I

Una solución al problema de la división relacional difusa ..... 51
*J.Galindo, J.M.Medina, M.C.Aranda*

Redes de restricciones temporales disyuntivas borrosas ..... 57
*I.Navarrete, R.Marín, M.Balsa*

Restricciones de diferente importancia en problemas de satisfacción de restricciones difusas ..... 65
*V.Torra*

Principios muy generales para la educación en lógica fuzzy. Algunas reflexiones personales ..... 71
*A.Sobrino*

## SESIÓN: CONTROL I

Software de compilación, simulación y emulación borrosa ..... 77
*G.Aranguren, M.Rodríguez, J.López de Arroyabe, N.Altarriba*

Influencia de la partición de entrada en la ley de control fuzzy: un ejemplo analítico ..... 83
*M.Santos, S.Dormido, J.M. de la Cruz*

Controlador borroso multivariable cooperativo-corrector ..... 89
*C.Rivero, P.Lasso, F.Monasterio*

Controlador fuzzy para la evitación de obstáculos en un robot móvil no holónomo ..... 95
*J.P.Uribe, J.Urzelai*

## SESIÓN: MODELIZACIÓN

Descomposición jerárquica de controladores borrosos multiobjetivo basada en metaconocimiento ..... 103
*L.Magdalena, F.Monasterio, C.Rivero*

Nuevos métodos de razonamiento en sistema de clasificación basados en reglas difusas..... 109
*O.Cordón, M.J. del Jesús, F.Herrera*

Hibridación de técnicas para la generación y ajuste de reglas en el modelado difuso... 115
*M.Delgado, A.F.Gómez, J.Gómez, H.Martínez*

Modelado a partir de ejemplos imprecisos mediante algoritmos GA-P..... 121
*L.Sánchez, I.Couso*

## SESIÓN: APLICACIONES II

Una metodología para el desarrollo de SBC..... 127
*J.L.Castro, J.M.Zurita, J.J.Castro, A.Espin*

Sistema experto fuzzy para la evaluación y entrenamiento físico ..... 133
*N.Juan, M.Santos*

Múltiple clasificación en nivel de medida aplicada al reconocimiento de textos..... 141
*A.Sanz, J.Echanove, J.R.González de Mendivil, J.R.Garitagoitia*

Cooperación mediante comunicación de conceptos borrosos ..... 147
*V.Matellán, J.M.Molina, L.Sommaruga*

Utilización de técnicas difusas para la detección de lesiones pequeñas de esclerosis múltiple..... 153
*F.X.Aymerich, P.Sobrevilla, J.Gili, E.Montseny*

## SESIÓN: TEORÍA III

Una primera aproximación a la unificación fuzzy vía relaciones de compatibilidad .... 159
*T.Alsinet*

Una medida de la bondad de la inferencia usando medidas difusas ..... 165
*M.C.Garrido, J.M.Cadenas, A.Ruiz*

Test del algoritmo fuzzy-minimals con datos reales..... 171
*A.Flores, J.M.Cadenas, F.Martín*

La envolvente probabilística de variables aleatorias difusas..... 177
*I.Couso, P.Gil*

## SESIÓN: REDES NEURONALES DIFUSAS

Nuevos aspectos de extracción de reglas difusas usando redes neuronales ..... 183
*J.M.Benítez, A.Blanco, M.Delgado, I.Requena*

Aprendizaje a partir de información con ruido en los modelos difuso-neuronales FasArt y FasBack..... 189
*J.M.Cano, Y.Dimitriadis, J.López*

Entrenamiento de redes de base radial realimentadas con filtros FIR..... 195
*L.Boquete, R.Barea, R.García, M.Mazo*

Predicción de la demanda por técnicas Neuro-Fuzzy basada en estructuras jerárquicas ARS (Autonomous Response System) ..... 201
*I.Escoda, A.Sanz, A.Herms*

## SESIÓN: TEORÍA IV

Medidas de desigualdad difusas de tipo Gastwirth: definición y estudio de condiciones de existencia ..... 207
*H.López, M.López*

Medidas de información de un conjunto difuso ..... 213
*C.Bertoluzza, A.Bodini*

Fusión de atributos con técnicas fuzzy en data mining ..... 217
*D.Nettleton, K.Gibert*

Definición de parámetros estadísticos bidimensionales basados en la integral de Choquet ..... 221
*M.J.Bolaños, F.Reche*

## SESIÓN: CONTROL II

On Defuzzification with Continuous WOWA Operators ..... 227
*V.Torra, L.Godo*

Control borroso adaptativo de un convertidor DC/AC conmutado ..... 233
*P.Carbonell*

Detección de fallos en cajas de engranajes mediante bases de reglas difusas ..... 239
*R.Ferreiro, X.C.Pardo, J.Vidal, V.Carneiro*

Controlador híbrido difuso-deslizante de motores de continua ..... 245
*E.Vidal, J.Calvente, R.Giral, R.Leyva, L.Martínez*

Seguimiento del punto de máxima potencia de un sistema fotovoltaico mediante control difuso ..... 251
*S.Gomáriz, J.A.Guerrero, F.Guinjoan*

## SESIÓN: HARDWARE

Xfuzzy 2.0: un entorno de descripción, verificación y síntesis de sistemas basados en lógica difusa ..... 257
*D.López, F.J.Moreno, C.J.Jiménez, S.Sánchez, A.Barriga*

Entorno multicontexto y metacontrol en microprocesadores borrosos ..... 263
*P.E.Bernad*

Xfuzzy 2.0: herramientas de síntesis de controladores difusos ..... 269
*C.J.Jiménez, D.López, E.Lago, M.A.Hinojosa, A.Barriga, S.Sánchez*

## SESIÓN: TEORÍA V

Operadores de consecuencias borrosos, su coherencia y generación de relaciones..... 275
*J.Elorza, P.Burillo*

μ-T-condicionalización de una relación difusa. Medida de μ-T-condicionalidad ..... 281
*L.Garmendia*

La λ-desviación cuadrática media para variables aleatorias difusas en poblaciones finitas ..... 287
*A.Lubiano, A.Colubi, M.A.Gil*

Completitud y dirección en ciertos sistemas relacionales difusos ..... 293
*P.Burillo, R.Fuentes, L.González*

## SESIÓN: TEORÍA VI

Divergencias entre conjuntos y entre particiones difusas ..... 299
*S.Montes, P.Gil*

Un modelo multiobjetivo fuzzy para el control medioambiental agrícola ..... 305
*J.M.Cadenas, J.L.Verdegay*

El λ-cálculo-B..... 311
*D.Sánchez*

Satisfiability Testing in Finitely-Valued Logics..... 317
*F.Manyà, G.Escalada*

## III JORNADA SOBRE TRANSFERENCIA DE TECNOLOGÍA FUZZY

Algunas experiencias didácticas en el campo de la lógica difusa..... 323
*J.S.Domingo, E.Realp*

Principales cálculos sobre incertidumbre difusa a través del programa C.I.D..... 329
*J.Jiménez, S.Montes, M.Díez*

Xfuzzy 2.0: entorno de diseño para sistemas basados en lógica difusa..... 335
*D.López, S.Sánchez, A.Barriga*

# APROXIMACIONES SUPERIOR E INFERIOR DE CONJUNTOS BORROSOS

**D. Boixader**
Sec. Matemàtiques i Inf.
E.T.S.A.V.
Pere Serra 1-15,
08190 Sant Cugat.
e-mail: boixader@ea.upc.es

**J. Jacas**
Sec. Matemàtiques i Inf..
E.T.S.A.B.
Avda. Diagonal 649,
08028 Barcelona.
e-mail: jacas@ea.upc.es

**J. Recasens**
Sec. Matemàtiques i Inf.
E.T.S.A.V.
Pere Serra 1-15,
08190 Sant Cugat.
e-mail: recasens@ea.upc.es

## Resumen

Se estudian las aproximaciones superior e inferior de un conjunto borroso respecto a un operador de indistinguibilidad y se analiza su relación con *rough sets* borrosos y lógica modal borrosa.

Palabras clave: operador de indistinguibilidad, generador, conjunto observable, aproximación superior, aproximación inferior, *rough sets*, lógica modal, operador de clausura, topología borrosa.

## 1 INTRODUCCIÓN

Cuando se tiene definido un *operador de indistinguibilidad* $E$ en un conjunto $X$ (tales operadores fuzzyfican el concepto de relación de equivalencia), los únicos conjuntos borrosos *observables* respecto a $E$ son los generados por sus clases de equivalencia borrosas. A la hora de trabajar con un conjunto borroso $h$ es necesario aproximarlo por observables: la *aproximación superior* ($\phi_E h$) que se define como el menor conjunto borroso observable que contiene a $h$, y la *aproximación inferior* ($\psi_E h$) que es el mayor observable borroso contenido en $h$.

La aplicación $\phi_E$ ha sido ampliamente estudiada [5, 15, 16, 18, 19, 20, 21], pero no así $\psi_E$ básicamente por dos razones:

a) En la teoría de los *rough sets* [24] y en *lógica modal* [13] la aproximación inferior de $h$ se define por $\neg\phi_E\neg h$, si bien, en general, $\neg\phi_E\neg h$ no es un conjunto observable (sección 4).

b) No había una forma sencilla de calcular $\psi_E h$.

En este trabajo se da una fórmula sencilla para calcular $\psi_E h$ y se estudian sus propiedades (sección 4). También se recogen las propiedades más relevantes de $\phi_E$ y su restricción a $X$ se interpreta como la proyección canónica $\phi_{E|X} : X \to X/E$ (sección 3). En la sección 5 se comenta la conexión con los *rough sets* y la lógica modal.

## 2 PRELIMINARES

En esta sección se recuerdan algunos conceptos relacionados con los operadores de indistinguibilidad y se demuestran algunos lemas que se usarán más tarde.

Dada una t-norma continua, $T$, $\hat{T}$ denotará su *quasi-inversa*, definida por

$$\hat{T}(x|y) = \sup\{\alpha \in [0,1] \mid T(x,\alpha) \leq y\} \quad \forall x, y \text{ of } [0,1].$$

**Nota.** $([0,1], \leq, T)$ es un retículo residuado y $\hat{T}$ es la correspondiente *residuación* respecto a la t-norma $T$. En el contexto lógico, $\hat{T}$ se interpreta como la implicación $\underset{T}{\to}$ basada en la t-norma $T$.

**Lema 2.1.** Dada una t-norma continua $T$, para todo $x, y, z$ de $[0,1]$ se satisface

**2.1.1.** $\text{Min}(\hat{T}(x|y), \hat{T}(y|x)) = \hat{T}(\text{Max}(x,y) \mid \text{Min}(x,y))$

**2.1.2.** $\hat{T}(\hat{T}(x|z) \mid \hat{T}(y|z)) \geq \hat{T}(y|x)$

**2.1.3.** $\hat{T}(x|y) = \sup\{\alpha \in [0,1] \mid \hat{T}(\alpha \mid y) \geq x\}$

Demostración.

**2.1.1.** [27]

**2.1.2.** [2]

**2.1.3.** Si $\alpha \in [0,1]$ es tal que $T(\alpha, x) \leq y$, entonces $\hat{T}(\alpha|y) \geq x$ y por tanto $\hat{T}(x|y) \leq \sup\{\alpha \in [0,1] \mid \hat{T}(\alpha|y) \geq x\}$.

Si $\alpha \in [0,1]$ es tal que $\hat{T}(\alpha|y) \geq x$, entonces $\sup\{\beta \in [0,1] \mid T(\beta, \alpha) \leq y\} \geq x$ y por tanto $T(x, \alpha) \leq y$, y $\sup\{\alpha \in [0,1] \mid \hat{T}(\alpha|y) \geq x\} \leq \hat{T}(x|y)$.

**Definición 2.2.** Dada una t-norma $T$, un operador de $T$-indistinguibilidad $E$ en un conjunto $X$ es una relación borrosa en $X$ que satisface

(1) $E(x,x) = 1 \ \forall x \in X$ *(reflexividad)*,

(2) $E(x,y) = E(y,x) \ \forall x, y \in X$ *(simetría)*,

(3) $T(E(x,y), E(y,z)) \leq E(x,z) \ \forall x, y, z \in X$ (*T-transitividad*).

En [27] se demuestra que todo operador de indistinguibilidad se puede construir a partir de una familia de conjuntos borrosos.

**Lema 2.3.** Dado un subconjunto borroso $h$ de un conjunto $X$, la relación borrosa $E_h$ definida por

$$E_h(x,y) = \hat{T}\left(\text{Max}(h(x), h(y)) \mid \text{Min}(h(x), h(y))\right)$$

es un operador de indistinguibilidad en $X$.

**Teorema de Representación 2.4.** Una relación borrosa $E$ en un conjunto $X$ es un operador de indistinguibilidad si, y sólo si, existe una familia $\{h_i\}_{i \in I}$ de subconjuntos borrosos de $E$ tal que

$$E = \inf_{i \in I} E_{h_i}.$$

Demostración. [27].

El teorema anterior sugiere la siguiente definición [14].

**Definición 2.5.** Dado un operador de indistinguibilidad $E$ en $X$, un *generador* de $E$ es un subconjunto borroso de $X$ que pertenece a alguna familia que genere $E$ en el sentido del teorema anterior.

**Lema 2.6.** Si $H_E$ denota el conjunto de generadores de $E$, $h \in H_E$ si, y sólo si, $E_h \geq E$.

Demostración. Trivial.

**Lema 2.7.** Dado un operador de indistinguibilidad $E$ en un conjunto $X$ y un elemento $x \in X$, el subconjunto borroso $h_x$ de $X$ definido por $h_x(y) = E(x,y) \ \forall y \in X$ es un generador de $E$.

Demostración. [27].

Recordemos la definición de topología borrosa y de operador de clausura borroso [23].

**Definición 2.8.** Un subconjunto $\tau$ de $[0,1]^X$ es una topología borrosa de $X$ si y sólo si, satisface

(1) Si $A \subset \tau$, entonces $\sup\{h | h \in A\} \in \tau$.

(2) Si $h, h' \in \tau$, entonces $h \wedge h' \in \tau$.

(3) Si $h_k \in [0,1]^X$ es un conjunto borroso constante (i.e., $h_k(x) = k \ \forall x \in X$), entonces $h_k \in \tau$.

**Definición 2.9.** Una aplicación $\phi$ de $[0,1]^X$ en $[0,1]^X$ es un operador de *clausura borroso*, si, y sólo si, para todo $h, h'$ de $[0,1]^X$ satisface

(1) $h \leq \phi(h)$,

(2) $\phi(h \vee h') = \phi(h) \vee \phi(h')$,

(3) $\phi^2(h) = \phi(h)$,

(4) $\phi(h_k) = h_k$ para cada borroso constante $h_k$.

Los elementos de una topología borrosa $\tau$ se llaman *abiertos borrosos* y, dada una *negación fuerte* $\varphi$ (i.e. una biyección decreciente y continua de [0,1] en [0,1] con $\varphi(\varphi(x)) = x \ \forall x \in [0,1]$), un subconjunto borroso $h$ se llama *cerrado* si, y sólo si, $\varphi \circ h$ es abierto.

Se puede demostrar fácilmente que, dado un operador de clausura $\phi$ en $X$ y una negación fuerte $\varphi$, los subconjuntos borrosos $h$ de $X$ tales que $\phi(h) = h$ son los cerrados de la topología borrosa $\tau$ definida por

$$\tau = \left\{h \in [0,1]^X \mid \phi(\varphi \circ h) = \varphi \circ h\right\}.$$

# 3 LA APROXIMACIÓN SUPERIOR

En esta sección se estudiará la aplicación $\phi_E$ que envía cada subconjunto borroso $h$ sobre el menor borroso $\phi_E(h)$ que lo contiene.

**Definición 3.1.** Sea $E$ un operador de $T$-indistinguibilidad en $X$. La aplicación

$\phi_E : [0,1]^X \longrightarrow [0,1]^X$ se define como

$$\phi_E(h)(x) = \sup_{y \in X} T\left(E(x,y), h(y)\right).$$

**Proposición 3.2.** [14,15] $\phi_E$ es un operador de clausura borroso.

**Corolario 3.3.** $H_E$ es el conjunto de cerrados de una topología borrosa.

**Proposición 3.3.** $h \leq h' \Rightarrow \phi_E(h) \leq \phi_E(h')$.

**Demostración.** Trivial debido a la monotonía de las t-normas.

**Definición 3.4.** $h \in [0,1]^X$ es *observable* respecto a $E$ si, y sólo si, $\phi_E h = h$.

**Teorema 3.5.** $h \in [0,1]^X$ es un generador de $E$ si, y sólo si, es observable.

Demostración. [14]

**Corolario 3.6.** $H_E$ es el conjunto de borrosos observables respecto a $E$.

**Proposición 3.7.** $\forall h \in [0,1]^X$, $\phi_E(h) = \bigwedge_{h' \in H_E} \{h \leq h'\}$.

Demostración. Sea $h' \in H_E$ con $h' \geq h$. Debido a la monotonía de $\phi_E$ (Proposición 3.3), $h' = \phi_E(h') \geq \phi_E(h) \in H_E$.

En [14] se demuestra que $H_E$ es un retículo completo. El siguiente teorema permite una demostración alternativa.

**Teorema 3.7.** (Tarski). [25]. El conjunto de puntos fijos de una aplicación monótona en un retículo completo $A$ es un subretículo completo de $A$.

**Corolario 3.8.** $H_E$ es un subretículo completo.

Demostración. Basta aplicar 3.7 a $[0,1]^X$ y $\phi_E$.

El conjunto $H_E$ ha sido extensamente estudiado y sus elementos han sido caracterizados como los *eigenvectors* [19] y generadores [14] de $E$, los puntos fijos de $\phi_E$, los estados lógicos asociados a $E$ [6] y sus conjuntos extensionales [21].

Los elementos de $H_E$ son los únicos subconjuntos borrosos de $X$ observables si se tiene en cuenta $E$, y $\phi_E$ es el modo en que $E$ distorsiona los borrosos de $X$. Esto se muestra en los siguientes resultados:

**Proposición 3.9.** $\forall x \in X\ \phi_E(\{x\}) = E(x,\cdot)$.

Demostración. [14]

**Corolario 3.10.** $\forall A \subset X\ \phi_E(A) = \bigvee_{x \in A} E(x,\cdot)$.

Demostración. [14]

**Corolario 3.11.** Sea $P(X)$ el conjunto de las partes de $X$. $\phi_E(P(X) - \{0\})$ es el semiretículo $H_E^C$ de $H_E$ generado por subfamilias de $\{E(x,\cdot)\}_{x \in X}$ via supremos.

$H_E$ se puede interpretar como el conjunto de subconjuntos borrosos del conjunto cociente $X/E$ ($H_E = [0,1]^{X/E}$) y $\phi_E : [0,1]^X \to [0,1]^{X/E}$ es la proyección canónica. Nótese que si $E$ es una relación crisp (i.e. una relación de equivalencia crisp), entonces $\phi_{E|X}$ es la proyección canónica crisp $\pi : X \to X/E$.

# 4 LA APROXIMACIÓN INFERIOR

En esta sección se estudiará la aplicación $\psi_E$ que envía cada subconjunto borroso $h$ el mayor subconjunto borroso $\psi_E(h)$ contenido en él.

**Definición 4.1.** Sea $E$ un operador de T-indistinguibilidad en $X$. La aplicación $\psi_E : [0,1]^X \to [0,1]^X$ se define como

$$\psi_E(h)(x) = \inf_{x \in X} \hat{T}\left(E(x,y) | h(y)\right)$$

**Proposición 4.2.** $\forall h \in [0,1]^X\ \psi_E(h) \leq h$.

Demostración. Trivial $\Big(h(x) = \hat{T}(E(x,x) \mid h(x))\Big)$.

**Proposición 4.3.** $h \leq h' \Rightarrow \psi_E(h) \leq \psi_E(h')$.

Demostración. Trivial, puesto que $\hat{T}$ es no decreciente respecto a la segunda variable.

**Teorema 4.4.** $\forall h \in [0,1]^X\ \psi_E(h) \in H_E$.

Demostración.

$$\hat{T}\left(\psi_E(h)(x_1) \mid \psi_E(h)(x_2)\right) =$$

$$= \hat{T}\left(\inf_{y \in X} \hat{T}\left(E(x_1,y)|h(y)\right) | \inf_{y \in X} \hat{T}\left(E(x_2,y)|h(y)\right)\right)$$

$$\overset{*}{=} \inf_{y \in X} \hat{T}\left(\inf_{z \in X} \hat{T}\left(E(x_1,z)|h(z)\right) | \hat{T}\left(E(x_2,y)|h(y)\right)\right)$$

$\overset{**}{\geq} \inf_{y\in X} \hat{T}\left(\hat{T}(E(x_1,y)|h(y))|\hat{T}(E(x_2,y)|h(y))\right) \geq$

$\overset{***}{\geq} \inf_{y\in X} \hat{T}\left(E(x_2,y)|E(x_1,y)\right)$

$\overset{****}{\geq} \inf_{y\in X} E(x_1,x_2) = E(x_1,x_2).$

De forma similar $\hat{T}(\psi_E(h)(x_2)|\psi_E(h)(x_1)) \geq E(x_1,x_2)$.

Y por 2.1.1.

$$E_{\psi_E(h)}(x_1,x_2) \geq E(x_1,x_2)$$

lo cual signfica que $\psi_E(h) \in H_E$.

$*$ $\hat{T}$ es continua por la derecha respecto a la segunda variable.

$**$ $\hat{T}$ es no creciente respecto a la primera.

$***$ 2.1.2.

$****$ $E(\cdot,y)$ es un generador de $E$ (Lema 2.7). Aplíquese Lema 2.6.

**Teorema 4.5.** $\psi_E(h) = h$ si, y sólo si, $h \in H_E$.

Demostración.

$\Rightarrow$) Por el Teorema 4.4, $h = \psi_E(h) \in H_E$.

$\Leftarrow$) Dado que $\psi_E(h) \leq h$ $\forall h \in [0,1]^X$ (Proposición 4.2), basta probar que $h \in H_E \Rightarrow \psi_E(h) \geq h$:

Sea $h \in H_E$. $h$ es un generador de $E$, y de 2.1.1 se sigue que $\hat{T}(h(x)|h(y)) \geq E(x,y)$.

2.1.3. implica que $\hat{T}(E(x,y)|h(y)) \geq h(x)$ y, puesto que es cierto para todo $x,y$ de $X$, $\psi_E(h) \geq h$.

**Proposición 4.6.** $\forall h \in [0,1]^X$, $\psi_E(h) = \bigvee_{h'\in H_E}\{h \geq h'\}$.

Demostración. Sea $h' \in H_E$ con $h' \leq h$. Debido a la monotonía de $\psi_E$ (Proposición 4.3), $h' = \psi_E(h') \leq \psi_E(h) \in H_E$.

**Proposición 4.7.** Para todo $h_1, h_2 \in [0,1]^X$ se cumple

**4.7.1.** $\psi_E(h_1) \wedge \psi_E(h_2) = \psi_E(h_1 \wedge h_2)$

**4.7.2.** $\psi_E^2 = \psi_E$

**4.7.3.** $\psi_E(h_k) = h_k$ si $h_k$ es un borroso constante.

Demostración.

**4.7.1.** Tivial.

**4.7.2.** Sigue del Teorema 4.5.

**4.7.3.** Los borrosos constantes son generadores.

Como se decía en la introducción, en diversos contextos, tales como la teoría de los *rough sets* o en lógica modal, la aproximación inferior de $h$ se define por $\varphi\circ\phi_E\circ\varphi(h)$ donde $\varphi$ es una negación fuerte. Sin embargo, el subconjunto borroso obtenido, en general no es observable. Una forma elegante de verlo es notando que $\phi_E\circ\varphi(h)$ es un cerrado y por tanto $\varphi\circ\phi_E\circ\varphi(h)$ es un abierto, pero no necesariamente un cerrado.

Tras recordar un teorema de caracterización de las negaciones fuertes de Trillas [26], se estudiará en qué casos, dado un conjunto $X$, se satisface la relación $\psi_E = \varphi\circ\phi_E\circ\varphi$ para cualquier T-indistinguibilidad en $X$.

**Definición 4.8.** Una *negación fuerte* es una biyección decreciente y continua $\varphi : [0,1] \longrightarrow [0,1]$ tal que $\varphi(0) = 1$, $\varphi(1) = 0$ y $\varphi^2 =$id.

**Teorema 4.9.** Una aplicación $\varphi : [0,1] \longrightarrow [0,1]$ es una negación fuerte si, y sólo si, existe una aplicación continua y decreciente $t : [0,1] \longrightarrow \mathbb{R}^+$ con $t(1) = 0$ tal que

$$\varphi(x) = t^{-1}\left(t(0) - t(x)\right) \quad \forall x \in [0,1].$$

Demostración. [26].

Una aplicación como la del teorema anterior se llama *generador* de la negación.

El teorema 4.9 tiene la siguiente interpretación en el contexto de las MV-álgebras:

**Corolario 4.10.** $\varphi$ es una negación fuerte si, y sólo si, existe una t-norma arquimediana no estricta $T$ con generador $t$ [22] tal que $\varphi(x) = x \underset{T}{\rightarrow} 0$.

Demostración. $x \underset{T}{\rightarrow} 0 = \hat{T}(x|0) = t^{-1}(t(0) - t(x))$.

**Teorema 4.11.** Sea $T$ una t-conorma continua y $E$ un operador de T-indistinguibilidad en $X$. $\varphi\phi_E\varphi(h) = \psi_E(h)$ para todo subconjunto borroso $h$ de $X$ si, y sólo si, $T$ es una t-norma arquimediana no estricta con el mismo generador que $\varphi$ o $E(x,y) = 1$ ó $0$ para todo $x,y \in X$.

Demostración.

$\Leftarrow$ Sea $t$ el generador de $T$ y $\varphi$.

$\psi_E(h)(x) = \inf_{y\in X} \hat{T}\left(E(x,y)|h(y)\right) =$

$= \inf_{y\in X} t^{[-1]}\left(t(h(y)) - t\left(E(x,y)\right)\right) =$

$$= \inf_{y\in X} t^{[-1]}\left(t(0) - (t(E(x,y) + t(0) - t(h(y))))\right) =$$
$$= \inf_{y\in X} t^{[-1]}\left(t(0) - t\left(t^{[-1]}[tE(x,y)+\right.\right.$$
$$\left.\left.+tt^{[-1]}(t(0) - t(h(y)))]\right)\right) =$$
$$= \inf_{y\in X} \varphi T(E(x,y), \varphi(h(y))) =$$
$$= \varphi \sup_{y\in X} T\left(E(x,y), \varphi(h(y))\right) =$$
$$= \varphi\phi_E\varphi(h)(x).$$

$\Rightarrow$ Supongamos primero que $T$ es una t-norma arquimediana. Sean $x, y \in X$ tales que $E(x,y) \neq 0, 1$, y consideremos el subconjunto borroso $h_x = E(x,\cdot)$. $h_x \in H_E$ (lema 2.7) Vamos a demostrar que $\psi_E h_x \neq \varphi\phi_E\varphi h_x$, o lo que es equivalente $\varphi\psi_E h_x \neq \phi_E\varphi h_x$:

Puesto que $\phi_E\varphi h_x \in H_E$, para probar la última desigualdad basta ver que $\varphi\psi_E h_x \notin H_E$. Pero como $h_x \in H_E$, esto último equivale a $\varphi h_x \notin H_E$:

$$E_{\varphi h}(x,y) =$$
$$= \mathrm{Min}\left(\hat{T}(\varphi E(x,x) \mid \varphi E(x,y)), \hat{T}(\varphi E(x,y), \varphi E(x,x))\right)$$
$$= \hat{T}(\varphi E(x,y)|0) = 0 \leq E(x,y).$$

El resultado es fácil de comprobar para la t-norma Mínimo puesto que $\widehat{\mathrm{Min}}(x|0) = 0\ \forall x \in [0,1]$ y se puede extender a sumas ordinales.

# 5 ROUGH SETS Y LÓGICA MODAL

Los *rough sets* fueron propuestos por Pawlak [24] para tratar la *granularidad* generada por una partición. Dado un conjunto crisp $A \subset X$ y una relación de equivalencia $E$ en $X$, el crisp rough set asociado a $A$ se define (con las notaciones de este trabajo) por $(\phi_E A, \neg\phi_E\neg(A))$ donde $\neg A$ denota el complemento de $A$. Gracias al teorema 4.11 esto es equivalente a $(\phi_E(A), \psi_E(A))$. En diversos trabajos [8] se ha fuzzyficado el concepto de *rough set* considerando subconjuntos borrosos y procurando hallar sus aproximaciones respecto relaciones de equivalencia crisp (*rough fuzzy sets*) o aproximando conjuntos crisp por borrosos teniendo en cuenta un operador de indistinguibilidad (*fuzzy rough sets*). En ambos casos, la aproximación inferior se define como $\varphi\phi_E\varphi h$. Debido al teorema 4.11 parece aconsejable sustituirla por $\psi_E h$. En este contexto, un concepto que podría llamarse *fuzzy rough fuzzy set* es el par $(\phi_E h, \psi_E h)$, donde $h$ es un subconjunto borroso y $E$ un operador de indistinguibilidad.

Las aproximaciones superior e inferior de un borroso $h$ a veces se llaman la posibilidad y la necesidad respectivamente de $h$. Ambos conceptos son típicos de la lógica modal de modo que existe una relación estrecha entre esta lógica y los *rough sets*.

Vamos a definir un modelo Kripke borroso.

**Definición 5.1.** Un marco borroso (fuzzy frame) es un par $\mathcal{F} = (X, E)$ donde $X$ es un conjunto crisp y $E$ una relación borrosa en $X$.

**Definición 5.2.** Un modelo borroso $\mathcal{M} = (X, E, \mu)$ es una terna donde $(X, E)$ es un marco borroso y $\mu$ asigna a cada proposición atómica $p$ un subconjunto borroso $\mu_p$ de $X$. Así se ha fuzzyficado la relación de accesibilidad $E$ entre mundos (i.e. elementos de $X$) y la valuación. De este modo una proposición $p$ no es cierta o falsa en un mundo $x \in X$, sino que tiene un grado de verdad $\mu_p(x)$.

En un modelo de Kripke crisp se dice que una proposición $p$ es *posible* en un mundo $x \in X$ sii existe un mundo $y \in X$ tal que $x$ esta relacionado con $y$ y $p$ es cierta en $y$. Si la conjunción se modeliza mediante una t-norma $T$, la fuzzyficación de la posibilidad de $p$ en $x$ es

**Definición 5.3.** El grado de posibilidad de $p$ en el mundo $x \in X$ es

$$\phi_E\mu_p(x) = \sup_{y\in X} T\left(E(x,y), \mu_p(y)\right).$$

En el caso crisp, $p$ es necesario en el mundo $x \in X$ si, y sólo si, para todos los mundos $y \in Y$, $x$ relacionado con $y$ implica $p$ cierto en $y$.

**Definición 5.4.** El grado de necesidad de $p$ en el mundo $x \in X$ es

$$\psi_E\mu_p(x) = \inf_{y\in X} \hat{T}\left(E(x,y)|\mu_p(y)\right).$$

$(\phi_E\mu_p, \psi_E\mu_p)$ es un fuzzy rough set en el conjunto de mundos posibles de un modelo de lógica modal.

# 6 CONCLUSIONES

En este trabajo se han estudiado las aproximaciones superior e inferior de un conjunto borroso respecto a un operador de T-indistinguibilidad. Ambos conceptos son duales respecto a una negación fuerte sólo si la t-norma es arquimediana no estricta.

Se ha puesto de manifiesto su relación con la teoría de rough sets y la lógica modal.

# REFERENCIAS


[1] C. Alsina, J.L. Castro, E. Trillas (1995), On the characterization of $S$ and $R$ Implications. *Proc. IFSA'95.* Sao Paulo, 317-319.

[2] D. Boixader, J. Jacas (1995), Generators and Dual T-indistinguishabilities, in Fuzzy Logic and

Soft Computing, B. Bouchon et altri eds. World Scientific. Singapore, 283-291.

[3] D. Boixader, J. Jacas (1996), T-indistinguishability Operators and Approximate Reasoning via CRI, *Linz'96*, 14-18.

[4] D. Boixader, J. Jacas, J. Recasens (1996), Hacia una semántica de la defuzzyficación. Actas ESTYL'96, Oviedo, 41-45.

[5] D. Boixader, J. Jacas, J. Recasens (1997), Searching for meaning of defuzzyfication, FUZZ'IEEE97. Barcelona.

[6] J.L. Castro, E. Trillas, S. Cubillo (1994), On Consequence in Approximate Reasoning. *Journal of Applied Non-Classical Logics* 4, 91-103.

[7] A. Di Nola, S. Sessa, W. Pedrycz, E. Sanchez (1989), *Fuzzy Relation Equations and their Applications to Knowledge Engineering.* Kluwer Academic Publishers, Dordrecht.

[8] D. Dubois, H. Prade (1994), Similarity-Based Approximate Reasoning, en: *Computational Intelligence Imitating Life.* Zurada, J.M., Marles, R.J., Robinson, C.J. eds. IEEE Press, New York, 69-80.

[9] F. Esteva (1981), Negaciones en la teoría de conjuntos difusos. *Stochastica* **5**, 33-44.

[10] F. Esteva, X. Domingo (1980), Sobre funciones de negación en [0,1]. *Stochastica* **4**, 141-165.

[11] F. Esteva, P. García, L. Godo (1994), Relating and Extending Semantical Approaches to Possibilistic Reasoning. *IJAR*, **10**, 311-344.

[12] F. Esteva, E. Trillas, X. Domingo (1981), Weak and strong negations for Fuzzy Sets Theory, *Proc. XI ISMVL*, Oklahoma. 23-26.

[13] R. Goldblatt, Logics of time and computation. *CSLI Lecture Notes* **7**.

[14] J. Jacas (1989), On the Generators of a T-indistinguishability Operator. *Stochastica* **12**, 49-63.

[15] J. Jacas (1993), Fuzzy topologies induced by S-metrics. *The Journal of Fuzzy Mathematics* **1**, 173-191.

[16] J. Jacas, J. Recasens (1992) Eigenvectors and generators of fuzzy relations. *Proc. IEEE International Conference on Fuzzy Systems.* San Diego, 687-694.

[17] J. Jacas, J. Recasens (1993) Fuzzy numbers and equality relations. *Proc. IEEE International Conference on Fuzzy Systems.* San Francisco.

[18] J. Jacas, J. Recasens (1994) Fixed points and generators of fuzzy relations. *J. Math. Anal. Appl.* **186** 21-29.

[19] J. Jacas, J. Recasens (1995) Fuzzy T-transitive relations: eigenvectors and generators. *Fuzzy Sets & Systems* **72** 147-154.

[20] J. Jacas, L. Valverde (1990) On Fuzzy Relations, Metrics and Cluster Analysis. J.L. Verdegay, M. Delgado, Eds. *Approximate Reasoning Tools for Artificial Intelligence* (Verlag TUV, Rheinland) 15R, 96.

[21] F. Klawonn, J.L. Castro (1995) Similarity in Fuzzy Reasoning, *Mathware & Soft Computing* **2**, 197-228.

[22] C.H. Ling (1965) Representation of Associative Functions. *Publ. Math. Debrecen*, **12**, 182-212.

[23] R. Lowen (1976), Fuzzy topological spaces and fuzzy compactness. *J. Math. Anal. Appl.* **56**, 621-633.

[24] Z. Pawlak (1991) *Rough Sets.* Kluwer. Dordrecht.

[25] Tarski (1955) A lattice-theoretical fixpoint theorem and its applications. *Pacific J. Math.* **5**. 285-309.

[26] E. Trillas (1979). Sobre funciones de negación en la teoría de conjuntos difusos. *Stochastica* **3**, 47-60.

[27] L. Valverde (1985) On the structure of F-indistinguishability operators. *Fuzzy Sets & Systems* **17**, 313-328.

[28] L.A. Zadeh (1971) Similarity Relations and Fuzzy Orderings. *Information Sciences* **3**, 177-200.

# ANTÓNIMOS Y MODIFICADORES LINGÜÍSTICOS

**ADOLFO R. DE SOTO**
Dpto. Direc. y Economía de la Emp.
Universidad de León.
Campus de Vegazana s/n
24071 León.
e-mail: ddears@unileon.es

**ENRIC TRILLAS**
Dpto. de Inteligencia Artificial.
Universidad Politécnica de Madrid

**MARIA JESÚS HERRERO**
Dpto. de Inteligencia Artificial.
Universidad Politécnica de Madrid

## Resumen

La modelización de modificadores lingüísti cos o cercas semánticas en Lógica Fuzzy ha sido objeto de investigación desde casi el comienzo de la teoría. La cualidad de los predicados borrosos de ser graduales es la razón fundamental que posibilita este estudio. En el presente trabajo se distingue entre dos posibles representaciones de estas cercas semánticas. La primera viene dada por operaciones externas al universo de discurso, básicamente en el intervalo $[0, 1]$, a las que hemos denominado "modificadores externos". En cambio la segunda la determinan operaciones internas al universo de discurso, a las que hemos denominado "modificadores internos". El comportamiento de ambos sobre los predicados borrosos y sobre las conectivas lógicas es muy diferente. Zadeh comenzó utilizando modificadores externos para representar cercas como "muy" o expresiones adverviales como "más o menos", pero el uso del lenguaje natural parece exigir una representación mediante modificadores internos.

Esta división entre operadores externos e internos incluye también a la negación en Lógica Fuzzy, que sería un modificador externo, y al fenómeno de la antonimia, que se representaría mediante modificadores internos. La relación de las cercas lingüísticas, graditivos en terminología gramatical moderna, con un predicado y su antónimo se analiza en el trabajo, así como la posibilidad de definir nuevos modificadores a partir de otros y del antónimo.

**Palabras Clave** Modificadores Lingüísticos, Antónimos, Variables Lingüísticas.

## 1 Introducción

La representación de términos lingüísticos que posibilita la teoría de subconjuntos borrosos conlleva la necesidad, y la oportunidad, de representar algunas categorías gramaticales del lenguaje natural. Así en sus primeros trabajos, L. A. Zadeh [15] propone una representación, mediante operaciones sobre las funciones de pertenencia asociadas a los predicados borrosos, de los modificadores lingüísticos o cercas semánticas, adverbios de cantidad como "muy", "poco", "algo", "bastante", o expresiones adverbiales como "más o menos". La posibilidad de aplicar cercas semánticas de este tipo a un término lingüístico recae precisamente en su naturaleza graduable, se puede decir "este problema es muy difícil" y en cambio no tiene sentido "este problema es muy imposible".

La representación propuesta por Zadeh en [15] se basaba en dos operadores fundamentalmente, la concentración y la dilatación, modelizados respectivamente por las funciones $M_{con}(x) = x^2$ y $M_{dil}(x) = \sqrt{x}$, definidas en el intervalo $[0, 1]$. En concreto, si $P$ es un predicado borroso nombrando una propiedad vaga o difusa y $\mu_P$ es un subconjunto borroso representante de la función de pertenencia master del predicado [12], el gradativo "muy"[1] opera sobre $\mu_P$ mediante el operador *con*: $\mu_{muyP}(x) = M_{con}(\mu_P(x)) = \mu_P^2(x)$; análogamente, el gradativo "poco" se modeliza mediante una dilatación: $\mu_{poco\,P}(x) = M_{dil}(\mu_P(x)) = \sqrt{\mu_P(x)}$.

En diversos trabajos como [4, 5, 7] se analiza el comportamiento de las cercas semánticas, en especial de "muy", observándose que no se ajustan, en muchos de sus comportamientos, al operador utilizado por Zadeh. La obtención de la función de pertenencia de un predicado borroso modificado mediante "muy", a partir de la función de pertenencia del predicado original, requiere un desplazamiento de la función del predicado original dentro del universo. Este análisis exigía

[1] *muy* es un apócope de *mucho*.

nuevas formas de representar las cercas semánticas.

Al final del artículo [9] se propone, sin desarrollarlo, un modelo general para representar el efecto de un modificador lingüístico sobre el valor de una función de pertenencia mediante la expresión

$$\mu_{mP}(x) = f_m(\mu_P(g_m(x))).$$

El mismo modelo se estudia en [8] asociando a un modificador $m$ dos funciones: $\nu_m : [0,1] \to [0,1]$ y una traslación $\zeta_m : U \to U$ , donde $U$ es el universo base para el predicado $P$. Esta definición cubriría las dos interpretaciones de "muy", la dada por Zadeh y la presentada en el resto de trabajos comentados. De acuerdo con el autor, en ocasiones "muy" vendría definido mediante una precisión aumentada, cumpliendo entonces $\nu_{muy} \circ \mu_P \circ \zeta_{muy} \leq \mu_P$ con $\zeta_{muy}(x) = x$, o a veces como una traslación, es decir un desplazamiento, sobre el universo $U$, y entonces $\nu_{muy}(x) = x$ y $\zeta_{muy}(x) = x + t$. El modelo además da cuenta de otras operaciones en lógica borrosa. Por ejemplo, la negación no sería más que un modificador dado por una función de negación $\nu_n$, siendo en este caso $\zeta_n$ la identidad, o el antónimo de un predicado $P$ que vendría dado mediante $\nu_{ant} = Id$ y $\zeta_{ant}$ una biyección involutiva sobre el Universo $U$.

La representación de antónimos y sinónimos a través de automorfismos ha sido estudiada en diversos trabajos como en [3, 13] donde se han caracterizado los automorfismos sobre la clase de subconjuntos borrosos $\mathfrak{F}(U)$ dotada con las operaciones definidas por una t-norma y su t-conorma dual. Los resultados obtenidos para las parejas $\min - \max$, el producto y su dual $\Pi - \Pi^*$, y la t-norma de Lukasiewicz y su dual $W - W^*$, indican que la condición necesaria y suficiente para que una biyección $\theta$ sobre $\mathfrak{F}(U)$ sea un automorfismo respecto a alguna de esas parejas es que se descomponga de la forma

$$\theta(\mu(x)) = \theta_x(\mu(s(x))), \qquad (1)$$

donde $\theta_x$ es una familia de automorfismos sobre $[0,1]$, respecto a la pareja t-norma–t-conorma dada, y $s$ es una biyección del universo $U$. De hecho, toda biyección sobre $\mathfrak{F}(U)$ que conserve el orden puntual se descompone de la forma dada en (1).

Estos resultados sugirieron realizar un análisis de los modificadores lingüísticos y de la representación de antónimos en Lógica Fuzzy.

## 2 Modificadores externos e internos

A partir de ahora denominaremos *modificador externo* a cualquier función $M_f : \mathfrak{F}(U) \to \mathfrak{F}(U)$, de la clase de subconjuntos borrosos del universo $U$ en si mismo, definida como

$$M_f(\mu)(x) = f(\mu(x)),$$

donde $f$ es una función de $[0,1]$ en $[0,1]$. Un *modificador interno* será cualquier función $m_s : \mathfrak{F}(U) \to \mathfrak{F}(U)$ definida por

$$m_s(\mu)(x) = \mu(s(x)),$$

siendo $s$ una aplicación del universo $U$ en sí mismo. Habitualmente, se identificará el modificador con la función que lo define, bien sea en $[0,1]$ si es externo o en $U$ si es interno.

En este contexto, según Zadeh, las cercas semánticas "muy" y "poco" se representarían mediante los operadores de concentración $M_{con}(x) = x^2$ y dilatación $M_{dil}(x) = \sqrt{x}$ y serían, por tanto, modificadores externos. En cambio si se considera que estos modificadores desplazan la función de pertenencia del predicado al cual modifican —así parecen atestiguarlo diversos experimentos, como los que aparecen en [7, 9]—, deberían ser modificadores internos, dados, por ejemplo, por una traslación sobre el universo. Si un modificador externo $M_f$ verifica $f(0) = 0$ y $f(1) = 1$, dejará invariante el núcleo y el soporte de cualquier subconjunto borroso al que se aplique, como sucede con los operadores *con* y *dil* anteriormente citados, y que vienen definidos por biyecciones crecientes sobre $[0,1]$. Por el contrario, los modificadores internos podrán dejar invariante, dependiendo de su función de definición, el núcleo o el soporte de algún subconjunto borroso pero nunca de todos. Es de resaltar también que un modificador que desplace parte del núcleo de un subconjunto borroso y deje otra parte invariante no puede ser representado mediante un modificador externo.

Otros ejemplos de modificadores lingüísticos son los modificadores adaptativos estudiados en [2] y denominados "realmente" y "relativamente", y que resultan ser modificadores internos definidos por traslaciones. Un problema surge en la obra citada con estos modificadores, la traslación que define a cada uno de ellos no es una única debido a que ambos se comportan de diferente forma sobre los términos lingüísticos que componen una variable lingüística, dependiendo si se encuentran a la izquierda o a la derecha del centro de la misma. Este fenómeno es semejante al que se produce cuando se analiza como se comporta "muy" sobre un predicado y su antónimo: "muy bajo" indica una altura menor que simplemente bajo, en cambio "muy alto" indica una altura mayor que simplemente "alto", ¿quiere decir ésto que no es posible usar una única traslación para modelizar el comportamiento de "muy" sobre los dos predicados antónimos "alto" y "bajo"? Más adelante veremos como esto no es necesario.

En la misma obra citada se presenta el modificador lingüístico "alrededor de" ("about" en inglés) como un modificador que se comporta ampliando tanto el núcleo como el soporte de una función de pertenencia del predicado al que modifica, obteniendo una nueva función de pertenencia que contiene puntualmente a la original. En este trabajo mostramos como existe una relación entre este modificador y los dos anteriormente citados.

La relación entre las cercas semánticas descritas y las conectivas es una asunto relevante. En los siguientes apartados se muestra el diferente comportamiento de los modificadores, dependiendo si se definen externamente o internamente, con las conectivas y la negación.

## 3 Modificadores y Conectivas

Como ya ha sido comentado en los trabajos [3, 13] se caracterizan las biyecciones sobre $\mathfrak{F}(U)$ que conmutan con ciertas parejas t-norma–t-conorma. Gracias a los resultados allí obtenidos se tiene el siguiente

**Teorema 1** *Un modificador externo $M_f$ dado por una biyección $f$ conmutará con una pareja t-norma–t-conorma dual en las siguientes condiciones:*

1. *Para $\min - \max$, $f$ deberá ser estrictamente creciente*
2. *Para $\Pi - \Pi^*$ y $W - W^*$, no existen modificadores externos, dados por biyecciones, que conmuten con estas parejas.*

En cambio para los modificadores internos la situación es muy distinta.

**Teorema 2** *Un modificador interno $m_s$ dado por una aplicación $s$ sobre el universo $U$ siempre conmuta con cualquier conectiva definida funcionalmente.*

**Demostración.** Sean $\mu, \eta \in \mathfrak{F}(U)$ y $\wedge$ una conectiva definida como

$$(\mu \wedge \eta)(x) = F(\mu(x), \eta(x)),$$

para cierta función $F : [0,1] \times [0,1] \to [0,1]$, entonces para cualquier $m_s$ se tendrá que

$$\begin{aligned}(m_s(\mu \wedge \eta))(x) &= (\mu \wedge \eta)(s(x)) \\ &= F(\mu(s(x)), \eta(s(x))) \\ &= F(m_s(\mu)(x), m_s(\eta)(x)) \\ &= (m_s(\mu) \wedge m_s(\eta))(x).\end{aligned}$$

■

## 4 Modificadores y la negación

La negación en Lógica Fuzzy puede interpretarse como un modificador externo que vendrá dado por una función de negación, es decir una función $n : [0,1] \to [0,1]$ biyectiva, decreciente, involutiva y con $n(0) = 1$ y $n(1) = 0$. Para que un modificador externo $M_f$ conmute con una función de negación se deberá verificar que

$$f(n(x)) = n(f(x)), \tag{2}$$

para todo $x \in [0,1]$, lo que tomando $n$ como la negación clásica $n(x) = 1 - x$ da lugar a la ecuación

$$f(x) + f(1 - x) = 1.$$

Por ejemplo, los modificadores $M_{con}$ y $M_{dil}$ de Zadeh no verifican la ecuación y por tanto no conmutarán con la negación clásica, De hecho, ninguna negación fuerte conmutará con $M_{con}$ puesto que se debería verificar por (2) que

$$n(x)^2 = n(x^2),$$

pero al ser $x^2 < x$ para $0 < x < 1$, se tendrá que $n(x^2) > n(x)$; por otra parte, $n(x)^2 < n(x)$ para $0 < n(x) < 1$, luego se tiene a la vez

$$n(x) > n(x)^2 = n(x^2) > n(x),$$

lo cual es imposible. De forma análoga se puede proceder con $M_{dil}$ sin más que tener en cuenta que $x < \sqrt{x}$ para todo $x \in (0,1)$.

De nuevo, un modificador interno conmuta siempre con cualquier negación, fuerte o no, definida funcionalmente

$$\begin{aligned}m_s(M_n(\mu))(x) &= M_n(\mu)(s(x)) = n(\mu(s(x))) \\ &= n(m_s(\mu)(x)) \\ &= M_n(m_s(\mu))(x).\end{aligned}$$

Esto no tiene una adecuada interpretación si consideramos "muy" representado por un modificador interno: ¿significa lo mismo *no muy alto* que *muy no alto*? Como se puede ver p.e. en [6] la expresión *muy no alto* no suena muy gramatical, p.e. no lo es *muy no muchos* según Lyons. Parece, pues, que se debe imponer una restricción que impida aplicar los modificadores a predicados negados, aunque esto requiere sin lugar a dudas un mayor análisis.

## 5 Modificadores y el antónimo

La representación habitual en Lógica Fuzzy del antónimo de un predicado se realiza mediante un modificador interno involutivo. Por ejemplo, considerando un intervalo acotado de la recta real $[U^-, U^+]$ como

el universo, el antónimo se calcula mediante el modificador interno $m_{ant}$, donde $ant(x) = U^+ + U^- - x$, que resulta ser, evidentemente una biyección involutiva.

Teniendo presente el modelo anterior, y supuesto que en el universo $U$ existe un orden entre sus elementos, parece adecuado definir el antónimo de un predicado de la forma siguiente.

**Definición 3** *Un predicado $P'$ se dice antónimo de $P$ si y sólo si existe una biyección interna, $ant : U \to U$ del universo $U$, decreciente e involutiva tal que:*

$$\mu_{P'}(x) = \mu_P(ant(x)).$$

Con esta definición se cumplen algunas de las propiedades básicas de la antonimia, por ejemplo, el antónimo del antónimo es el predicado original, o la antonimia es recíproca.

Si como universo tomamos $[U^-, U^+]$, la función *ant* será continua y transformará $U^-$ en $U^+$ y $U^+$ en $U^-$, lo que permite demostrar que la función antónimo conserva la convexidad.

**Teorema 4** *Sea $\mu_P$ una función de pertenencia convexa, entonces $\mu_{ant\,P}$ también es convexa.*

Además el antónimo invierte el sentido de crecimiento de los predicados.

**Teorema 5** *$\mu_P$ es monótona decreciente si y sólo si $\mu_{ant\,P}$ es monótona creciente.*

**Demostración.** Evidente, por ser *ant* una función decreciente. ■

Es interesante resaltar que un modificador interno definido mediante una biyección, como el caso del antónimo, puede considerarse como una aplicación del principio de extensión ya que

$$\begin{aligned} m_{ant}(\mu)(y) &= \sup\{\mu(x) : y = ant^{-1}(x)\} \\ &= \sup\{\mu(ant(y)) : ant(y) = x\} \\ &= \mu(ant(y)). \end{aligned}$$

El antónimo, al ser un modificador interno, conmuta con las conectivas lo cual conlleva que el antónimo de *alto y joven* sería *bajo y viejo*. En [6] se señalan dos tipos de antonimia la antipolar y la ortogonal. El autor pone como ejemplos de ambas la relación que existe entre las direcciones cardinales: Norte y Sur serían antónimos antipolares, mientras que Norte y Este, o Norte y Oeste, serían antónimos ortogonales. El mismo caso puede darse entre *alto y joven* respecto a *bajo y viejo, bajo y joven,* y *alto y viejo*: el primero de los tres sería un antónimo antipolar de *alto y joven* y los dos siguientes serían antónimos ortogonales. Nuestro modelo adoptaría como antónimo de la conjunción el antónimo antipolar.

## 6 Modificadores concentradores y dilatadores

Siguiendo la terminología de Zadeh, un modificador $m$ se dice que es un concentrador si se verifica que

$$\mu_{m\,P}(x) \leq \mu_P(x) \tag{3}$$

para cualquier $x \in U$. Un modificador concentrador intensifica el grado de un predicado: *muy bajo* indica una altura menor que simplemente *bajo*.

Por contra, un modificador $m$ se dice que es un dilatador si se verifica la desigualdad contraria a (3). Un dilatador debilita el grado de un predicado: *poco bajo*[2], o también *algo bajo*, son expresiones menos restrictivas respecto de la altura de la persona a la que se le aplican que simplemente *bajo*.

Con estas definiciones, si $m$ es un concentrador definido internamente y $\mu_P$ es una función de pertenencia decreciente se tendrá que:

$$\mu_{m\,P}(x) = \mu_P(m(x)) \leq \mu_P(x)$$

lo que conlleva que $m(x) \geq x$. En cambio si $\mu_P$ es creciente tendrá que ser $m(x) \leq x$.

En una variable lingüística es muy frecuente tener simultáneamente términos lingüísticos cuyas funciones de pertenencia son o bien no decrecientes o bien no crecientes, de lo que se deduce que no podría definirse un único modificador lingüístico interno que fuese un concentrador para todos los términos de la variable lingüística distinto de la identidad. Una solución posible sería tener diferentes modificadores internos dependiendo del predicado, o término lingüístico, al que se aplican. De esta forma se procede en [2], definiendo distintos modificadores para los términos lingüísticos de una variable lingüística dependiendo del lugar en que se encuentren.

La relación entre modificadores concentradores y dilatadores respecto del antónimo de un predicado $P$ es interesante.

**Teorema 6** *Un concentrador dado por un modificador interno $m$ aplicado a una función decreciente $\mu_P$ se comporta como un dilatador con su antónimo.*

**Demostración.** Al ser $\mu_P$ decreciente y $m$ un concentrador tendremos que $m(x) \geq x$, entonces

$$\mu_{m\,(ant\,P)}(x) = \mu_{ant\,P}(m(x)) \geq \mu_{ant\,P}(x),$$

puesto que $\mu_{ant\,P}$ es una función creciente al ser $\mu_P$ decreciente. ■

[2]Usamos *poco* como el recíproco de *muy*. También puede considerarse expresiones como *más o menos*.

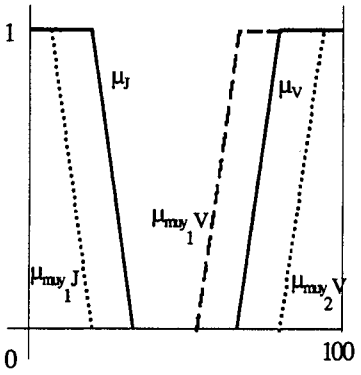



Figura 1: Los predicados *Joven* y *Viejo* y sus modificados

Para que un concentrador opere como tal en $P$ y en $ant\,P$ se debe definir como

$$\mu_{m\,(ant\,P)}(x) \triangleq \mu_{ant\,(m\,P)}(x).$$

Esta definición sería coherente con el comportamiento de los modificadores lingüísticos y el antónimo en el lenguaje natural; parece evidente que el antónimo de *muy alto* deba ser *muy bajo*.

Con la definición anterior, los concentradores y dilatadores se comportan como tales tanto con un predicado como con su antónimo, si $m$ es un concentrador para $P$, entonces también lo es para $ant\,P$ puesto que

$$\begin{aligned}\mu_{m\,(ant\,P)}(x) &\triangleq \mu_{ant\,(m\,P)}(x) = \mu_{m\,P}(ant(x)) \\ &\leq \mu_P(ant(x)) = \mu_{ant\ P}(x),\end{aligned}$$

y si es un dilatador también lo es para $ant\,P$,

$$\begin{aligned}\mu_{m\,(ant\,P)}(x) &\triangleq \mu_{ant\,(m\,P)}(x) = \mu_{m\,P}(ant(x)) \\ &\geq \mu_P(ant(x)) = \mu_{ant\ P}(x).\end{aligned}$$

**Ejemplo 7** *La variable lingüística EDAD*

Tomemos la variable lingüística EDAD con los términos lingüísticos (*Joven*, *Viejo*). Una representación posible de estos predicados, mediante funciones de pertenencia trapezoidales en el universo $U = [0, 100]$, puede ver en la figura 1 en trazo continuo.

En esta variable *Joven* y *Viejo* son antónimos y por tanto sus respectivas funciones de pertenencia se pueden obtener la una de la otra a partir del modificador interno definido por la función $ant(x) = 100 - x$.

El gradativo "muy" actua sobre *Joven* intensificando su significado, exigiendo que una persona tenga menos edad para alcanzar el mismo valor de verdad con el predicado *muy Joven* que con el predicado *Joven*, de hecho lo que parece exigir su comportamiento es que *muy Joven* implique *Joven*, lo que, interpretando la implicación como una inclusión puntual de subconjuntos borrosos, conllevaría que *muy Joven* $\subseteq$ *Joven*. De esta forma una representación de "muy" para el predicado Joven podría venir dada (ver 1) por un modificador interno definido por una traslación acotada del tipo

$$muy_1(x) = \min(100, x + 10).$$

Cuando se aplica este "muy$_1$" al predicado *Viejo*, el resultado obtenido no corresponde a la interpretación de *muy Viejo*. Para conseguir el predicado *muy Viejo* correctamente debemos utilizar como modificador interno representante de "muy" la función

$$muy_2(x) = \max(0, x - 10).$$

Tomando como representación de "muy" $muy_1$, se puede calcular correctamente el predicado *muy Viejo* — que sería $muy_2$ *Viejo*— utilizando el antónimo.

$$\begin{aligned}\mu_{muy\,V}(x) &\triangleq \mu_{ant\,(muy_1\,J)}(x) = \mu_{muy_1\,J}(100 - x) \\ &= \mu_J(\min(100, 100 - x + 10) \\ &= \mu_J(100 - \max(0, x - 10)) \\ &= \mu_{ant\,J}(\max(0, x - 10)) \\ &= \mu_{muy_2\,V}(x).\end{aligned}$$

Por tanto es posible definir el modificador lingüístico "muy" sobre el predicado *Viejo* a través del antónimo de *muy Joven*.

## 7 Modificadores definidos a partir de otros y del antónimo

La relación entre el antónimo y los modificadores es incluso más profunda puesto que es posible definir modificadores a partir de otros modificadores y el antónimo. Considerese la siguiente definición.

**Definición 8** *Un modificador interno se dice a la izquierda (respectivamente a la derecha) si* $m(x) \geq x$ *(resp. si* $m(x) \leq x$*).*

Sea $P$ un predicado no creciente, un modificador interno a la izquierda $m_I$ es un concentrador para $P$, como se ha visto, puesto que:

$$\mu_{m_I\,P}(x) = \mu_P(m_I(x)) \leq \mu_P(x).$$

Recíprocamente, un modificador interno a la derecha $m_D$ será un dilatador para $P$.

Si se define un modificador como

$$m^* = ant \circ m_I \circ ant,$$

es evidente que $m^*$ será un modificador interno a la derecha ya que como $m_I(ant(x)) \geq ant(x)$, entonces

$$m^*(x) = ant(m_I(ant(x))) \leq ant(ant(x)) = x,$$

teniendo en cuenta que *ant* es decreciente e involutiva.

De esta forma un modificador interno a la derecha para $P$, es decir un dilatador, será un modificador "conjugado" de un modificador interno a la izquierda para $P$, es decir un concentrador. El comportamiento de $m^*$ con el antónimo de $P$ es el adecuado, el de un dilatador,

$$\begin{aligned} \mu_{m^*(ant\,P)}(x) &= \mu_{ant(m^*\,P)}(x) = \mu_{m^*\,P}(ant(x)) \\ &= \mu_P(m^*(ant(x))) \\ &\geq \mu_P(ant(x)) = \mu_{ant\,P}(x), \end{aligned}$$

ya que $m^*(ant(x)) \leq ant(x)$, y $\mu_P$ es decreciente.

**Ejemplo 9** *(Continuación)*

En el ejemplo mostrado anteriormente, el predicado $muy_1$ *Viejo* correspondería al modificador lingüístico *poco Viejo*, es decir a *dilatar* el predicado *Viejo*. Podemos definir este modificador como

$$\mu_{poco\,V}(x) \triangleq \mu_{muy_1\,V}(x).$$

Para definir la aplicación de "poco" al predicado *Joven* utilizamos de nuevo el antónimo:

$$\begin{aligned} \mu_{poco\,J}(x) &\triangleq \mu_{ant\,(poco\,V)}(x) = \mu_{ant\,(muy_1\,V)}(x) \\ &= \mu_{muy_1\,V}(100 - x) \\ &= \mu_V(\min(100, 100 - x + 10) \\ &= \mu_V(100 - \max(0, x - 10)) \\ &= \mu_{ant\,V}(\max(0, x - 10)) \\ &= \mu_J(\max(0, x - 10)) \\ &= \mu_{muy_2\,J}(x). \end{aligned}$$

que se adecua a la interpretación habitual de *poco Joven*.

La relación entre $muy_1$ y $muy_2$ es, efectivamente,

$$muy_1 = ant \circ muy_2 \circ ant.^3$$

Estos resultados posibilitan definir todos los términos lingüísticos de la variable EDAD que hemos venido utilizando a partir únicamente de tres elementos: (*Joven*, $muy_1$, *ant*) o bien con (*Viejo*, $muy_2$, *ant*), puesto que el resto se obtendrá como una combinación de estos y de las conectivas estándar en Lógica Fuzzy, lo que se puede interpretar como la existencia de una relación de dualidad entre el comportamiento de los modificadores lingüísticos con un predicado y su antónimo.

[3] Algo similar a esta relación ya fue utilizada por Zadeh en [15] con la negación externa habitual $\neg$ al definir *more or less tall* como $\neg SF(\neg tall, k)$.

## 8 El término central de una variable lingüística

Dado un predicado y su antónimo ($P$, *ant* $P$) tiene sentido considerar el predicado $Q = ni\ P\ ni\ ant\ P$, el cual se puede calcular a partir de las conectivas intersección y negación estándar de la siguiente forma:

$$\mu_Q(x) = \min(1 - \mu_P(x), 1 - \mu_{ant\,P}(x)). \quad (4)$$

Supongamos que $Q$ es normal, es decir, que existe un $x$ de universo tal que $\mu_Q(x) = 1$, lo que ocurrirá si y sólo si $\min(1 - \mu_P(x), 1 - \mu_{ant\,P}(x)) = 1$, es decir si y sólo si existe $x$ en $U$ tal que $\mu_P(x) = 0$ y $\mu_{ant\,P}(x) = 0$.

Sea $m$ un modificador interno para $P$, que suponemos tiene asociada una función de pertenencia $\mu_P$ no creciente, veamos como se comportan este modificador con el término central $Q$,

$$\begin{aligned} \mu_{mQ}(x) &= \mu_Q(m(x)) \quad (5) \\ &= \min(1 - \mu_P(m(x)), 1 - \mu_{ant\,P}(m(x))) \\ &= \min(1 - \mu_{m\,P}(x), 1 - \mu_{m\,(ant\,P)}(x)) \\ &= \min(1 - \mu_{m\,P}(x), 1 - \mu_{ant\,(m\,P)}(x)), \end{aligned}$$

si $m$ es un concentrador se tendrá tanto $\mu_{m\,P}(x) \leq \mu_P(x)$ como $\mu_{ant\,(m\,P)}(x) \leq \mu_{ant\,P}(x)$, luego

$$\begin{aligned} \mu_{mQ}(x) &= \min(1 - \mu_{m\,P}(x), 1 - \mu_{ant\,(m\,P)}(x)) \\ &\geq \min(1 - \mu_P(x), 1 - \mu_{ant\,P}(x)) \\ &= \mu_Q(x). \end{aligned}$$

y $m$ se comporta como un dilatador con $Q$. Es fácil comprobar que si $m$ fuese un dilatador para $P$ se comportaría como un concentrador con $Q$.

La forma anterior de definir un modificador sobre $Q$ viene dada al aplicar un modificador al antónimo de un predicado mediante el antónimo del predicado modificado, es decir definiendo la aplicación del modificador mediante el antónimo. Es interesante observar que si $m$ fuese un modificador interno a la izquierda y utilizamos su definición directa, sin el antónimo, en su aplicación a $Q$ mediante

$$\mu_{m\,ant\,P}(x) \triangleq \mu_{ant\,P}(m(x)), \quad (6)$$

entonces $\mu_{m\,Q}(x)$ sería un predicado resultado de un "desplazamiento" de $Q$, de tal forma que no sería ni un concentrador ni un dilatador como se puede ver en el siguiente ejemplo.

**Ejemplo 10** *(Continuación)*

El comportamiento de los modificadores $muy_1$ y $muy_2$ sobre el predicado central *Mediana*, significando "de mediana edad" en la variable lingüística EDAD y con función de pertenencia

$$\mu_M(x) = Min(1 - \mu_J(x), 1 - \mu_V(x)),$$

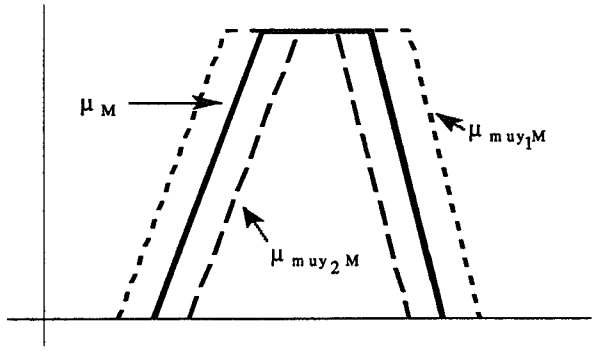



Figura 2: El predicado *Mediana* y sus modificados, utilizando el antónimo

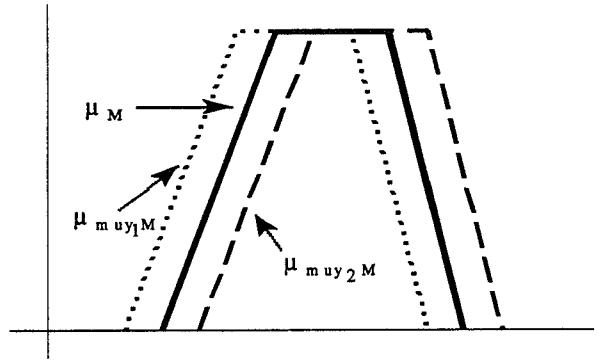



Figura 3: El predicado *Mediana* y sus modificados sin el antónimo

bajo el criterio dado en las ecuaciones (5) es el que aparece en la figura 2. Como se puede ver, uno de ellos es un concentrador, sería $\mu_{muy_2\,M}$ y viene dado por el trazo más discontinuo en la figura, y correspondería al comportamiento de *muy de Mediana edad,* mientras que el otro es un dilatador, $\mu_{muy_1\,M}$ dado por el trazo menos discontinuo, y reflejaría el comportamiento de *más o menos de Mediana edad.* Este último modificador es el que se denomina "alrededor de" (about) en [2].

Si definimos el comportamiento de $muy_1$ y de $muy_2$ sobre el predicado *Mediana* mediante la ecuación (6), los predicados modificados se dan en la figura 3. Conllevan un desplazamiento a la izquierda, sería $\mu_{muy_1\,M}$ dado por el trazo más discontinuo, o la derecha, en este caso $\mu_{muy_2\,M}$ dado por el trazo menos discontinuo en la figura, de la función de pertenencia $\mu_M$. Estos modificadores serían los denominados "realmente" (really) y "relativamente" (relatively) en [2].

## Referencias


[1] BELLO, A. (1988) *Gramática de la Lengua Castellana* Arco/Libros, Madrid.

[2] BOUCHON-MEUNIER, B.; JIA, Y. (1992) "Linguistic Modifiers and Imprecise Categories." *Inter. Journal of Intelligent Systems,* Vol. 7, 25–36.

[3] DE SOTO, A. R. (1996) "On automorphisms, Synonyms and Antonyms in Fuzzy Set Theory." *Proc. ITHURS'96,* Vol. I, 217–221.

[4] HERSH, H. M.; CARAMAZZA, A. (1976) "A Fuzzy Set Approach to Modifiers and Vagueness in Natural Language." *Journal of Experimental Psychology: General,* Vol. 105, No. 3, 254–276.

[5] LAKOFF, G. (1973) "Hedges: A study of meaning criteria and the logic of fuzzy concepts." *Journal of Philosophical Logic,* Vol. 2, 458–508.

[6] LYONS, J. (1980) *Semántica.* Ed. Teide, Barcelona.

[7] MACVICAR-WHELAN, P.J. (1978) "Fuzzy Sets, the Concept of Height, and the Hedge "very"." *IEEE Systems, Man and Cybernetics,* Vol. 8, 507–511.

[8] NOVAK, V. (1989) *Fuzzy Sets and their Applications.* Adam Hilger, Bristol-Philadelphia.

[9] NORWICH, A.M.; TURKSEN, I.B. (1984) "A model for the measurement of membership and the consequences of its empirical implementation." *Fuzzy Sets and Systems,* Vol. 12, 1–25.

[10] OVCHINNIKOV, S.V. (1981) "Representations of Synonymy and Antonymy by Automorphisms in Fuzzy Set Theory." *Stochastica,* Vol. 2, 95–107.

[11] PIERA, N. (1983) "Morphisms and Synonymy" *Proc. F.I.S.A.L.,* 101-109.

[12] TRILLAS, E.; CUBILLO, S.; DE SOTO, A. R. (1994) An Essay on Name and Extension of Rule-Given Properties. *Proc. IPMU'94,* 571–576.

[13] TRILLAS, E.; DE SOTO, A.R.; CUBILLO, S. (1994) "Ovchinnikov's automorphisms Revisited." *Mathware & Soft Computing,* Vol. I, No. 1, 83–92.

[14] TRILLAS, E.; RIERA, T. (1980) "Towards a representation of synonyms and antonyms by Fuzzy Sets" *Busefal,* 5, 42-68.

[15] ZADEH, L. A. (1972) A Fuzzy Set Theoretic Interpretation of Linguistic Hedges. *Journal of Cybernetics,* 2, 4–34.

# UN ENFOQUE LÓGICO PARA CALCULAR EL GRADO DE CUMPLIMIENTO DE SENTENCIAS CON CUANTIFICADORES LINGUÍSTICOS

Miguel Delgado, Daniel Sánchez y María Amparo Vila
Departamento de Ciencias de la Computacion e I. A. Universidad de Granada
E.T.S.I.Informatica, Avd. de Andalucia n. 38 18071, Granada, España
email: vila@decsai.ugr.es, daniel@decsai.ugr.es

## 1 Introduccion

Las sentencias con cuantificadores que describen propiedades difusas se utilizan en diversos problemas. Varios métodos han sido propuestos para la evaluación del grado de cumplimiento de estas sentencias. Podemos destacar las propuestas de Zadeh [4], Yager [2] y Vila et. al. [1]. En [1] además se realiza una comparación entre estos enfoques, señalando ventajas e inconvenientes de los mismos en determinados casos.

En este trabajo queremos exponer un enfoque distinto para la evaluación de los grados de cumplimiento. Se basa en el uso de la lógica, dando una interpretación semántica a cada uno de los pasos que se realizan. Una de las principales ventajas es que es posible aplicarlo a ambos tipos de sentencias, y dentro de ellas a cualquier tipo de cuantificador, sea absoluto o relativo. En realidad no se propone una función de evaluación concreta, sino una familia de funciones que utilizan t-normas, t-conormas, negaciones e implicaciones. Cada combinación de estas funciones lógicas utilizada dará lugar a una forma distinta de evaluar las sentencias. Por otra parte, las fórmulas lógicas en las que se utilizan estos operadores tienen un significado semántico claro, por lo que podemos decidir el operador a utilizar en cada caso teniendo en cuenta sus propiedades y la forma en que estas afectan a la evaluación de la fórmula. De esta forma se puede evaluar una sentencia haciendo que determinados pasos sean optimistas y otros más restrictivos, según cómo entendemos que deben considerarse.

## 2 Sentencias y cuantificadores

Vamos a considerar dos tipos de sentencias, que pueden esquematizarse como:

Tipo I: Q de los X son A

Tipo II: Q de los D son A

donde X es un conjunto finito, D y A son propiedades difusas sobre los elementos de X, y Q es un cuantificador linguístico. Ejemplos de este tipo de sentencias podrían ser:

*La mayoría de los estudiantes son jóvenes*

*Unos pocos de los estudiantes jóvenes son eficientes*

En estos casos, los cuantificadores linguísticos son "la mayoría" y "unos pocos", el conjunto X es un conjunto finito de estudiantes, y A y D son las propiedades difusas "joven" y "eficiente" respectivamente, que se aplican sobre el conjunto X.

El objetivo es, partiendo de estos datos, evaluar el grado de cumplimiento de la sentencia. Es decir, en qué grado "la mayoría de los estudiantes son jóvenes", y en qué grado "unos pocos de los estudiantes jóvenes son eficientes".

En general consideraremos dos tipos de cuantificadores que tienen una representación matemática distinta:

Cuantificadores absolutos: se definen sobre un intervalo de los números enteros no negativos, y son de la forma "no mucho más de 5", "3 o más", "alrededor de 7", etc... Son versiones suavizadas de cantidades enteras o intervalos enteros.

Cuantificadores relativos: se definen sobre el intervalo [0,1], y representan proporciones del total, como por ejemplo "todos", "la mayoría", etc...

Cada combinación de tipo de sentencia y tipo de cuantificador requiere un tratamiento distinto con los métodos utilizados hasta ahora.

## 3 Un enfoque alternativo

El enfoque que proponemos se basa en la definición de una fórmula lógica cuyo valor final constituye la evaluación de la sentencia.

### 3.1 Evaluando sentencias tipo I

#### 3.1.1 Definición:

Sea X un conjunto finito $\{P_1, ..., P_n\}$ y A un conjunto difuso sobre X que refleja alguna propiedad imprecisa. Defini mos el grado con el cual k o más elementos verifican la propiedad A, $L_k(A)$ de la siguiente manera:

$$L_k(A) = \begin{array}{ll} 1 & si\ k = 0 \\ 0 & si\ k > n \\ \bigoplus_{(i_1,\ldots,i_k) \in I_k} (A(P_{i_1}) \bigodot_{\ldots} \bigodot A(P_{i_k})) & si\ 1 \leq k \leq n \end{array}$$

donde $I_k$ es el conjunto de k-uplas de índices dado por:
$I_k = \{(i_1,...,i_k)/i_1 < i_2 ... < i_k; \quad i_j \in \{1,...,n\}$
$\forall j \in \{1,...,k\}\}$
y $\bigodot$ y $\bigoplus$ son una t-norma y una t-conorma respectivamente.

Es fácil comprobar que la definición cumple las siguientes propiedades:

Propiedad 1: Se verifica

$$\forall l, m \in \{1,...,n\} \qquad l < m \Rightarrow L_l(A) > L_m(A)$$

Propiedad 2: Supongamos que tomamos: $\bigodot$=mínimo, $\bigoplus$=máximo. Sea $a_i = A(P_i)$, y sean $b_i$ los valores de $a_i$ ordenados de mayor a menor. Entonces se verifica

$$L_k(A) = b_k$$

Propiedad 3: En el caso no difuso, basta con que k o más de los valores sean 1 para que obtengamos $L_k(A) = 1$. Este resultado es coherente con la semántica de la función.

### 3.1.2 Definición:

En las condiciones anteriores definimos el grado en que exactamente k de los elementos verifican la propiedad A como:

$$E_k(A) = L_k(A) \bigodot \overline{L_{k+1}(A)}$$

donde la barra representa un operador de negación, y $\bigodot$ es una t-norma cualquiera, no necesariamente la utilizada en las funciones L.

Propiedad 4: Tomando para la función E como negación n(x)=1-x y como t-norma la de Lukasiewicz, L(x,y)=max{x+y-1,0}, obtenemos que

$$E_k(A) = L_k(A) - L_{k+1}(A)$$

Como consecuencia

$$\sum_{i=0}^{n} E_i(A) = 1$$

Propiedad 5: En el caso no difuso, es necesario que exactamente k de los elementos cumplan la propiedad para que $E_k(A) = 1$.

### 3.1.3 Caso de cuantificadores absolutos

Pasamos ahora a utilizar las definiciones de $L_k(A)$ y $E_k(A)$ para evaluar el grado de cumplimiento de "Q de X son A" cuando Q representa un cuantificador absoluto. Dicha evaluación se basa en las siguientes consideraciones:

- Q(i) identifica por definición el grado en que podemos decir que, si i de los elementos cumplen una cierta propiedad, "Q" de los elementos verifican la propiedad
- $E_i(A)$ identifica el grado en que i de los n elementos verifican la propiedad A

Luego para evaluar finalmente el grado de cumplimiento de la sentencia, proponemos calcular el grado de adecuación entre ambos conjuntos difusos. Para ello definimos la función G como:

$$G_Q(A) = \bigoplus_{i=1,...,n} (E_i(A) \bigodot Q(i))$$

donde $\bigodot$ y $\bigoplus$ son una t-norma y una t-conorma respectivamente, no necesariamente las utilizadas en las definiciones de L y E.

### 3.1.4 Caso de cuantificadores relativos

De la misma forma que en el caso anterior se podría decir que:

- Q(i/n) identifica por definición el grado en que podemos decir que, si i/n del total de elementos verifican una cierta propiedad, "Q" de los elementos verifican dicha propiedad
- $E_i(A)$ identifica el grado en que i de los n elementos verifican la propiedad, es decir, i/n del total de elementos

Luego para evaluar finalmente el grado de cumplimiento de la sentencia, proponemos nuevamente calcular el grado de adecuación entre ambos difusos mediante la función G definida como:

$$G_Q(A) = \bigoplus_{i=1,...,n} (E_i(A) \bigodot Q(i/n))$$

donde $\bigodot$ y $\bigoplus$ son una t-norma y una t-conorma cualesquiera, respectivamente.

Esta definición tiene como caso particular los operadores OWA de Yager, como lo prueba la siguiente propiedad:

Propiedad 6: Tomando para la función L como t-conorma el máximo y como t-norma el mínimo, para la función E como negación n(x)=1-x y como t-norma la de Lukasiewicz, y tomando para la función G como t-norma el producto y como t-conorma la de Lukasiewicz, obtenemos que

$$G_Q(A) = Y_Q(A)$$

es decir, el operador OWA de Yager aplicado a la evaluación de sentencias tipo I con cuantificadores relativos.

Como caso particular, y que nos será útil en el siguiente apartado, reseñar que se puede verificar:

$$G_{\exists}(A) = b_1 = L_1(A) = Y_{\exists}(A) = \max_{x \in X} A(x)$$

y, además

$$G_{\forall}(A) = b_n = L_n(A) = Y_{\forall}(A) = \min_{x \in X} A(x)$$

resultados que son coherentes desde el punto de vista de la lógica.

Propiedad 7: En las condiciones de la propiedad anterior, tanto en el caso de cuantificadores absolutos como relativos, no es necesario que los cuantificadores sean crecientes para que el valor obtenido pertenezca al intervalo [0,1].

Propiedad 8: En el caso no difuso, y para cualesquiera t-normas y t-conormas en las definiciones de L, E y G, si k de los n elementos del conjunto X verifican la propiedad A con grado 1, se verifica que

$$G_Q(A) = Q(k)$$

para cuantificadores absolutos, y

$$G_Q(A) = Q(k/n)$$

para cuantificadores relativos.

## 3.2 Evaluando sentencias tipo II

### 3.2.1 Definición:

En las condiciones de las definiciones anteriores, para todo elemento x de X y para todo valor k en {1,...,n}, definimos

$f_Q(x, A/D) = [o_Q * (D(x) \wedge A(x))] + [(1 - o_Q) * ((1 - D(x)) \vee A(x))]$

Donde $o_Q$ representa la medida "orness" de Yager para un cuantificador, que representa el grado en que está cercano a la disyunción (existencia) de los valores de X. Dicha medida se define para cuantificadores relativos como

$$o_Q = \sum_{i=1}^{n} [((n-i)/(n-1)) * (Q(i/n) - Q((i-1)/n))]$$

y proponemos definirla para cuantificadores absolutos como

$$o_Q = \sum_{i=1}^{n} [((n-i)/(n-1)) * (Q(i) - Q(i-1))]$$

Puede comprobarse que la forma de definir $o_Q$ nos permite dar más importancia a la conjunción o implicación según el cuantificador se mueve entre $\exists$ y $\forall$. Un ejemplo de ello es la siguiente propiedad

Propiedad 9: como casos particulares fácilmente demostrables obtenemos

- Si $Q = \exists$ entonces

$$o_Q = 1 \Rightarrow f_{\exists}(x, A/D) = (D(x) \wedge A(x))$$

- Si$Q = \forall$ entonces

$$o_Q = 0 \Rightarrow f_{\forall}(x, A/D) = (1 - D(x)) \vee A(x)$$

Propiedad 10: para el caso en que $D(x) = 1 \; \forall x \in X$, tenemos que

$$f_Q(x, A/D) = A(x) \; \forall Q$$

### 3.2.2 Definición:

Para el caso de sentencias tipo II definimos L como sigue:

$$L_k(A/D, Q) = \begin{cases} 1 & si\ k = 0 \\ 0 & si\ k > n \\ \bigoplus_{(i_1,\ldots,i_k) \in I_k} (f_Q(P_{i_1}, A/D) \odot \ldots \\ \odot f_Q(P_{i_k}, A/D)) & si\ 1 \leq k \leq n \end{cases}$$

donde $I_k$ es el conjunto de k-uplas de índices de la definición 3.1.1 y $\odot$ y $\bigoplus$ son una t-norma y una t-conorma respectivamente.

Propiedad 11: Se mantiene la propiedad 1 para la nueva función L, es decir

$\forall l, m \in \{1,...,n\} l < m \Rightarrow L_l(A/D, Q) > L_m(A/D, Q)$

Propiedad 12: Se mantiene la propiedad 2 para la nueva función L, es decir, si consideramos el conjunto $c_i$ como los valores de $f_Q(P_j, A/D)$ ordenados de mayor a menor, tenemos que

$$L_k(A/D, Q) = c_k$$

Propiedad 13: Como casos particulares

$$L_1(A/D, \exists) = \max_{i \in \{1,\ldots,n\}} f_{\exists}(P_i, A/D) = \max_{i \in \{1,\ldots,n\}} (D(P_i) \wedge A(P_i))$$

y de la misma forma

$$L_n(A/D, \forall) = \min_{i \in \{1,\ldots,n\}} f_{\forall}(P_i, A/D) = \min_{i \in \{1,\ldots,n\}} ((1 - D(P_i)) \vee A(P_i))$$

### 3.2.3 Evaluación de sentencias tipo II

Definimos

$$E_k(A/D, Q) = L_k(A/D, Q) \odot \overline{L_{k+1}(A/D, Q)}$$

con el mismo sentido que en la definición 3.1.2

Propiedad 14: La función E así definida sigue verificando la propiedad 4, es decir, utilizando como negación n(x)=1-x y como t-norma la de Lukasiewicz obtenemos

$$E_k(A/D,Q) = L_k(A/D,Q) - L_{k+1}(A/D,Q)$$

De la misma forma, para la evaluación de sentencias tipo II con cuantificadores absolutos definimos

$$G_Q(A/D) = \bigoplus_{i=1,...,n} (E_i(A/D,Q) \bigodot Q(i))$$

y para cuantificadores relativos

$$G_Q(A/D) = \bigoplus_{i=1,...,n} (E_i(A/D,Q) \bigodot Q(i/n))$$

Propiedad 15: Es fácil verificar que, en las condiciones de la propiedad 6, se siguen verificando los casos particulares

$$G_\exists(A/D) = L_1(A/D,\exists)$$

y además

$$G_\forall(A/D) = L_n(A/D,\forall)$$

y por lo tanto, teniendo en cuenta la propiedad 13, tene mos que

$$G_\exists(A/D) = \max_{i \in \{1,...,n\}} (D(P_i) \wedge A(P_i))$$

y además

$$G_\forall(A/D) = \min_{i \in \{1,...,n\}} ((1 - D(P_i)) \vee A(P_i))$$

resultados que son coherentes con los expresados por la lógica. Un pequeño matiz sobre esto se discute en el apartado 4, referido a consideraciones semánticas.

Propiedad 16: en las condiciones de la propiedad 15, no se verifica que la función G sea igual al método de Yager para evaluar sentencias tipo II, basado también en el uso de operadores OWA.

Propiedad 17: Una propiedad adicional, y semánticamente correcta, es que en el caso de que D(x)=1 para todo x, tenemos que, independientemente del cuantificador Q, por la propiedad 9

$$f_Q(x, A/D) = A(x)$$

de lo que se deduce

$$G_Q(A/D) = G_Q(A)$$

Propiedad 18: Se verifica la propiedad 7 también para el caso de la evaluación de sentencias tipo II.

Propiedad 19: G es simétrico en todos los casos con res pecto al orden de los elementos de X.

Propiedad 20: En las condiciones de la propiedad 6, se verifica lo siguiente: dado el cuantificador absoluto $Q_k$ ="k o más", para un cierto k, se verifica que

$$G_{Q_k}(A/D) = L_k(A/D,Q_k)$$

Propiedad 21: Para cualquier selección de operadores lógicos, sea $Q^k$ el cuantificador absoluto "exactamente k" para un cierto k, se verifica

$$G_{Q^k}(A/D) = E_k(A/D,Q)$$

# 4 Consideraciones semánticas

Como hemos visto, las expresiones lógicas para L, E y G utilizadas por el método tienen una semántica clara. Por otro lado, la definición de cada una de las expresiones deja libre la posible elección de los operadores lógicos por parte del usuario del método.

De cualquier modo, es claro para nosotros que el modo de evaluar las sentencias debe cumplir ciertas propiedades, como lo son las dadas en las propiedades 15, 17, 20 y 21, coherentes con la lógica difusa. En realidad, la propiedad 15 es un caso particular de la propiedad 20.

Nuestro método permite determinar la t-norma y la t-conorma que queremos utilizar para la conjunción y disyunción de los valores. De hecho, estos son la t-norma y la t-conorma que utilizamos para definir la función L. Así, es fácil demostrar que si utilizamos como t-conorma el máximo y como t-norma el producto en las condiciones que hacen equivalente el método al de Yager en sentencias tipo I, obtenemos:

$$G_\exists(A) = L_1(A) = \max_{i \in \{1,...,n\}} A(P_i)$$

que era el valor obtenido antes, mientras que

$$G_\forall(A) = L_n(A) = \prod_{i=1}^{n} A(P_i)$$

De la misma forma, otras propiedades como la simetría con respecto al orden de los elementos de X, deben ser aseguradas por cuanto son propiedades irrenunciables en un operador de agregación del tipo que estamos estudiando. Hemos demostrado que esto se verifica (propiedad 19).

# 5 Comparación con otros métodos

En esta sección pretendemos establecer una comparación entre nuestro método y otros, partiendo del análisis comparativo realizado por Vila *et al.* en [1]. En [1] se muestra en particular como el método propuesto por Zadeh es demasiado estricto en la evaluación se sentencias de ambos

tipos. Por otra parte, el método de Yager para sentencias tipo II no proporciona un resultado consistente con la lógi ca difusa en el caso del cuantificador $\exists$, como hemos des crito en la propiedad 14. Por otro lado, hemos establecido que tanto el método de Zadeh como el de Yager para sentencias tipo II se reducen al caso de sentencias tipo I cuando D(x)=1 para todo X.

Algunos de estos puntos pueden observarse en los experimentos realizados considerando un conjunto X de 6 elementos, y con diferentes valores para los conjuntos A (aleatorio) y D y para el cuantificador Q. Hemos tomado para el método los operadores lógicos que dan como caso particular el método de los operadores OWA de Yager.

Ejemplo 1:

| k | A | D | Q(k/6) | $E_k(A/D,Q) * Q(k/n)$ |
|---|---|---|---|---|
| 0 | | | 0 | 0.00 |
| 1 | 0.48 | 0.3 | 1 | 0.07 |
| 2 | 0.58 | 0.4 | 1 | 0.04 |
| 3 | 0.08 | 0.5 | 1 | 0.00 |
| 4 | 0.47 | 0.5 | 1 | 0.04 |
| 5 | 0.51 | 0.8 | 1 | 0.34 |
| 6 | 0.43 | 1 | 1 | 0.08 |

| Método | Valor obtenido |
|---|---|
| $Z_Q(A)$ | 1 |
| $Y_Q(A)$ | 0.58 |
| $G_Q(A)$ | 0.58 |
| $Z_Q(A/D)$ | 1 |
| $Y_Q(A/D)$ | 0.70 |
| $G_Q(A/D)$ | 0.51 |
| $V_Q(A/D)$ | 0.51 |

Ejemplo 2:

| k | A | D | Q(k/6) | $E_k(A/D,Q) * Q(k/n)$ |
|---|---|---|---|---|
| 0 | | | 0 | 0.00 |
| 1 | 0.66 | 1 | 0 | 0.00 |
| 2 | 0.58 | 1 | 0 | 0.00 |
| 3 | 0.80 | 1 | 0 | 0.00 |
| 4 | 0.54 | 1 | 0 | 0.00 |
| 5 | 0.40 | 1 | 0 | 0.00 |
| 6 | 0.43 | 1 | 1 | 0.40 |

| Método | Valor obtenido |
|---|---|
| $Z_Q(A)$ | 0 |
| $Y_Q(A)$ | 0.40 |
| $G_Q(A)$ | 0.40 |
| $Z_Q(A/D)$ | 0 |
| $Y_Q(A/D)$ | 0.40 |
| $G_Q(A/D)$ | 0.40 |
| $V_Q(A/D)$ | 0.40 |

Ejemplo 3:

| k | A | D | Q(k/6) | $E_k(A/D,Q) * Q(k/n)$ |
|---|---|---|---|---|
| 0 | | | 0 | 0.00 |
| 1 | 0.98 | 0.3 | 0 | 0.00 |
| 2 | 0.34 | 0.4 | 0 | 0.00 |
| 3 | 0.43 | 0.5 | 0 | 0.00 |
| 4 | 0.45 | 0.5 | 0 | 0.00 |
| 5 | 0.82 | 0.8 | 0 | 0.00 |
| 6 | 0.98 | 1 | 1 | 0.34 |

| Método | Valor obtenido |
|---|---|
| $Z_Q(A)$ | 0 |
| $Y_Q(A)$ | 0.34 |
| $G_Q(A)$ | 0.34 |
| $Z_Q(A/D)$ | 0 |
| $Y_Q(A/D)$ | 0.50 |
| $G_Q(A/D)$ | 0.50 |
| $V_Q(A/D)$ | 0.50 |

Ejemplo 4:

| k | A | D | Q(k/6) | $E_k(A/D,Q) * Q(k/n)$ |
|---|---|---|---|---|
| 0 | | | 0 | 0.00 |
| 1 | 0.39 | 0.30 | 0.17 | 0.05 |
| 2 | 0.21 | 0.40 | 0.33 | 0.03 |
| 3 | 0.77 | 0.50 | 0.50 | 0.09 |
| 4 | 0.13 | 0.50 | 0.67 | 0.04 |
| 5 | 0.49 | 0.80 | 0.83 | 0.01 |
| 6 | 0.14 | 1 | 1 | 0.13 |

| Método | Valor obtenido |
|---|---|
| $Z_Q(A)$ | 0.35 |
| $Y_Q(A)$ | 0.35 |
| $G_Q(A)$ | 0.35 |
| $Z_Q(A/D)$ | 0.51 |
| $Y_Q(A/D)$ | 0.46 |
| $G_Q(A/D)$ | 0.41 |
| $V_Q(A/D)$ | 0.32 |

# 6 CONCLUSIONES

Hemos definido una forma de evaluar sentencias que involucran propiedades difusas y cuantificadores linguísticos de una forma semánticamente intuitiva. Algunas de las ventajas que ofrece este método con respecto a otros estudiados es que trata todos los casos de forma homogénea, no distinguiendo entre sentencias de tipo I o II, siempre que se use la última de las expresiones señaladas (para tipo II).

Por otra parte, hemos visto que el método propuesto por Yager es un caso particular de nuestro operador para sentencias tipo I con D(x)=1 para todo x, y suponiendo Q(0)=0

y que Q es creciente. No ocurre así en el caso de sentencias tipo II.

También hemos establecido que es posible utilizar el método con cuantificadores que no verifiquen las condiciones anteriores, es decir, que no sean crecientes y/o que verifiquen $Q(0)>0$ y $Q(1)<1$, cuantificadores que tienen un sentido semántico claro (por ejemplo, "la mayoría pero no todos"). En este sentido, podríamos incluso calcular dicho cuantificador a través de operaciones lógicas entre los cuantificadores "la mayoría" y "todos", lo cual abre múltiples posibilidades de combinación entre cuantificadores, todos los cuales son evaluables por el método.

Otro aspecto confirmado es que, utilizando las t-normas y t-conormas que dan lugar al método de Yager tanto para sentencias tipo I como para tipo II, se obtienen resultados coherentes con la lógica en la evaluación usando como cuantificadores el $\forall$ y el $\exists$

# Referencias


[1] . Vila, M.A., Cubero J.C., Medina J.M., Pons O. "Using OWA operator in flexible query processing". Por aparecer en el libro "The Ordered Weighted Averaging Operators: Theory, Methodology and Applications". Eds. R.R. Yager y J. Kacprzyk

[2] . Yager, R.R. "On ordered weighted averaging aggregation operators in Multicriteria DecisionMaking". IEEE Transactions on Systems, Man and Cybernetics, vol. 18, n. 1 January/February, (1988)

[3] . Yager, R.R. "Families of OWA operators". Fuzzy Sets and Systems 59 (1993) 125-148

[4] . Zadeh, L.A. "A computational approach to fuzzy quantifiers in natural languages". Computing and Mathematics with Applications Vol. 9 (1983) pp. 149-184

# ON THE SEMANTICS OF FUZZY STATEMENTS BASED ON POSSIBILISTIC CONSTRAINTS

**Francesc ESTEVA, Pere GARCIA and Lluís GODO**
Institut d'Investigació en Intel.ligència Artificial (IIIA) - CSIC,
Campus Univ. Autònoma Barcelona
08193 Bellaterra, Barcelona, Spain
e-mails: {esteva,pere,godo}@iiia.csic.es

## Abstract

The meaning of statement "X is A" has been traditionally stated in terms of the so-called Possibility Assignment Principle (PAP). However, some times this semantics is not fully satisfactory. In this paper we understand each fuzzy statement as a constraint on the unknown (or partially known) possibility distribution of the variables involved in the expression. Based on that, we explore the different semantics that make sense for the most usual fuzzy statements, i.e. the elementary fuzzy statements of the type "X is A" and the fuzzy "if-then" rules.

**Keywords:** Possibility, Necessity, Fuzzy rules, Constraint combination, Belief-based rules.

## 1 INTRODUCTION

Following Zadeh, see [9], the canonical form of a fuzzy statements is of the form "X is A", where X denotes a (possibly complex) variable taking values in some domain U, and A denotes a fuzzy subset of U. The meaning of such an statement has been traditionally stated in terms of the so-called Possibility Assignment Principle (PAP), (see [9]), the assertion of the statement "X is A" induces the following possibility distribution on U:

$$\pi(X = u) = \mu_A(u) \qquad (i)$$

that is, the degree in which u is a possible value for X is just the membership degree of u in the fuzzy set A. Throughout the paper we shall also use the notation $\pi_X(u)$ for $\pi(X = u)$. However this semantics is not fully satisfactory. Suppose that besides (i) it is asserted that "X is B", where B is another fuzzy set of U. Then, according to PAP, the possibility distribution induced would be:

$$\pi(X = u) = \mu_B(u) \qquad (ii)$$

Further, if both "X is A" and "X is B" are asserted (or observed), one could also be able to assert "X is A$\cap$B", and thus, the corresponding possibility distribution should be:

$$\pi(X = u) = \mu_{A\cap B}(u) = \min(\mu_A(u), \mu_B(u)) \qquad (iii)$$

But obviously (i), (ii) and (iii) cannot simultaneously hold[1] (neither (i) and (ii)). Then, to restablish consistency, one could think of two ways out. One is to look at the different possibility distributions as a result of an ongoing conditioning process:

if: $\pi(X = u \mid \text{"X is A"}) = \mu_A(u)$,
$\pi(X = u \mid \text{"X is B"}) = \mu_B(u)$
then: $\pi(X = u \mid \text{"X is A", "X is B"}) = \min(\mu_A(u), \mu_B(u))$,

and so on. The other alternative is to understand each fuzzy assertion as a constraint on the unknown or partially known possibility distribution of X, see [3]. For instance:

if: $\pi(X = u) \le \mu_A(u)$,
$\pi(X = u) \le \mu_B(u)$
then: $\pi(X = u) \le \min(\mu_A(u), \mu_B(u))$,

In this paper we shall adopt this latter approach to explore the semantics of most usual fuzzy statements (elementary canonical expressions and if-then rules), that is, which kind of constraints correspond to each kind of statement and how they are combined into new ones. Notice that

[1] Notice that this problem is independent from the fuzziness of A and B, i.e., even if A and B are classical subsets the problem remains.

under this understanding, inference in fuzzy logic is nothing but constraint propagation.
In section 2 we pay attention to the different semantics that can be attached to elementary fuzzy statements, while Section 3 is devoted to fuzzy if-then rules and their typology. Besides the conjunction and implication based models we introduce what we call belief based model of fuzzy rules. We finish with some conclusions and final remarks.

## 2 WHAT DOES "X IS A" MEAN

Most elementary expressions in fuzzy logic are of the form "X is A" where X is variable and A is a fuzzy subset representing a linguistic value. But even these simplest expressions are sometimes misleading since they can be provided with several meanings. Usually, the meaning of such fuzzy expressions is described by their induced possibility distributions, or better, by their induced constraints on the unknown (or partially known) possibility distribution of the variable X. Therefore, from the possibility theory point of view, different meanings have to be reflected into different kinds of constraints induced, even in the non-fuzzy case.

### 2.1 THE NON FUZZY CASE

For the sake of clarity, we shall start out by considering the four following situations which can be possibly described by an expression "X is A":

1. In a sample, the variable X has been observed to take values $u_1$, ...., $u_n$, and A is the set $\{u_1, .., u_n\}$.
2. The variable X has taken some values in an imprecise range A.
3. The set of possible values the variable X can take are all in A.
4. The variable X has taken one value, but all we know is that the value belongs to some set A.

Now the question is which kind of constraints induce each of the above informations in the possibility distribution $\pi_X$ of the variable X, how they are combined (separately) and which "specificity principles" apply to these different representations.
The constraints induced by these informations depend on which kind of hypothesis we make on the variable X. In situations (1), (2) and (3) the variable is supposed to take a range of values ( in repeated experiences), and we are interested in knowing about this range.
In case (1) the information tells us that all the values of A have to be fully possible, since they have been observed, and moreover, they don't exclude in principle any other (possibly different) value. This is reflected by the constraint:

$$\pi_X(u) \geq \mu_A(u), \text{ for all } u \qquad \text{(Type I)}$$

Observe that u belongs to A, $\mu_A(u) = 1$, then necessarily $\pi_X(u) = 1$, i.e. u is a possible value for X, while elements outside A don't perform any constraint on $\pi_X$.
However information (2) only reflects that some of the elements of A are possible values for X, and again does not exclude any other possible value. This constraint is expressed by

$$\pi_X \cap \mu_A \neq \emptyset, \qquad \text{(Type II)}$$

in the sense that at least there must exist some $x \in A$ such that $\pi_X(u) = 1$.
In situation (3), we are saying that the whole range of possible values for X is included in the set A. This basically tell us that what we know for sure is that elements outside A are complete impossible for X. Then the induced constraint is:

$$\pi_X(u) \leq \mu_A(u), \text{ for all } u \qquad \text{(Type III)}$$

under the assumption that $\pi_X$ is normalized. In this case, notice that if x does not belong to A, i.e. $\mu_A(u) = 0$, then necessarily $\pi_X(u) = 0$, i.e. x is an impossible value for X.
Finally, situation (4) is similar to (3) but in (4) we are interested in one single value of the variable, and we have only some imprecise information about the value the variable has taken. In this case this is reflected by the compound constraint:

$$\pi^*_X(u) \leq \mu_A(u), \text{ for all } u \qquad \text{(Type IV)}$$

where $\pi^*_X$ stands for a singleton
These four interpretations of the statement "X is A" correspond, respectively, to the following assignments of different possibilistic measures and to the following set relationships:

| **Descriptive measure** | **Set relationships** |
|---|---|
| Type I: | |
| $\Delta_{\pi_X}(A) = 1$ (guaranteed possibility) | $rang(X) \supseteq A$ |
| Type II: | |
| $Pos_{\pi_X}(A) = 1$ (possibility) | $rang(X) \cap A \neq \emptyset$ |
| Type III: | |
| $Nec_{\pi_X}(A) = 1$ (necessity) | $rang(X) \subseteq A$ |
| Type IV: | |
| $Nec_{\pi^*_X}(A) = 1$ (necessity) | $val(X) \in A$ |

where, as usual, $\Delta_{\pi_X}(A) = \inf_{u \in A} \pi_X(u)$, $Pos_{\pi_X}(A) = \sup_{u \in A} \pi_X(u)$, and $Nec_{\pi_X}(A) = \inf_{u \notin A} 1 - \pi_X(u)$. Also, rang(X) and val(X) denote the rang of values and the current value of X respectively.

### 2.2 THE FUZZY CASE

As natural generalizations of the above crisp cases, the possible meanings of the expression "X is A", when A is

fuzzy, together with their induced constraints on the possibility distribution of the variable X, are:

Type I: The variable X has been observed to take fuzzy values $\hat{u}_1, ..., \hat{u}_n$ and A is the fuzzy union of these fuzzy values, i.e. $\mu_A(u) = \max_i \mu_{\hat{u}_i}(u)$. The corresponding constraint is:

$$\pi_X(u) \geq \mu_A(u), \text{ for all } u \in U$$

Type II: Some fuzzy values taken by the variable X are in the fuzzy range A. Then the crisp constraint $\pi_X \cap \mu_A \neq \emptyset$ can be generalized as

$$\exists u \in U, \min(\pi_X(u), \mu_A(u)) \geq \alpha > 0$$

If the fuzzy value is supposed to be normalized, then we should take $\alpha = 1$ in the above constraint.

Type III: All the fuzzy values the variable X can take are contained in A. In this case, the constraint is:

$$\pi_X(u) \leq \mu_A(u), \text{ for all } u \in U$$

Type IV: The unknown (and unique) fuzzy value taken by X is in A. It is analogous to type III, but, contrary to the non-fuzzy case, it is not clear what a "unique fuzzy value" (a fuzzy singleton) can be. Therefore, this kind of interpretation will not be considered any more.

The usual extensions to the fuzzy case of the possibilistic measures appearing in the pervious subsection are the following ones see [9] and [3]:

$$Pos_\pi(A) = \sup_{u \in U} \min(\pi(u), \mu_A(u))$$

$$Nec_\pi(A) = 1 - Pos_\pi(\neg A) = \inf_{u \in U} \max(1-\pi(u), \mu_A(u))$$

$$Nec^*_\pi(A) = \inf_{u \in U} I(\pi(u), \mu_A(u))$$

$$\Delta_\pi(A) = \inf_{u \in U} I(\mu_A(u), \pi(u))$$

where I is an implication function verifying $I(x, y) = 1$ iff $x \leq y$. Notice that when A is crisp all of them coincide with the expressions of the end of subsection 1.1. Using these measures, we can characterize the above constraints as follows:

| **Descriptive measure** | **Set relationships** |
|---|---|
| Type I: | |
| $\Delta_\pi(A) = 1$ | $rang(X_\alpha) \supseteq A_\alpha, \forall \alpha \in [0, 1]$ |
| Type II: | |
| $Pos_\pi(A) = 1$ ($\alpha = 1$) | $rang(X_1) \cap A_1 \neq \emptyset$ |
| Type III: | |
| $Nec^*_\pi(A) = 1$ | $rang(X_\alpha) \subseteq A_\alpha, \forall \alpha \in [0, 1]$ |

where $rang(X_\alpha)$ and $A_\alpha$ denote the $\alpha$-cuts of $\pi_X$ and $\mu_A$ respectively. Notice that condition defining type II and type III are not duals as they are in the classical case. The negation of the condition of type III is equivalent to the existence of a value $\alpha$ such that $rang(X_\alpha) \not\subset A_\alpha$ which is equivalent to $rang(X_\alpha) \cap \neg A_\alpha \neq \emptyset$ but this is not equivalent to the condition of type II that requires that the inequality has to be true for all $\alpha \in [0, 1]$.

## 2.3 CONSTRAINT COMBINATION

From the point of view of combining constraints, we focus on types I and III, since combining constraints of type II does not lead to interesting results.

Type I. With this kind of constraints, if we have that both "X is A" and "X is B", it means that X has taken fuzzy values which fuzzy union are A and B, that is, which fuzzy union is $A \cup B$. This is in accordance with inferring the constraint

$$\pi_X(u) \geq \max(\mu_A(u), \mu_B(u)) = \mu_{A \cup B}(u), \text{ for all } u$$

from the constraints $\pi_X \geq \mu_A$ and $\pi_X \geq \mu_B$.

Type III. With the third type, having both "X is A" and "X is B" it means that we know that X has taken fuzzy values contained in A and B, i.e. contained in $A \cap B$. Again, this is complete accordance with inferring the new constraint

$$\pi_X(u) \leq \min(\mu_A(u), \mu_B(u)) = \mu_{A \cap B}(u), \text{ for all } u$$

from $\pi_X \leq \mu_A$ and $\pi_X \leq \mu_B$.

So, "max" and "min" are the natural combination operators for constraints corresponding to the first and third type respectively.

## 2.4 SPECIFICITY PRINCIPLES

If we have a constraint on a possibility distribution that does not completely specifies a unique distribution, the question arises when we want (or need) to choose one of them. The most natural principle is to choose that one which conveys less informational content.

Under the first interpretation of "X is A" (Type I) as

$$\pi_X \geq \mu_A, \qquad (*)$$

notice that the bigger is A, the more informative is "X is A", while under the third interpretation (Type III) as

$$\pi_X \leq \mu_A, \qquad (**)$$

is in other way round, the smaller is A, the more informative is the expression "X is A". Therefore, given that the only information we know about X is "X is A", under the first interpretation, the less informative distribution which is in accordance with (*) will be the most specific, i.e. $\pi_X = \mu_A$. This is the *maximum specificity principle* (Max-SP) which allows us to momentaneously assume as possible values for X those outside A. Dually, under the Type III interpretation of "X is A, the less informative possibility distribution satisfying (**) is also $\pi_X = \mu_A$, i.e. the less specific distribution satisfying (**). This the so-called *minimum*

*specificity principle* (Min-SP). Now, in contraposition to Max-SP, the Min-SP assumes all elements of A fully possible values for X.

With respect to the second type of interpretations (type II), it is not clear whether such specificity principles may be of use: namely the less informative possibility distribution satisfying $\pi_X \cap A \neq \emptyset$ doesn't exist, all the distributions representing singletons of A are minimal w.r.t. the information ordering.
Notice the defeasible character of the consequences derived on the basis of the Max-SP or the Min-SP, since if further information is known nothing prevents to retract some of the consequences.

# 3 FUZZY "IF THEN" RULES: FROM DATA TO KNOWLEDGE.

Most simple conditional statements in fuzzy logic have the form

If "X is A" then "Y is B"

In this section we shall discuss different semantics we can attach to such fuzzy statements. We start with the non-fuzzy case. Let us suppose there is a sample $S = \{(x_1, y_1), ..., (x_n, y_n)\}$ of pairs of values for variables X and Y. Suppose all we know from this sample is the set of values taken by X, i.e. $A = \{x_1, ..., x_n\}$, and all the values taken by Y, i.e. $B = \{y_1, ..., y_n\}$. We can describe this information (data) by the following conjunctive statement:

"X has taken values in A" *and* "Y has taken values in B"

which, in turn, can be equivalently expressed as the elementary expression

"(X, Y) has taken values in A×B"

involving the two-dimensional variable (X, Y). Now, let us suppose the variables X and Y have been observed to take fuzzy values $\hat{u}_1, ..., \hat{u}_n$ and $\hat{v}_1, ..., \hat{v}_n$ respectively, and let A and B be respectively the fuzzy union of these sets of fuzzy values, i.e.
$\mu_A(u) = \max_i \mu_{\hat{u}_i}(u)$ and $\mu_B(v) = \max_i \mu_{\hat{v}_i}(v)$.
Analogously to the non-fuzzy case, we can summarize the information as two fuzzy statements of type I (see section 2.2)

"X has taken fuzzy values in A" *and*
"Y has taken fuzzy values in B"

and thus its corresponding meaning is given by the following two constraints:

$$\pi_X \geq \mu_A \text{ and } \pi_Y \geq \mu_B$$

The fuzzy values in A and in B have been observed, so they have to be possible. In other words, if $\pi_{X\,Y}$ represents the joint possibility distribution of the joint variable (X,Y) we have the following constraint:

$$\pi_{XY} \geq \mu_{A\times B} \quad \text{(CbM)}$$

where $\mu_{A\times B}(u, v) = \min(\mu_A(u), \mu_B(v))$. This interpretation leads to the so-called *conjunction-based models of fuzzy rules*.
However, if we want to use this information to make inference, we have to *generalize* in order to get some "generic knowledge" to describe the "conditional" behaviour of variable Y given some knowledge about variable X and we will write

If X takes value in A, then Y takes value in B

in the sense that we are now supposing that this piece of knowledge will be always valid, i.e. in all future samplings. This interpretation leads either to the *implication-based rules* or what we call *belief-based* rules. Namely, the traditional representation of a fuzzy rule, in terms of the constraint induced on the joint possibility distribution of the two dimensional variable (X, Y), is given by

$$\pi_{X,Y}(u, v) \leq I(\mu_A(u), \mu_B(v)), \forall(u,v)\in UxV \quad \text{(IbM)}$$

where I: $[0, 1]x[0,1] \rightarrow [0,1]$ is a many-valued implication function. These are the *implication-based models* of fuzzy rules. But there is still another interpretation which take into account, not the joint possibility distribution for (X,Y), but the marginals instead:

$$\text{if } \pi_X \leq \mu_A \text{ then } \pi_Y \leq \mu_B \quad \text{(BbM)}$$

This interpretation leads to what we call *belief-based models* of fuzzy rules. The reason for that name will become clearer below.
The different types of fuzzy rules corresponding to conjunction-based and implication-based models are summarized in the next two subsections, following [4] and [5]. In the last subsection we discuss the representation of the belief-based model of fuzzy rules.

## 3.1 CONJUNCTION-BASED MODELS OF FUZZY RULES

Within the conjunction-based model (CbM), the fact that X moves away from the core of A naturally leads to reduce the minimal level of guaranteed possibility. Namely, if X = u, "Y is B" is declared to be possible at least at level $\mu_A(u)$. This corresponds to the kind of rule "the more X is A, the more possible Y is B", that are called ***possibility rules***. The representation of possibility-qualified statements leads, according to

Mamdani in [7], to the following constraint on the joint possibility distribution :

$$\pi_{X,Y}(u, v) \geq \min(\mu_A(u), \mu_B(v)) \qquad \text{(i)}$$

In [5] Dubois and Prade introduce a second kind of conjunction-based model of fuzzy rule by considering this interpretation: if X takes value in the core of A then all values in the support of B are fully possible for Y, while the subset of values with some guaranteed possibility for Y becomes smaller as X moves away from the core of A. This corresponds to a rule of the type "the more X is A, and the less Y is related to X, the less Y is B". Such rules are called ***anti-gradual rules*** and are interpreted by the following inequality

$$\pi_{X,Y}(u, v) \geq \mu_A(u) *_T \mu_B(v), \qquad \text{(ii)}$$

where $*_T$ is the adjoint of a strong implication. Notice that (ii) is similar to (i) since $*_T$ is a non-commutative many-valued conjunction such that $0 *_T 0 = 1 *_T 0 = 0 *_T 1 = 0$ and $1 *_T 1 = 1$. For instance, when T = min, we get

$$\pi_{X,Y}(u, v) \geq \begin{cases} 0, \text{ if } \mu_A(u) + \mu_B(v) \leq 1 \\ \mu_B(v), \text{ otherwise} \end{cases}$$

## 3.2 IMPLICATION-BASED MODELS OF FUZZY RULES

It is self evident that, depending on the particular chosen implication function, and thus on the particular properties of it, the meaning of such a conditional statement may vary, and indeed it does. Main types of fuzzy rules correspond to S-implications and R-implications which lead to the so-called *certainty* and *gradual rules* respectively. The intended semantics of certainty rules can be stated by the expression, "the more X is A, the more certain that Y lies in B" while the semantics of gradual rules corresponds to the expression, "the more X is A, the more Y is B". The first type of rules provides a lower bound of the certainty of B in terms of the truth value of A while the second type establish a relationship between the corresponding $\alpha$-cuts of the fuzzy sets A and B. If the input is crisp ($x = u_0$) the constraints of the output are:

***Certainty rules***: $\pi_{X,Y}(u_0, v) \leq I_{KD}(\mu_A(u_0), \mu_B(v))$, where $I_{KD}$ is Kleene-Dienes implication, i.e. $I_{KD}(x, y) = \max(1-x, y)$. In [5] it is shown that this constraint can be interpreted as saying that a (suitable) certainty degree of Y lying in B is lower bounded by $\mu_A(u_0)$.

***Gradual rules:*** $\pi_{X,Y}(u_0, v) \leq I_G(\mu_A(u_0), \mu_B(v))$, where $I_G$ denotes Gödel's implication, i.e. $I_G(x, y) = 1$ if $x \leq y$, $I_G(x, y) = 0$ otherwise. Also in [5] it is shown that this constraint can be interpreted by saying that "y is B" is at least $\mu_A(u_0)$-true. In this case if we take the crisp relation closest to $\pi_{X,Y}(u_0, v)$ i.e.,

$$\begin{aligned} &\pi_{XY}(u_0, v) \leq 1 \quad \text{if } \mu_A(u_0) \leq \mu_B(v) \\ &\pi_{XY}(u_0, v) = 0 \quad \text{otherwise} \end{aligned}$$

then the corresponding relation between variables can be expressed in terms of $\alpha$-cuts as:

if x belongs to $A_\alpha$, then y has to belong to $B_\alpha$ (gradual implication) (see, [4] and [2]).

More general expressions of certainty and gradual rules can be obtained by substituting $I_{KD}$ by any strong implication function and $I_G$ by any residuated implication function respectively.

## 3.3 BELIEF-BASED RULES

Conjunction-based and implication-based models of fuzzy rules are the most usual ones, but, as already mentioned before, there is another interpretation which leads to other possible representations for a fuzzy rule. Let us consider for a moment the case of a non-fuzzy rule " if X is A then Y is B", i.e., when A and B are crisp subsets. Then, the components of the rule "X is A", and "Y is B" can be represented by the following constraints:

$$\begin{aligned} &\text{C1:} \quad (\forall u)(\pi_X(u) \leq \mu_A(u)), \\ &\text{C2:} \quad (\forall v)(\pi_Y(v) \leq \mu_B(v)). \end{aligned}$$

Accordingly, the above rule admits the following interpretation:

C1 *implies* C2 (*)

which, in turn, has the following easy possibilistic interpretation:

"*if* A is certain ($N_{\pi_X}(A) = 1$) *then*
B is certain ($N_{\pi_Y}(B) = 1$)"

where $N_\pi$ denotes the necessity measure generated by the possibility distribution $\pi$.

Now, let us generalize this interpretation to the fuzzy case. Basically, substituting the classical quantifiers "for all" and "exists" by "inf" and "sup" respectively, and the classical inequality "$\leq$" by a many-valued residuated implication $\rightarrow$, the expression (*) becomes

$$\inf_u \pi_X(u) \rightarrow \mu_A(u) \leq \inf_v \pi_Y(v) \rightarrow \mu_B(v)$$

Observe that $\text{Cert}(A \mid \pi_X) = \inf_u \pi_X(u) \rightarrow \mu_A(u)$ and $\text{Cert}(B \mid \pi_Y) = \inf_v \pi_Y(v) \rightarrow \mu_B(v)$ are certainty-like indices. In these terms, expression (*) means that

"Y is B" is as certain as "X is A".

Finally, let us show how this model of fuzzy rules behave in front of a fuzzy input "X is $A_1$", again understood as the constraint $(\forall u)(\pi_X(u) \leq \mu_{A_1}(u))$. Since

$$\inf_u \pi_X(u) \rightarrow \mu_A(u) \leq \inf_v \pi_Y(v) \rightarrow \mu_B(v),$$

we have that

$$\pi_Y(v) \leq [\inf_u \pi_X(u) \rightarrow \mu_A(u)] \rightarrow \mu_B(v), \text{ for all } v$$

but since $\pi_X \leq \mu_{A_1}$, we finally have

$$\pi_Y(v) \leq [\inf_u \mu_{A_1}(u) \rightarrow \mu_A(u)] \rightarrow \mu_B(v) =$$
$$= \text{Cert}(A \mid A_1) \rightarrow \mu_B(v)$$

In other words, we can formulate the following generalized modus ponens pattern

If "X is A" then "Y is B"
"X is $A_1$"
---
"Y is $B_1$"

in the sense that from

$$\inf_u \pi_X(u) \rightarrow \mu_A(u) \leq \inf_v \pi_Y(v) \rightarrow \mu_B(v),$$
$$(\forall u)(\pi_X(u) \leq \mu_{A_1}(u))$$

we can infer

$$(\forall v)(\pi_Y(v) \rightarrow \mu_{B_1}(v)),$$

where $\mu_{B_1}(v) = \text{Cert}(A \mid A_1) \rightarrow \mu_B(v)$.

This interpretation has been considered in [8] and in [1] from different approaches and recently in [6] from a formal logical (many-valued) point of view.

# 4 CONCLUSIONS

In this paper we have been concerned with the possible semantics of statements like "X is A" and "If-then" rules in terms of the constraints induced on the possibility distributions of the vriables invoced. Although we have not dealt with negated statements but it is interesting to remark that the representation of "X is A" as $(\forall u)(\pi_X(u) \leq \mu_A(u))$ leads to two interpretations of negation that are of worth considering:

- "X is not-A", i.e. $(\forall u)(\pi_X(u) \leq 1 - \mu_A(u))$
- not "X is A", i.e. $(\exists u)(\pi_X(u) > \mu_A(u))$

Notice that in the crisp case, if we identify $\pi_X$ with its corresponding subset, the two types of negation of $X \subset A$ correspond to $X \subseteq A^c$ and $X \not\subseteq A$ respectively and both expressions become equivalent only if X stands for a singleton. Obviously the two interpretations are not equivalent in the general fuzzy case. The study of the different semantics provided by these two interpretations of the negation will be a matter of future works.

**References**


[1] D. Boixader, J. Jacas (1996). CRI as approximate reasoning tool: an analysis via T-indistinguishability operators. Proc. Fuzz-ieee'96 (New Orleans), pp. 2094-2097.

[2] D. Dubois, M. Grabisch, H. Prade (1994). Gradual rules and the approximation of control laws. In: Theoretical Aspects of Fuzzy Control (H.T. Nguyen, M. Sugeno, R. Tong, R.R. Yager, eds.), Wiley, New York, pp. 147-181.

[3] D. Dubois, H. Prade (1991). Fuzzy sets in approximate reasoning - Part 1: Inference with possibility distributions. Fuzzy Sets and Systems Vol.40, pp.143-202.

[4] D. Dubois, H. Prade (1992). Gradual inference rules in approximate reasoning. Information Sciences Vol. 61, pp. 103-122.

[5] D. Dubois, H. Prade (1996). What are fuzzy rules and how to use them. Fuzzy Sets and Systems Vol. 84, pp. 169-185.

[6] L. Godó, P. Hájek (1997). Fuzzy Inference as Deduction. To appear in the Journal of Applied non-Classical Logics.

[7] E.H. Mamdani (1977). Application of fuzzy logic to approximate reasoning using linguistic systems. IEEE Trans. Comput. Vol. 26, pp. 1182-1191.

[8] P. Magrez, P. Smets (1989). Fuzzy modus ponens: a new model suitable for applications in knowledge-based systems. Int. J. of Intelligent Systems, 4, 181-200.

[9] L.A. Zadeh (1978). Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems, 1, 3-28.

# UN MODELO MATEMÁTICO DEL "Y" EN LA LÓGICA BORROSA

**Enric Trillas**
Dpto. Inteligencia Artificial.
Universidad Politécnica de Madrid.
28660 Boadilla del Monte.
etrillas@fi.upm.es

**Ana Pradera**
Dpto. Ciencias de la Computación.
Universidad de Alcalá.
28871 Alcalá de Henares.
ccapg@cc.alcala.es

**Susana Cubillo**
Dpto. Matemática Aplicada.
Universidad Politécnica de Madrid.
28660 Boadilla del Monte.
scubillo@fi.upm.es

## Resumen

La modelización matemática de los conectivos lógicos es uno de los problemas fundamentales de la lógica borrosa teórica. Además de las t-normas y t-conormas, se dispone actualmente de una gran variedad de operadores de agregación que proporcionan formas más flexibles para la combinación de subconjuntos borrosos.

El presente trabajo propone un modelo matemático para el uso de diversas modalidades del "y" en lógica borrosa, y establece distintas caracterizaciones que permiten averiguar en qué subconjunto de $[0,1]^2$ un operador de agregación se comporta como una conjunción. Los resultados obtenidos se aplican, además de a las t-normas, a dos casos concretos: las medias cuasi-lineales [1] y las medias ponderadas ordenadas [9].

**Palabras Clave**: Conjunciones. Agregaciones. T-normas. Medias cuasi-lineales. Medias ponderadas ordenadas.

## 1 INTRODUCCIÓN

Así como en lógica clásica los conectivos lógicos se suponen definidos de forma única, en los entornos del razonamiento aproximado la experiencia ha demostrado que la forma de interpretar dichas operaciones - y por lo tanto la forma de definirlas - varía en función del contexto. En general, el "y" se refiere a una fusión de información. Tal fusión puede ser *interna*, como en el caso, por ejemplo, de " $x$ es $P$ $y$ $z$ es $Q$ ", o *externa*, como en el de "$x$ es $P$ $y$ si $x$ es $P$, entonces $z$ es $Q$ ". En cada uno de estos casos el modelo matemático elegido para representar el "y" depende, en las aplicaciones, tanto del contexto informacional como del objetivo perseguido en la afirmación resultante.

Es por ello que la definición y estudio de estos conectivos constituye una de las actividades fundamentales de la investigación en lógica borrosa teórica, que ha producido, desde sus orígenes, por lo menos tres modelos básicos, que, citados de forma cronológica según su aparición en la literatura, son los siguientes: los **operadores borrosos estándar** (definidos por Zadeh como el mínimo para la conjunción y el máximo para la disyunción), las ampliamente usadas **t-normas y t-conormas** y los **operadores de agregación**. Estos últimos surgen para expresar la idea de agregación, compensación o promediación de valores, conceptos éstos que aparecen frecuentemente en el mundo real. Son operadores definidos de $[0,1]^n$ en $[0,1]$ continuos, monótonos y con las condiciones de contorno $h(0,..,0) = 0$ y $h(1,..,1) = 1$. En los últimos años han aparecido numerosas propuestas y estudios sobre operadores de este tipo (véase por ejemplo [5], [9], [2], [6]). Constituyen un modelo muy general que no distingue claramente, como los otros dos, entre conjunción y disyunción.

En [8], Trillas, Cubillo y Castro proponen sendas definiciones para modelizar conjunciones y disyunciones borrosas, y demuestran que el nuevo modelo incluye tanto el caso clásico como los dos primeros modelos borrosos citados más arriba. Comprueban sin embargo que otros operadores más generales - por ejemplo las medias cuasi-lineales - no siempre se comportan como conjunciones o disyunciones, e introducen entonces el concepto de conjunción/disyunción local en un punto.

El presente trabajo parte de los conceptos y resultados expuestos en [3] y [8], generalizando la definición de conjunción a conjunción en un subconjunto cualquiera de $[0,1]^2$ y estudiando en qué subconjuntos los operadores de agregación se comportan como conjunciones.

A lo largo de este trabajo se designará por $I$ al intervalo $[0,1]$, y, para todo $A \subseteq I^2$, $A \neq \varnothing$, y cada $a \in I$ será $A_a = \{b \in I : (a,b) \in A \text{ o } (b,a) \in A\}$. Además, para cada función $\& : I^2 \to I$ y para cada $a \in I$, la

sección vertical de & en $a$ será la función $\&_a : I \to I$ dada por $\&_a(b) = \&(a,b)$ para todo $b$ de $I$.

# 2 CONJUNCIONES Y CONJUNCIONES ESTRICTAS

El presente apartado amplía los conceptos de conjunción y conjunción estricta, propuestos en [8], a un subconjunto cualquiera de $I^2$ y obtiene distintas caracterizaciones con respecto a dicho subconjunto. Se estudia asimismo el caso en el que el subconjunto coincide con todo $I^2$.

## 2.1 CONJUNCIONES

**Definición 2.1** Dado un subconjunto $A \subseteq I^2$, una función $\& : I^2 \to I$ *es una conjunción en* $A$ si para todo $(a,b)$ de $A$ existe $c \in I$ tal que $\&(\max(a,b),c) = \min(a,b)$.

**Nota 1** Si una función $\& : I^2 \to I$ es conjunción en un punto $(a,b)$ de $I^2$ también lo es en $(b,a)$, por lo que el mayor subconjunto $A \subseteq I^2$ en el que una función es conjunción constituye una región simétrica respecto a la recta $a = b$, lo que ni significa ni supone que & sea conmutativa.

**Teorema 2.1** *Dado un subconjunto* $A \subseteq I^2$, *una función* $\& : I^2 \to I$ *es una conjunción en* $A$ *si y sólo si se verifica* $[0,a] \cap A_a \subseteq \mathrm{Im}(\&_a)$ *para todo* $a$ *de* $I$.

**Demostración.**

$\Longrightarrow$ Sea $a \in I$ tal que $A_a \neq \varnothing$ (si $A_a = \varnothing$ la inclusión es trivial). Si $x \in [0,a] \cap A_a$, se tiene que $x \leq a$ y $(x,a) \in A$ o $(a,x) \in A$, con lo que existirá un $c \in I$ tal que $\&(a,c) = x$ y por lo tanto $x \in \mathrm{Im}(\&_a)$.

$\Longleftarrow$ Sea $(a,b) \in A$. Si $a \leq b$, se tiene que $a \in A_b \cap [0,b] \subseteq \mathrm{Im}(\&_b)$, y por lo tanto existirá un $c \in I$ tal que $\&_b(c) = \&(b,c) = a$. En el caso $a > b$ la demostración es análoga tomando como punto de partida $b \in A_a \cap [0,a]$.□

Como corolario inmediato se llega a la caracterización de conjunción en todo $I^2$ obtenida en el artículo [8].

**Corolario 2.2** *Una función* $\& : I^2 \to I$ *es una conjunción en* $I^2$ *si y sólo* $[0,a] \subseteq \mathrm{Im}(\&_a)$ *para todo* $a$ *de* $I$.

El siguiente teorema expresa una condición suficiente para que una función continua sea una conjunción.

**Teorema 2.3** *Dado un subconjunto* $A \subseteq I^2$, *si una función* $\& : I^2 \to I$ *es continua en la segunda variable y además verifica:* $\forall (a,b) \in A$, $\exists a_1, a_2 \in I$ :

$$\&(\max(a,b),a_1) \leq \min(a,b) \leq \&(\max(a,b),a_2)\ ,$$

& *es una conjunción en* $A$.

**Demostración.** Basta con aplicar las propiedades de continuidad.□

Del teorema anterior se deduce la siguiente condición suficiente para que una función & sea conjunción en $I^2$.

**Corolario 2.4** *Si una función* $\& : I^2 \to I$ *es continua en la segunda variable y además verifica que:* $\forall a \in I$ $\exists a_1, a_2 \in I : \&(a,a_1) = 0$ *y* $\&(a,a_2) \geq a$, & *es una conjunción en* $I^2$.

El siguiente teorema expresa una condición necesaria para que una función monótona sea una conjunción.

**Teorema 2.5** *Sea* $\& : I^2 \to I$ *una función que es monótona no decreciente en la segunda variable y conjunción en un subconjunto* $A \subseteq I^2$. *Entonces* & *cumple:* $\&(\max(a,b),0) \leq \min(a,b) \leq \&(\max(a,b),1)$ *para todo* $(a,b)$ *de* $A$.

**Demostración.** Basta con considerar la definición de conjunción y aplicar las propiedades de monotonía. □

Como consecuencia de lo anterior se obtiene la condición necesaria dada en [8] para que una función monótona sea conjunción en $I^2$.

**Corolario 2.6** *Sea* $\& : I^2 \to I$ *una función que es monótona no decreciente en la segunda variable y conjunción en* $I^2$. *Entonces* $\&(a,0) = 0$ *y* $\&(a,1) \geq a$ *para todo* $a$ *de* $I$.

## 2.2 CONJUNCIONES ESTRICTAS

**Definición 2.2** Dado un subconjunto $A \subseteq I^2$, una función $\& : I^2 \to I$ *es una conjunción estricta en* $A$ si se cumple:

(i) & es una conjunción en $A$.

(ii) Para cualquier $(a,b)$ de $A$, si $a \neq b$ no existe ningún $d \in I$ tal que $\&(\min(a,b),d) = \max(a,b)$.

**Teorema 2.7** *Dado un subconjunto* $A \subseteq I^2$, *una función* $\& : I^2 \to I$ *es conjunción estricta en* $A$ *si y sólo si se verifica* $[0,a] \cap A_a = \mathrm{Im}(\&_a) \cap A_a$ *para todo* $a$ *de* $I$.

**Demostración.**

$\Longrightarrow$ Sea $a \in I$ tal que $A_a \neq \varnothing$ (si $A_a = \varnothing$ la igualdad del teorema es trivial). La inclusión $[0,a] \cap A_a \subseteq$

$\mathrm{Im}(\&_a) \cap A_a$ es consecuencia del teorema 2.1. Por otro lado, si $b \in \mathrm{Im}(\&_a) \cap A_a$, existirá un $c \in I$ tal que $\&_a(c) = \&(a,c) = b$. Pero al ser $b \in A_a$, $\&$ es conjunción estricta en $(a,b)$ o $(b,a)$, y por lo tanto no puede existir ningún $d \in I$ tal que $\&(\min(a,b),d) = \max(a,b)$. De ambas igualdades se deduce $\min(a,b) \neq a$, y por lo tanto $b \in [0,a]$.

$\Longleftarrow$ La propiedad (i) es evidente puesto que por el teorema 2.1 $\&$ es conjunción. Para demostrar (ii), supóngase que $a \neq b$ y existe un $d \in I$ tal que $\&(\min(a,b),d) = \max(a,b)$, que implica que $\max(a,b) \in \mathrm{Im}(\&_{\min(a,b)})$. Como además $\max(a,b) \in A_{\min(a,b)}$ por ser $(a,b) \in A$, se tendría $\max(a,b) \in \mathrm{Im}(\&_{\min(a,b)}) \cap A_{\min(a,b)} = [0,\min(a,b)] \cap A_{\min(a,b)}$, y se llegaría a la contradicción $\max(a,b) \leq \min(a,b)$. □

**Nota 2** En la caracterización de conjunción estricta del teorema anterior la fórmula $[0,a] \cap A_a = \mathrm{Im}(\&_a) \cap A_a$ es equivalente a "$\&$ es conjunción en $A$ y además para todo $a, c$ de $I$, si $\&(a,c) \in A_a$ entonces $\&(a,c) \leq a$".

Como corolario inmediato se obtiene la caracterización de conjunción estricta en $I^2$ propuesta en el artículo [8].

**Corolario 2.8** *Una función $\& : I^2 \to I$ es una conjunción estricta en $I^2$ si y sólo $[0,a] = \mathrm{Im}(\&_a)$ para todo $a$ de $I$ (es decir, si y sólo si $\&$ es conjunción en $I^2$ y además para todo $a, c$ de $I$, $\&(a,c) \leq a$).*

El siguiente teorema expresa una condición suficiente para que una función $\&$ continua sea conjunción estricta en un subconjunto $A \subseteq I^2$.

**Teorema 2.9** *Dado un subconjunto $A \subseteq I^2$, si una función $\& : I^2 \to I$ es continua en la segunda variable y además verifica:*

- $\forall (a,b) \in A$, $\exists a_1, a_2 \in I$ :
  $\&(\max(a,b),a_1) \leq \min(a,b) \leq \&(\max(a,b),a_2)$.
- $\forall a, c \in I$, *si* $\&(a,c) \in A_a$ *entonces* $\&(a,c) \leq a$ ,

*$\&$ es una conjunción estricta en $A$.*

**Demostración.** Es evidente teniendo en cuenta la condición suficiente para conjunción demostrada en el teorema 2.3 y la nota 2.□

De la particularización del teorema anterior al caso $A = I^2$ se obtiene:

**Corolario 2.10** *Si una función $\& : I^2 \to I$ es continua en la segunda variable y además verifica:*

- $\forall a \in I$, $\exists a_1, a_2 \in I : \&(a,a_1) = 0$ *y* $\&(a,a_2) = a$.
- $\forall a, c \in I$, $\&(a,c) \leq a$ ,

*$\&$ es una conjunción estricta en $I^2$.*

El siguiente teorema expresa una condición necesaria para que una función monótona y continua sea conjunción estricta.

**Teorema 2.11** *Sea $\& : I^2 \to I$ una función que es monótona no decreciente, continua en la segunda variable, y conjunción estricta en un subconjunto $A \subseteq I^2$. Entonces para cualquier $(a,b) \in A$ se cumple:*

- $\&(\max(a,b),0) \leq \min(a,b) \leq \&(\max(a,b),1)$.
- *si $a \neq b$, entonces* $\&(\min(a,b),1) < \max(a,b)$.

**Demostración.** La primera propiedad es condición necesaria para que $\&$ sea conjunción en $A$ (teorema 2.5). Para demostrar la segunda, supóngase que existe $(a,b)$ tal que $a \neq b$ y $\&(\min(a,b),1) \geq \max(a,b)$. Claramente $\&$ no es conjunción estricta si se cumple la igualdad. Si es $\&(\min(a,b),1) > \max(a,b)$, por ser $\&$ monótona y conjunción se tiene $\&(\min(a,b),0) \leq \&(\max(a,b),0) \leq \min(a,b) \leq \max(a,b)$. De ambas desigualdades se obtiene la cadena $\&(\min(a,b),0) \leq \max(a,b) < \&(\min(a,b),1)$, que, por ser $\&$ continua, implica que debe existir un $c \in I$ tal que $\&(\min(a,b),c) = \max(a,b)$, con lo que se llega a que $\&$ no es conjunción estricta. □

Se obtiene como corolario de este teorema la condición necesaria dada en [8] para conjunciones estrictas en $I^2$.

**Corolario 2.12** *Sea $\& : I^2 \to I$ una función monótona no decreciente y conjunción estricta en $I^2$. Entonces $\&(a,0) = 0$ y $\&(a,1) = a$ para todo $a$ de $I$.*

# 3 EL CASO DE LOS OPERADORES DE AGREGACIÓN

El presente apartado está dedicado a estudiar cuándo los operadores de agregación de dos variables son conjunciones. Se obtienen los subconjuntos de $I^2$ en los que un operador de este tipo es conjunción y conjunción estricta.

A partir de las propiedades dadas en [4] se puede establecer la siguiente definición de operador de agregación para el caso $n = 2$ :

**Definición 3.1** Una función $h : I^2 \to I$ es un *operador de agregación* si cumple las siguientes propiedades:

1. $h(0,0) = 0$ y $h(1,1) = 1$ (condiciones de contorno)

2. $h(a,b) \geq h(c,d)$ para todo $a,b,c,d \in I$ tal que $a \geq c$, $b \geq d$ (monotonía)

3. $h$ es continua

**Teorema 3.1** *Dado un subconjunto $A \subseteq I^2$, un operador de agregación $h : I^2 \to I$ es una conjunción en $A$ si y sólo si $A \subseteq \{(a,b) \in I^2 : h(\max(a,b),0) \leq \min(a,b) \leq h(\max(a,b),1)\}$.*

**Demostración.** Al ser $h$ una función monótona no decreciente y continua, la equivalencia anterior viene dada por los teoremas 2.5 y 2.3.□

**Corolario 3.2** *Un operador de agregación $h : I^2 \to I$ es una conjunción en $I^2$ si y sólo si para todo $a$ de $I$ se tiene $h(a,0) = 0$ y $h(a,1) \geq a$.*

**Teorema 3.3** *Dado un subconjunto $A \subseteq I^2$, un operador de agregación $h : I^2 \to I$ es una conjunción estricta en $A$ si y sólo si $A \subseteq \{(a,b) \in I^2 : h(\max(a,b),0) \leq \min(a,b) \leq h(\max(a,b),1)$ y si $a \neq b$, $h(\min(a,b),1) < \max(a,b)\}$.*

**Demostración.** Al ser $h$ una función monótona y continua, la condición necesaria viene dada por el teorema 2.11. Para demostrar la implicación inversa, el último teorema prueba que $h$ es conjunción. Además, si $(a,b)$ es un punto tal que $a \neq b$, por ser $h$ monótona se tiene que $h(\min(a,b),d) \leq h(\min(a,b),1) < \max(a,b)$ para todo $d \in I$, y por lo tanto $h$ es conjunción estricta en $(a,b)$.□

**Corolario 3.4** *Un operador de agregación $h : I^2 \to I$ es una conjunción estricta en $I^2$ si y sólo si $h(a,0) = 0$ y $h(a,1) = a$ para todo $a$ de $I$.*

Las t-normas continuas son operadores de agregación que verifican este último corolario, y por lo tanto son conjunciones estrictas en $I^2$. Se incluye a continuación el estudio de otros dos casos concretos de operadores de agregación: las medias cuasi-lineales y las medias ponderadas ordenadas.

## 3.1 MEDIAS CUASI-LINEALES

En el presente apartado se considera una media cuasi-lineal continua $M : I^2 \to I$, definida, para cualquier $p \in (0,1)$, como $M(a,b) = f^{-1}(pf(a) + (1-p)f(b))$, siendo $f : I \to R$ una función continua, inyectiva y monótona denominada función generadora de $M$ ([1]).

Las medias así definidas son operadores de agregación, con lo que, particularizando los resultados generales obtenidos más arriba, se obtienen las siguientes caracterizaciones.

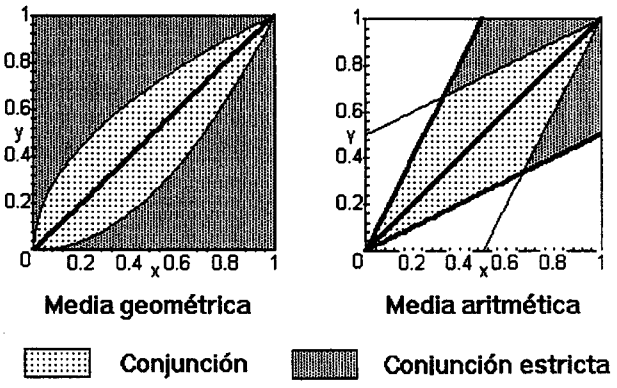



Figura 1: Algunas medias cuasi-lineales

**Teorema 3.5** *$M$ es una conjunción en $A \subseteq I^2$ si y sólo si*

$$A \subseteq \left\{ \begin{array}{c} (a,b) \in I^2 : \\ \frac{f(\min(a,b)) - pf(\max(a,b))}{1-p} \left\{ \begin{array}{c} \leq \\ \geq \end{array} \right\} f(0) \end{array} \right\}$$

*correspondiendo el signo $\leq$ al caso en que $f$ es decreciente y el signo $\geq$ al caso en que $f$ es creciente.*

**Corolario 3.6**

*Si $f$ es decreciente, $M$ es una conjunción en $I^2$ si y sólo si $f(0) = +\infty$*

*Si $f$ es creciente, $M$ es una conjunción en $I^2$ si y sólo si $f(0) = -\infty$*

**Teorema 3.7** *$M$ es una conjunción estricta en $A \subseteq I^2$ si y sólo si*

$$A \subseteq \left\{ \begin{array}{c} (a,b) \in I^2 : \\ \frac{f(\min(a,b)) - pf(\max(a,b))}{1-p} \left\{ \begin{array}{c} \leq \\ \geq \end{array} \right\} f(0) \ y \\ \left[ a = b \ \acute{o} \ \frac{f(\max(a,b)) - pf(\min(a,b))}{1-p} \left\{ \begin{array}{c} < \\ > \end{array} \right\} f(1) \right] \end{array} \right\}$$

*correspondiendo los signos $\leq$ y $<$ al caso en que $f$ es decreciente y los signos $\geq$ y $>$ al caso en que $f$ es creciente.*

**Corolario 3.8** *$M$ no es conjunción estricta en $I^2$.*

**Ejemplo 3.1** *Las medias geométrica y aritmética son dos ejemplos típicos de medias cuasi-lineales para las que se obtienen los siguientes resultados (figura 1):*

- *La media geométrica es una media cuasi-lineal con $p = 1/2$ y función generadora decreciente $f(x) = -\log x$. Es conjunción en $I^2$ (ya que $f(0) = +\infty$),*

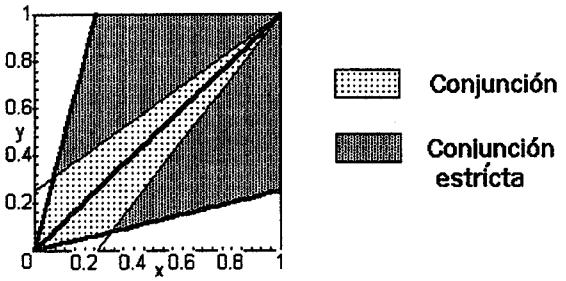



Figura 2: Bi-OWA con $\alpha = 1/4$

*y conjunción estricta en los puntos* $(a, b)$ *tales que* $a = b$ *o* $\min(a, b) < \max(a, b)$[2].

- *La media aritmética es una media cuasi-lineal con* $p = 1/2$ *y función generadora decreciente* $f(x) = 1 - x$. *Es conjunción en los puntos* $(a, b)$ *tales que* $\max(a, b) \leq 2\min(a, b)$, *y conjunción estricta en aquellos que, además, verifican* $a = b$ *o* $\max(a, b) > \frac{1+\min(a,b)}{2}$.

### 3.2 MEDIAS PONDERADAS ORDENADAS

En [9], Yager propone un tipo de operador de agregación de $I^n$ en $I$ denominado media ponderada ordenada (Ordered Weighted Averaging, OWA). La particularización de este operador al caso $n = 2$ da lugar a la siguiente definición:

**Definición 3.2** Se llama *bi-OWA* a una función $h_\alpha : I^2 \to I$ definida, para cualquier $\alpha \in I$ y para cualquier par $(a, b)$ de $I^2$, por

$$h_\alpha(a, b) = \alpha \max(a, b) + (1 - \alpha)\min(a, b).$$

Los resultados generales obtenidos previamente permiten caracterizar estos operadores de la siguiente forma (véase la figura 2 para un ejemplo con $\alpha = 1/4$).

**Teorema 3.9** *Una bi-OWA* $h_\alpha : I^2 \to I$ *es una conjunción en* $A \subseteq I^2$ *si y sólo si*

$$A \subseteq \{(0,0)\} \cup \left\{(a, b) \in I^2 : \tfrac{\min(a,b)}{\max(a,b)} \geq \alpha\right\}.$$

**Teorema 3.10** *Una bi-OWA* $h_\alpha : I^2 \to I$ *es una conjunción estricta en* $A \subseteq I^2$ *si y sólo si*

$$A \subseteq \{(0,0)\} \cup \left\{ \begin{array}{c} (a, b) \in I^2 : \tfrac{\min(a,b)}{\max(a,b)} \geq \alpha \text{ y} \\ \left[a = b \text{ ó } \alpha < \tfrac{\max(a,b)-\min(a,b)}{1-\min(a,b)}\right] \end{array} \right\}.$$

**Corolario 3.11** *Una bi-OWA* $h_\alpha : I^2 \to I$ *es una conjunción (estricta) en* $I^2$ *si y sólo si* $\alpha = 0$ *(es decir,* $h_\alpha = \min$*).*

## Referencias


[1] J. Aczél. "Lectures on Functional Equations and their Applications". *Academic Press. New York.* (1966).

[2] C. Alsina, G. Mayor, M.S. Tomás and J. Torrens. "A characterization of a class of aggregation functions". *Fuzzy Sets and Systems* 53 (1993) 33-38.

[3] S. Cubillo. "Contribución al estudio de la lógica y de los condicionales borrosos". *Tesis Doctoral. Universidad Politécnica de Madrid.* (1993).

[4] D. Dubois and H. Prade. "A Review of Fuzzy Set Aggregation Connectives". *Information Sciences* 36 (1985) 85-121.

[5] G. Mayor and E. Trillas. "On the representation of some aggregation functions". *Proc. 16th ISMV, Blasksburg, VA* (1986) 110-114.

[6] G. Mayor and T. Calvo. "On extended Aggregation Functions". *Próxima publicación en Proc. IFSA'97, Praga* (1997).

[7] E. Trillas, C. Alsina, S. Cubillo and J. Jacas. "On 'and' in [0,1]". *Proc. IPMU'92* (1992) 324-332.

[8] E. Trillas, S. Cubillo and J.L. Castro. "Conjunction and disjunction on ([0,1], ≤)". *Fuzzy Sets and Systems* 72 (1995) 155-165.

[9] R.R. Yager. "On Ordered Weighted Averaging Operators in Multicriteria Decisionmaking." *IEEE Transactions on Systems, Man and Cybernetics* 18 (1988) 183-190.

# MEDIAS CASI LINEALES PONDERADAS EXTENDIDAS

**T. Calvo**
Dept. Matemàtiques i Informàtica
Universitat de les Illes Balears
07071-Palma de Mallorca
dmitcs0@ps.uib.es

**P. Canet**
Dept. Matemàtiques i Informàtica
Universitat de les Illes Balears
07071-Palma de Mallorca
dmipce0@ps.uib.es

**M. Carbonell**
Dept. Matemàtiques i Informàtica
Universitat de les Illes Balears
07071-Palma de Mallorca
dmimch0@ps.uib.es

## Resumen

En este trabajo se introduce el concepto de media casi lineal ponderada extendida. A continuación, se caracterizan las medias que son funciones de agregación extendidas por medio de los triángulos descendentes. Además se estudia la propiedad de auto-identidad de estas medias quedando caracterizada por otro tipo de triángulos. Estos triángulos vienen generados por la sucesión $\{w_n^n\}$, $n \geq 2$ .

**Palabras clave.** Media casi lineal ponderada extendida, función de agregación extendida, triángulos probabilísticos o de pesos, auto-identidad.

## 1 INTRODUCCIÓN

En el problema de agregar información juegan un papel importante las medias casi lineales ponderadas extendidas, así como las funciones de agregación extendidas y los operadores OWA (Ordered Weighted Averaging operator), como puede verse en [2]. Estas funciones vienen caracterizadas por ciertas propiedades pero además pueden verificar otras, como la auto-identidad, cuyo estudio forma parte del objetivo de este trabajo y que determina otro tipo de triángulos. Estos triángulos nos permiten ver la relación que existe entre las medias y las funciones de agregación extendidas.
Por otra parte, se estudian las medias casi lineales ponderadas extendidas definidas por una sucesión de pesos $\{w_n\}$, $n \geq 1$, en vez de un triángulo de pesos $\bigwedge w_i^n$ . En este caso se ve que la propiedad de auto-identidad queda totalmente caracterizada.

## 2 PRELIMINARES

En esta sección presentamos las definiciones y resultados básicos que se irán utilizando a lo largo del trabajo.

Dado un retículo $(L, \leq)$ con máximo 0 y mínimo 1, denotaremos por $E$ al conjunto $E = \bigcup_{n\geq 1} L^n$ de todas las listas ordenadas de elementos de $L$.

**Definición 1** Las relaciones $\leq_\pi, \leq_\alpha$ y $\leq_\beta$ detalladas a continuación definen órdenes sobre el conjunto $E$. Dados $x = (x_1, \ldots, x_n)$ , $y = (y_1, \ldots, y_m)$ elementos de $E$:

- $x \leq_\pi y$ si y sólo si $n = m$ y $x_1 \leq y_1, \ldots, x_n \leq y_n$ (*orden producto*)
- $x \leq_\alpha y$ si y sólo si $n \leq m$, $x_1 \leq y_1, \ldots, x_n \leq y_n$, y si $n < m$ entonces $sup(x_1, \ldots, x_n) \leq inf(y_{n+1}, \ldots, y_m)$
- $x \leq_\beta y$ si y sólo si $n \geq m$, $x_1 \leq y_1, \ldots, x_n \leq y_n$, y si $n > m$ entonces $sup(x_{m+1}, \ldots, x_n) \leq inf(y_1, \ldots, y_m)$

El siguiente resultado establece una caracterización a menudo útil de la monotonía respecto de los órdenes $\alpha$ y $\beta$.

**Proposición 1** Sea $A : E \to L$ una aplicación monótona respecto al orden producto. Entonces, para todo $(x_1, \ldots, x_n) \in E$ se verifica:
a) $A$ es monótona respecto a $\leq_\alpha$ si y sólo si $A(x_1, \ldots, x_n) \leq A(x_1, \ldots, x_n, sup(x_1, \ldots, x_n))$ .
b) $A$ es monótona respecto a $\leq_\beta$ si y sólo si $A(x_1, \ldots, x_n, inf(x_1, \ldots, x_n)) \leq A(x_1, \ldots, x_n)$ .

**Definición 2** Una aplicación $A : E \to L$ es una *Función de Agregación Extendida* ($A$ es una FAE) si verifica las siguientes condiciones:

1) $A$ es monótona respecto a los órdenes $\leq_\alpha$ y $\leq_\beta$

2) $A$ es idempotente, es decir, $A(\overbrace{x,\ldots,x}^{n}) = x$ para todo $x \in L$ y $n \geq 1$.

**Definición 3** Una aplicación $A : E \to L$ verifica la propiedad de *auto-identidad* si:

$$A(x_1,\cdots,x_n,x_{n+1}) = A(x_1,\cdots,x_n),$$

$$\text{siempre que} \quad x_{n+1} = A(x_1,\cdots,x_n)$$

**Definición 4** Sea $N : L \to L$ una negación ($N$ involutiva y decreciente) y $A$ una aplicación $A : E \to L$ . Definimos la aplicación $A^* : E \to L$ (*N-dual de A*) por

$$A^*(x_1,\cdots,x_n) = N(A(N(x_1),\cdots,N(x_n)))$$

En [2] se demuestra el siguiente lema.

**Lema 1** Sea $N$ unanegación sobre $L$ y $A : E \to L$. Si $A$ es monótona respecto a $\leq_\alpha$ entonces $A^*$ es monótona respecto a $\leq_\beta$ , y si $A$ es monótona respecto a $\leq_\beta$ entonces $A^*$ es monótona respecto a $\leq_\alpha$ .

**Definición 5** (Yager, 1988) Una aplicación $A : [0,1]^n \to [0,1]$ es un operador OWA (*Ordered Weighted Averaging* operator), $n-$dimensional si existe $W = (w_1,\ldots,w_n) \in [0,1]^n$ con $\sum_{i=1}^{n} w_i = 1$ y tal que

$$A(x_1,\ldots,x_n) = W \cdot R(x_1,\ldots,x_n)$$

$$\text{para todo} \quad (x_1,\ldots,x_n) \in [0,1]^n$$

donde $R(x_1,\ldots,x_n) = (y_1,\ldots,y_n)$ no es más que la $n-$pla $(x_1,\ldots,x_n)$ reordenada de forma decreciente y $W \cdot R(x_1,\ldots,x_n) = \sum_{i=1}^{n} w_i y_i$ .

**Definición 6** Diremos que una aplicación $A : \bigcup_{n\geq 1} [0,1]^n \to [0,1]$ es un operador EOWA (un operador OWA extendido) si la restricción de $A$ a $[0,1]^n$ es un operador OWA $n-$dimensional. Es decir, si para cada $n$ existe un $W_n = (w_1^n,\ldots,w_n^n)$ tal que $A(x_1,\ldots,x_n) = W_n \cdot R(x_1,\ldots,x_n)$ para todo $(x_1,\ldots,x_n) \in [0,1]^n$.

**Nota 1** Observese que un operador EOWA es siempre idempotente, simétrico y monótono respecto del orden producto.

Los siguientes ejemplos muestran cómo algunos operadores clásicos son operadores EOWA a la vez que funciones de agregación extendidas.

**Ejemplo 1** $A(x_1,\ldots,x_n) = min(x_1,\ldots,x_n)$ es un operador EOWA con $W_n = (\overbrace{0,\ldots,0}^{n-1},1)$ para cada $n$.

Análogamente, $A(x_1,\ldots,x_n) = max(x_1,\ldots,x_n)$ es también un operador EOWA con $W_n = (1,\overbrace{0,\ldots,0}^{n-1})$ para cada $n$. Se ve fácilmente que ambos operadores son también FAE.

**Ejemplo 2** La media aritmética, $A(x_1,\ldots,x_n) = (x_1+\ldots+x_n)/n$ es un operador EOWA con $W_n = (\overbrace{1/n,\ldots,1/n}^{n})$ para cada $n$. Es fácil comprobar que $A$ és además FAE.

El siguiente resultado (ver [2]) da una caracterización de los operadores EOWA que son a la vez FAE.

**Proposición 2** Sea $A$ un operador EOWA con $W_n = (w_1^n,\ldots,w_n^n)$ para cada $n \geq 1$ . Entonces $A$ es una FAE si y sólo si las siguientes desigualdades se verifican para todo $n$ y $p = 1,\ldots,n$

$$\sum_{i=1}^{p} w_i^{n+1} \leq \sum_{i=1}^{p} w_i^n \leq \sum_{i=1}^{p+1} w_i^{n+1} .$$

## 3 MEDIAS CASI LINEALES PONDERADAS EXTENDIDAS

**Definición 7** Llamaremos *triángulo probabilístico o de pesos* a una colección de números $w_i^n \in [0,1]$, $i = 1,\ldots,n$, $n \geq 1$ que verifica: $\sum_{i=1}^{n} w_i^n = 1$, $n \geq 1$

$$\begin{array}{ccccc} & & 1 & & \\ & w_1^2 & & w_2^2 & \\ w_1^3 & & w_2^3 & & w_3^3 \\ w_1^4 & & w_2^4 & & w_3^4 \quad w_4^4 \\ & & \ldots & & \end{array}$$

Lo denotaremos por $\bigwedge w_i^n$

**Definición 8** Sea $f : [0,1] \to [-\infty,+\infty]$ una función real continua, estrictamente monótona tal que $\{f(0),f(1)\} \neq \{-\infty,+\infty\}$; y sea $\bigwedge w_i^n$ , un triángulo de pesos. Diremos que una función

$$A : \bigcup_{n\geq 1} [0,1]^n \to [0,1]$$

definida por

$$A(x_1,\ldots,x_n) = f^{-1}\left(\sum_{i=1}^{n} w_i^n f(x_i)\right)$$

es una *media casi lineal ponderada extendida.*

Estas medias verifican claramente las propiedades

- Idempotencia: $A(x,....x) = x$.
- Monotonía respecto del orden producto.

que son comunes a la mayoría de funciones de agregación.

**Nota 2** Estas medias pueden definirse también sobre cualquier intervalo real $[a,b]$ ; de manera que todos los resultados que aparecen a continuación también son válidos para este intervalo.

**Ejemplo 3** Si consideramos $f$ la función identidad $f(x) = x$ y $w_i^n = 1/n$, $i = 1,\ldots,n$, entonces la definición anterior coincide con la media aritmética de $x_1,\ldots,x_n$.

Las propiedades de idempotencia y monotonía imponen ciertas condiciones sobre estas medias.

**Proposición 3** Una media casi lineal ponderada extendida $A$ verifica:

$$Min(x_1,\ldots,x_n) \leq A(x_1,\ldots,x_n) \leq Max(x_1,\ldots,x_n)$$

**Demostración.** Si llamamos $m = Min(x_1,\ldots,x_n)$, $M = Max(x_1,\ldots,x_n)$ tenemos

$$m = A(m,\ldots,m) \leq A(x_1,\ldots,x_n) \leq$$

$$A(M,\ldots,M) = M.$$

■

Una caracterización de cuando las medias casi lineales extendidas son funciones de agregación extendidas viene dado por el siguiente resultado.

**Teorema 1** Una media casi lineal ponderada extendida $A$ es una FAE si y sólo si el triángulo de pesos asociado verifica

$$w_i^{n+1} \leq w_i^n, \qquad n \geq 1, \qquad i = 1,\ldots,n.$$

**Demostración.** Consideremos primeramente que la función $f$ es creciente.
Supongamos que $A$ es una FAE, entonces si $m = Min(x_1,\ldots,x_n)$, $M = Max(x_1,\ldots,x_n)$ tenemos

$$A(x_1,\ldots,x_n,m) \leq A(x_1,\ldots,x_n) \leq A(x_1,\ldots,x_n,M)$$

para todo $(x_1,\ldots,x_n)$, i.e.

$$f^{-1}\left(\sum_{j=1}^{n} w_j^{n+1} f(x_j) + w_{n+1}^{n+1} f(m)\right) \leq$$

$$f^{-1}\left(\sum_{j=1}^{n} w_j^n f(x_j)\right) \leq$$

$$f^{-1}\left(\sum_{j=1}^{n} w_j^{n+1} f(x_j) + w_{n+1}^{n+1} f(M)\right)$$

Si consideramos $x_i = 1$; $x_j = 0$, si $j \neq i$, tenemos:

$$\sum_{\substack{j=1\\ j\neq i}}^{n+1} w_j^{n+1} f(0) + w_i^{n+1} f(1) \leq$$

$$\sum_{\substack{j=1\\ j\neq i}}^{n} w_j^n f(0) + w_i^n f(1) \leq$$

$$\sum_{\substack{j=1\\ j\neq i}}^{n} w_j^{n+1} f(0) + (w_i^{n+1} + w_{n+1}^{n+1}) f(1). \qquad (1)$$

La primera de las dos desigualdades anteriores queda:

$$(1 - w_i^{n+1}) f(0) + w_i^{n+1} f(1) \leq (1 - w_i^n) f(0) + w_i^n f(1)$$

o bien

$$(w_i^n - w_i^{n+1})(f(0) - f(1)) \leq 0$$

y como f es creciente deducimos que $w_i^n - w_i^{n+1} \geq 0$ .

De la segunda de las dos desigualdades de (1) obtenemos:

$$(1 - w_i^n) f(0) + w_i^n f(1) \leq (1 - w_i^{n+1} - w_{n+1}^{n+1}) f(0)$$

$$+(w_i^{n+1} + w_{n+1}^{n+1}) f(1)$$

o bien

$$(w_i^{n+1} + w_{n+1}^{n+1} - w_i^n)(f(0) - f(1)) \leq 0$$

y como $f$ es creciente deducimos que $w_i^{n+1} + w_{n+1}^{n+1} - w_i^n \geq 0$ . Finalmente, esta última desigualdad es fácilmente deducible a partir de $w_i^n - w_i^{n+1} \geq 0$ .

Recíprocamente, como $m \leq x_j$ $j = 1,\ldots,n$ entonces $f(m) \leq f(x_j)$ de donde

$$(w_j^n - w_j^{n+1}) f(m) \leq (w_j^n - w_j^{n+1}) f(x_j)$$

$$f(m) \sum_{j=1}^{n} (w_j^n - w_j^{n+1}) \leq \sum_{j=1}^{n} (w_j^n - w_j^{n+1}) f(x_j)$$

$$f(m)(1 - \sum_{j=1}^{n} w_j^{n+1}) \leq \sum_{j=1}^{n} w_j^n f(x_j) - \sum_{j=1}^{n} w_j^{n+1} f(x_j)$$

$$w_{n+1}^{n+1} f(m) \leq \sum_{j=1}^{n} w_j^n f(x_j) - \sum_{j=1}^{n} w_j^{n+1} f(x_j)$$

reordenando y aplicando $f^{-1}$ a cada término de la desigualdad se obtiene

$$f^{-1}\left(\sum_{j=1}^{n} w_j^{n+1} f(x_j) + w_{n+1}^{n+1} f(m)\right) \leq$$

$$f^{-1}\left(\sum_{j=1}^{n} w_j^n f(x_j)\right)$$

que es equivalente a

$$A(x_1, \ldots, x_n, m) \leq A(x_1, \ldots, x_n).$$

La desigualdad

$$A(x_1, \ldots, x_n) \leq A(x_1, \ldots, x_n, M)$$

se obtiene de manera análoga considerando $x_j \leq M, \quad j = 1, \ldots, n.$

Por último, cuando $f$ es decreciente, la demostración es análoga a la que acabamos de realizar. ∎

A partir de ahora los triángulos de pesos que verifiquen $w_i^{n+1} \leq w_i^n$ les llamaremos *descendentes.*

El siguiente resultado determina cuando una media casi lineal ponderada extendida verifica la propiedad de auto-identidad.

**Teorema 2** Una media casi lineal ponderada extendida $A$ verifica la propiedad de auto-identidad si y sólo si el triángulo de pesos verifica:

$$w_i^{n+1} = w_i^n (1 - w_{n+1}^{n+1}), \quad n \geq 1, \quad i = 1, \ldots, n. \quad (2)$$

**Demostración.** Si $A$ verifica la propiedad de auto-identidad, i.e.

$$A(x_1, \ldots, x_n, A(x_1, \ldots, x_n)) = A(x_1, \ldots, x_n)$$

para todo $n \geq 1$ , según la definición de $A$ se obtiene

$$\sum_{i=1}^{n} w_i^{n+1} f(x_i) + w_{n+1}^{n+1}\left(\sum_{i=1}^{n} w_i^n f(x_i)\right) =$$

$$\sum_{i=1}^{n} w_i^n f(x_i) \quad (3)$$

Tomando en (3) $x_i = 1$ y $x_j = 0$ para todo $i = 1, \ldots, n$ y $j \neq i$ se sigue:

$$(w_i^{n+1} + w_{n+1}^{n+1} w_i^n - w_i^n) f(1) =$$

$$(\sum_{\substack{j=1 \\ j \neq i}}^{n} w_j^n - w_{n+1}^{n+1} \sum_{\substack{j=1 \\ j \neq i}}^{n} w_j^n - \sum_{\substack{j=1 \\ j \neq i}}^{n} w_j^{n+1}) f(0)$$

de donde

$$(w_i^{n+1} + w_{n+1}^{n+1} w_i^n - w_i^n) f(1) =$$

$$[(1 - w_{n+1}^{n+1}) \sum_{\substack{j=1 \\ j \neq i}}^{n} w_j^n - (1 - w_i^{n+1} - w_{n+1}^{n+1})] f(0)$$

o equivalentemente

$$(w_i^{n+1} + w_{n+1}^{n+1} w_i^n - w_i^n) f(1) =$$

$$[(1 - w_{n+1}^{n+1})(\sum_{\substack{j=1 \\ j \neq i}}^{n} w_j^n - 1) + w_i^{n+1}] f(0)$$

Esta última igualdad se expresa también de la forma

$$[w_i^{n+1} + w_i^n (w_{n+1}^{n+1} - 1)] f(1) =$$

$$[-(1 - w_{n+1}^{n+1}) w_i^n + w_i^{n+1}] f(0)$$

o bien, en la forma

$$[w_i^{n+1} + w_i^n (w_{n+1}^{n+1} - 1)](f(1) - f(0)) = 0 \,,$$

de donde, al ser $f(1) \neq f(0)$ , se obtiene (2).

Recíprocamente, teniendo en cuenta la condición (2), se deduce fácilmente

$$\sum_{i=1}^{n} (w_i^{n+1} + w_{n+1}^{n+1} w_i^n - w_i^n) f(x_i) = 0$$

de donde,

$$\sum_{i=1}^{n} w_i^{n+1} f(x_i) + \sum_{i=1}^{n} w_{n+1}^{n+1} w_i^n f(x_i) = \sum_{i=1}^{n} w_i^n f(x_i)$$

es decir,

$$A(x_1, \ldots, x_n, A(x_1, \ldots, x_n)) = A(x_1, \ldots, x_n)$$

∎

**Nota 3** Se comprueba fácilmente que todo triángulo de pesos que verifica la condición (2) es descendente.

**Nota 4** Los triángulos que verifican la propiedad (2) quedan determinados por los valores $w_n^n$ con $n \geq 2$. Luego, la forma genérica de estos triángulos es la siguiente:

$$
\begin{array}{ccccc}
 & & 1 & & \\
 & 1\text{-a} & & a & \\
(1\text{-a})(1\text{-b}) & & a(1\text{-b}) & & b \\
(1\text{-a})(1\text{-b})(1\text{-c}) & & a(1\text{-b})(1\text{-c}) & b(1\text{-c}) & c \\
 & & \ldots\ldots\ldots & &
\end{array}
$$

De los resultados anteriores podemos enunciar el siguiente corolario.

**Corolario 1** Si una media casi lineal ponderada extendida verifica la propiedad de auto-identidad entonces es una función de agregación extendida.

**Demostración.** De la Nota 3 se deduce que si una media casi lineal ponderada extendida $A$ verifica la propiedad de auto-identidad, el triángulo probabilístico asociado es descendente y por el Teorema 1 resulta que $A$ es una función de agregación extendida. ∎

Presentamos ahora un resultado relacionado con el recíproco de la propiedad de auto-identidad.

**Proposición 4** Sea $A$ una media casi lineal ponderada extendida con triángulo de pesos que verifica la condición (2), entonces se verifica que:

$$A(x_1,\ldots,x_{n+1}) = A(x_1,\ldots,x_n) \Longrightarrow$$

$$x_{n+1} = A(x_1,\ldots,x_n)$$

**Demostración.** De $A(x_1,\ldots,x_{n+1}) = A(x_1,\ldots,x_n)$ se obtiene

$$\sum_{i=1}^{n} w_i^{n+1} f(x_i) + w_{n+1}^{n+1} f(x_{n+1}) = \sum_{i=1}^{n} w_i^n f(x_i)$$

y aplicando (2) se sigue que

$$-\sum_{i=1}^{n} w_i^n w_{n+1}^{n+1} f(x_i) + w_{n+1}^{n+1} f(x_{n+1}) = 0$$

de donde fácilmente se tiene

$$f(x_{n+1}) = \sum_{i=1}^{n} w_i^n f(x_i)$$

luego

$$x_{n+1} = f^{-1}\left(\sum_{i=1}^{n} w_i^n f(x_i)\right) = A(x_1,\ldots,x_n)$$

∎

El siguiente resultado relaciona la propiedad de auto-identidad con la de dualidad.

**Proposición 5** Sea $N$ una negación y $A$ una media casi lineal ponderada extendida que verifica la propiedad de auto-identidad, entonces la aplicación $N$-dual de $A$ también satisface esta propiedad.

**Demostración.** La aplicación $N$-dual $A^*$ verifica la propiedad de auto-identidad si y sólo si

$$\sum_{i=1}^{n} w_i^{n+1} f(N(x_i)) +$$

$$w_{n+1}^{n+1} f N N f^{-1}\left(\sum_{i=1}^{n} w_i^n f(N(x_i))\right) = \sum_{i=1}^{n} w_i^n f(N(x_i)),$$

pero como $A$ verifica la propiedad de auto-identidad se tiene que

$$w_i^{n+1} + w_i^n w_{n+1}^{n+1} = w_i^n, \qquad i = 1,\ldots,n\ .$$

Y a partir de esta última igualdad se deduce la anterior. ∎

A continuación introducimos un caso particular de medias casi lineales ponderadas extendidas que vienen generadas por una sucesión de numeros reales positivos $w_1, w_2, \ldots$ (*pesos*) y una función $f : [0,1] \to [-\infty,\infty]$ continua, estrictamente monótona con $\{f(0), f(1)\} \neq \{-\infty,\infty\}$, de la siguiente forma:

$$A : \bigcup_{n\geq 1} [0,1]^n \to [0,1]$$

donde

$$A(x_1,\ldots,x_n) = f^{-1}\left(\frac{\sum_{i=1}^{n} w_i f(x_i)}{\sum_{i=1}^{n} w_i}\right)\ .$$

Observemos que en este caso, llamando $w_i^n = w_i / \sum_{i=1}^{n} w_i$ resulta que la función $A$ introducida es una media casi lineal ponderada extendida.

Este tipo de medias casi lineales ponderadas extendidas verifican siempre la propiedad de auto-identidad así como el recíproco, lo que se comprueba en el siguiente resultado.

**Teorema 3** Sea $A$ una media casi lineal ponderada extendida definida por

$$A(x_1,\ldots,x_n) = f^{-1}\left(\frac{\sum_{i=1}^{n} w_i f(x_i)}{\sum_{i=1}^{n} w_i}\right)\ .$$

Entonces :

$$A(x_1, \ldots, x_{n+1}) = A(x_1, \ldots, x_n) \Longleftrightarrow$$

$$x_{n+1} = A(x_1, \ldots, x_n)$$

**Demostración.** Es suficiente comprobar que los pesos $w_i^n = w_i / \sum_{i=1}^{n} w_i$ verifican la condición (2) ya que entonces el resultado se deduce del teorema 2 y la proposición 4. Veamos que en efecto satisfacen la condición (2):

$$w_i^n \left(1 - w_{n+1}^{n+1}\right) = \frac{w_i}{\sum_{i=1}^{n} w_i} \left(1 - \frac{w_{n+1}}{\sum_{i=1}^{n+1} w_i}\right) =$$

$$\frac{w_i}{\sum_{i=1}^{n} w_i} \left(\frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{n+1} w_i}\right) = \frac{w_i}{\sum_{i=1}^{n+1} w_i} = w_i^{n+1}.$$

■


### Agradecimientos

Este trabajo ha sido, en parte, financiado por la CICYT, proyecto: TIC96-1393-C06-06.


### Referencias


1. Carbonell, M., Mas, M., Mayor, G. "On a class of Monotonic Extended OWA Operators". FUZZ-IEEE'97. Barcelona

2. Mayor, G., Calvo, T. "On Extended Aggregation Functions". IFSA'97. Praga.

3. Yager, R.R."On Ordered weighted averaging operators in multicriteria decisionmaking". *IEEE Transactions on Systems, Man and Cybernetic*, 18(1): 183-190, 1988.

4. Yager, R.R."Connectives and Quantifiers in Fuzzy Sets". *Fuzzy Sets and Systems*, 40 (1991) 39-75.

5. Yager, R.R. "On mean type aggregation". Technical report, Iona College. New Rochelle, NY 10801, 1994.

6. Yager, R.R. "Constrained OWA aggregation". *Fuzzy Sets and Systems*, 810 (1996) 89-101.

# GENERACIÓN DE FUNCIONES DE AGREGACIÓN EXTENDIDAS

**M. Mas**
Dept. Matemàtiques i Informàtica
Universitat de les Illes Balears
07071-Palma de Mallorca
dmimmg0@ps.uib.es

**G. Mayor**
Dept. Matemàtiques i Informàtica
Universitat de les Illes Balears
07071-Palma de Mallorca
dmigmf0@ps.uib.es

**J. Suñer**
Dept. Matemàtiques i Informàtica
Universitat de les Illes Balears
07071-Palma de Mallorca
dmijsl0@ps.uib.es

**J. Torrens**
Dept. Matemàtiques i Informàtica
Universitat de les Illes Balears
07071-Palma de Mallorca
dmijts0@ps.uib.es

## Resumen

En este trabajo, vemos dos formas de generar funciones de agregación extendidas. Inicialmente obtenemos operadores OWA extendidos de dos formas diferentes, una a través de cuantificadores y otra a partir de sucesiones. En el primer caso vemos que todos los operadores obtenidos son a la vez funciones de agregación extendidas y caracterizamos las que provienen de cuantificadores. En el caso de sucesiones, caracterizamos las que dan lugar a funciones de agregación extendidas y presentamos algunos ejemplos y propiedades.

**Palabras Clave.** Operadores OWA, funciones de agregación extendidas, cuantificadores, triángulo de pesos.

## 1 INTRODUCCIÓN

En los últimos años el problema de la agregación de la información ha sido objeto de estudio desde diversos puntos de vista. Concretamente, operadores que han demostrado ser interesantes en el manejo de dicho problema son los llamados operadores OWA $n$–dimensionales introducidos por Yager en [5] y que han sido extensamente estudiados por diversos autores ([1], [2], [4], [5], [6], [7], [8]). En particular, en [4] se estudian los llamados operadores WOWA, una generalización a la vez de los operadores OWA y de las medias ponderadas.

Sin embargo, el común denominador de estos operadores es siempre su carácter $n$-dimensional. En cambio, en [1] se introducen los operadores OWA extendidos (EOWA) cuya principal aportación es su carácter multidimensional, es decir, que se aplican indistintamente a listas de $n$ elementos para cualquier valor de $n \geq 1$. Un paso más viene dado por la introducción en [3] de las funciones de agregación extendidas (FAE) que, además de ser operadores multidimensionales, aportan la novedad de ser operadores monótonos respecto de ciertos órdenes que permiten comparar listas de elementos de diferente longitud. La relación entre estos dos últimos tipos de operadores viene dada en [3] donde se caracteriza cuándo un operador EOWA es una FAE a través de ciertas relaciones entre los pesos (ver Proposición 2).

En este trabajo, presentamos dos formas de obtener operadores EOWA que son a la vez FAE. La primera, a partir de los llamados cuantificadores (llamados también cuantificadores borrosos o cuantificadores lingüísticos) representados a través de subconjuntos borrosos de la recta real, es decir, como aplicaciones $Q : [0,1] \to [0,1]$ monótonas (ver [6], [7]). En estas referencias se ve cómo dichos cuantificadores son usados para interpretar conceptos como "mayoría", "muchos", etc... En esta parte, comprobamos que la forma de definir operadores OWA a partir de cuantificadores borrosos dada en [7] puede generalizarse para definir operadores EOWA y que éstos resultan ser siempre funciones de agregación extendidas. Caracterizamos los operadores EOWA que, siendo FAE, provienen de un cuantificador borroso y estudiamos sus propiedades.

La segunda forma consiste en construir operadores EOWA a través de sucesiones de números reales. En este caso, no todos los operadores obtenidos son FAE, pero caracterizamos cómo son las sucesiones que sí dan lugar a FAE y vemos algunos ejemplos de las mismas, como son las sucesiones decrecientes, las progresiones aritméticas y geométricas, la sucesión de Fibonacci, etc...

## 2 PRELIMINARES

En esta sección presentamos las definiciones y resultados básicos que se irán utilizando a lo largo del trabajo.

Dado un retículo $(L, \leq)$ con máximo 0 y mínimo 1, denotaremos por $E$ al conjunto $E = \bigcup_{n\geq 1} L^n$ de todas las listas ordenadas de elementos de $L$.

**Definición 1** Las relaciones $\leq_\pi, \leq_\alpha$ y $\leq_\beta$ detalladas a continuación definen órdenes sobre el conjunto $E$. Dados $x = (x_1, \ldots, x_n)$ , $y = (y_1, \ldots, y_m)$ elementos de $E$:

- $x \leq_\pi y$ si y sólo si $n = m$ y $x_1 \leq y_1, \ldots, x_n \leq y_n$ (*orden producto*)
- $x \leq_\alpha y$ si y sólo si $n \leq m$, $x_1 \leq y_1, \ldots, x_n \leq y_n$, y si $n < m$ entonces $sup(x_1, \ldots, x_n) \leq inf(y_{n+1}, \ldots, y_m)$
- $x \leq_\beta y$ si y sólo si $n \geq m$, $x_1 \leq y_1, \ldots, x_n \leq y_n$, y si $n > m$ entonces $sup(x_{m+1}, \ldots, x_n) \leq inf(y_1, \ldots, y_m)$

El siguiente resultado establece una caracterización, a menudo útil, de la monotonía respecto de los órdenes $\alpha$ y $\beta$.

**Proposición 1** Sea $A : E \to L$ una aplicación monótona respecto del orden producto. Entonces se verifica:
a) $A$ es monótona respecto de $\leq_\alpha$ si y sólo si $A(x_1, \ldots, x_n) \leq A(x_1, \ldots, x_n, sup(x_1, \ldots, x_n))$.
b) $A$ es monótona respecto de $\leq_\beta$ si y sólo si $A(x_1, \ldots, x_n, inf(x_1, \ldots, x_n)) \leq A(x_1, \ldots, x_n)$
para todo $(x_1, \ldots, x_n) \in E$.

**Definición 2** Una aplicación $A : E \to L$ es una *Función de Agregación Extendida* ($A$ es una FAE) si verifica las siguientes condiciones:

1) $A$ es monótona respecto de los órdenes $\leq_\alpha$ y $\leq_\beta$

2) $A$ es idempotente, es decir, $A(\overbrace{x, \ldots, x}^{n}) = x$ para todo $x \in L$ y $n \geq 1$.

**Definición 3** (Yager) Una aplicación $A : [0,1]^n \to [0,1]$ es un operador OWA (*Ordered Weighted Averaging* operator) $n$–dimensional si existe $W = (w_1, \ldots, w_n) \in [0,1]^n$ (llamado vector de pesos) con $\sum_{i=1}^{n} w_i = 1$ y tal que

$$A(x_1, \ldots, x_n) = W \cdot R(x_1, \ldots, x_n)$$

para todo $(x_1, \ldots, x_n) \in [0,1]^n$ donde $R(x_1, \ldots, x_n) = (y_1, \ldots, y_n)$ es la propia lista de elementos $(x_1, \ldots, x_n)$ pero reordenada de forma decreciente y $W \cdot R(x_1, \ldots, x_n) = \sum_{i=1}^{n} w_i y_i$.

**Definición 4** Diremos que una aplicación $A : \bigcup_{n\geq 1} [0,1]^n \to [0,1]$ es un operador EOWA (un operador OWA extendido) si la restrición de $A$ a $[0,1]^n$ es un operador OWA $n$–dimensional. Es decir, si para cada $n \geq 1$ existe un vector $n$–dimensional de pesos, $W_n = (w_1^n, \ldots, w_n^n)$, tal que $A(x_1, \ldots, x_n) = W_n \cdot R(x_1, \ldots, x_n)$ para todo $(x_1, \ldots, x_n) \in [0,1]^n$.

**Nota 1** Obsérvese que un operador EOWA es siempre idempotente, simétrico y monótono respecto del orden producto. Notemos además que un operador EOWA puede ser representado a través de un *triángulo de pesos* o *probabilístico* de forma que cada fila $n$ de dicho triángulo está formada por el vector $n$–dimensional de pesos $W_n$ correspondiente. Es decir,

$$\begin{array}{c} 1 \\ w_1^2 \quad w_2^2 \\ w_1^3 \quad w_2^3 \quad w_3^3 \\ w_1^4 \quad w_2^4 \quad w_3^4 \quad w_4^4 \\ \ldots \end{array}$$

Recíprocamente, cada triángulo de pesos (es decir, tal que los elementos de cada fila suman 1) define de forma obvia un EOWA. De esta forma, representaremos un EOWA o un triángulo de pesos por $\wedge w_j^n$.

Los siguientes ejemplos muestran cómo algunos operadores clásicos son operadores EOWA a la vez que Funciones de Agregación Extendidas.

**Ejemplo 1** $A(x_1, \ldots, x_n) = min(\overbrace{x_1, \ldots, x_n}^{n-1})$ es un operador EOWA con $W_n = (\overbrace{0, \ldots, 0}^{}, 1)$ para cada $n$. Análogamente, $A(x_1, \ldots, x_n) = max(x_1, \ldots, x_n)$ es también un operador EOWA con $W_n = (1, \overbrace{0, \ldots, 0}^{n-1})$ para cada $n$. Se ve fácilmente que ambos operadores son también FAE.

**Ejemplo 2** La media aritmética, $A(x_1, \ldots, x_n) = \frac{x_1 + \ldots + x_n}{n}$ es un operador EOWA con $W_n = (\overbrace{1/n, \ldots, 1/n}^{n})$ para cada $n$. Es fácil comprobar que $A$ es además FAE.

El siguiente resultado (ver [3]) da una caracterización de los operadores EOWA que son a la vez FAE.

**Proposición 2** Sea $A$ un operador EOWA con $W_n = (w_1^n, \ldots, w_n^n)$ para cada $n \geq 1$ . Entonces $A$ es una FAE si y sólo si las siguientes desigualdades se verifican

para todo $n$ y $p = 1, \ldots, n$

$$\sum_{i=1}^{p} w_i^{n+1} \leq \sum_{i=1}^{p} w_i^n \leq \sum_{i=1}^{p+1} w_i^{n+1} . \qquad (1)$$

**Definición 5** Diremos que un triángulo de pesos $\wedge w_j^n$ es *regular* si satisface la condición (1).

**Definición 6** Dado $A$ un operador EOWA con $W_n = (w_1^n, \ldots, w_n^n)$ para cada $n \geq 1$ , definimos el *recíproco de* $A$ (representado por $A^r$) como el operador EOWA definido por $U_n = (u_1^n, \ldots, u_n^n)$ para cada $n \geq 1$ , donde $u_j^n = w_{n-j+1}^n$ para todo $n \geq 1$ y para todo $j = 1, \ldots, n$. Por otra parte, diremos que un operador EOWA $A$ es *simétrico* cuando $A = A^r$

# 3 GENERACIÓN POR CUANTIFICADORES

**Definición 7** Llamaremos *cuantificador* a una aplicación $Q : [0,1] \to [0,1]$ tal que

*i)* $Q(0) = 0,\ Q(1) = 1$

*ii)* $r < t \Longrightarrow Q(r) \leq Q(t)$

Esta definición puede hallarse ya en [6] y [7] y en ambos casos dichos cuantificadores se utilizan para la representación de conceptos como "mayoría", "muchos", etc... Concretamente, en [7] se ve cómo dado un cuantificador $Q : [0,1] \to [0,1]$, se puede construir un operador OWA $n-$dimensional, $A_Q$, con los pesos dados por

$$w_j = Q\left(\frac{j}{n}\right) - Q\left(\frac{j-1}{n}\right) \quad \text{para todo } j = 1, \ldots, n.$$

Entonces para evaluar por ejemplo, una proposición de la forma:

"La mayoría de los estudiantes son pobres"

se procedería de la siguiente forma: Se representa el concepto de mayoría por un cuantificador $Q$, el "ser pobre" a través de un conjunto borroso $B$ y entonces, dada una muestra de $n$ estudiantes $x_1, \ldots, x_n$, el valor de la proposición vendría dado por $A_Q(b_1, \ldots, b_n)$ donde $A_Q$ es el OWA $n-$dimensional obtenido a partir de $Q$ y cada $b_i = B(x_i)$.

Sin embargo, no hay razón ninguna para limitarse a operadores $n-$dimensionales. Este mismo sistema puede generalizarse a operadores multidimensionales y así, a partir de un cuantificador $Q$, podemos construir un operador EOWA $A_Q$ definiendo los pesos de la siguiente manera:

$$\forall n \geq 1,\ \omega_j^n = Q(\frac{j}{n}) - Q(\frac{j-1}{n}),\ j = 1, \ldots, n$$

y entonces $A_Q(x_1, \ldots, x_n) = \sum_{j=1}^{n} w_j^n y_j$, donde $(y_1, \ldots, y_n) = R_n(x_1, \ldots, x_n)$, siendo $R_n$ la reordenación decreciente sobre $[0,1]^n$.

Observemos que de esta forma obtenemos efectivamente un triángulo de pesos:

- $w_j^n \in [0,1],\ \forall j = 1, \ldots, n,\ \forall n \geq 1$
- $\sum_{j=1}^{n} w_j^n = \sum_{j=1}^{n} [Q(\frac{j}{n}) - Q(\frac{j-1}{n})] = Q(1) - Q(0) = 1$

Observemos también que
$\sum_{j=1}^{k} w_j^n = \sum_{j=1}^{k} [Q(\frac{j}{n}) - Q(\frac{j-1}{n})] = Q(\frac{k}{n})$.

**Ejemplo 3** A partir de los cuantificadores

$$Q_1(x) = \begin{cases} 0 & \text{si } x = 0 \\ 1 & \text{si } x \in (0,1] \end{cases}$$

$$Q_2(x) = \begin{cases} 1 & \text{si } x = 1 \\ 0 & \text{si } x \in [0,1) \end{cases}$$

$$Q_3(x) = \begin{cases} 0 & \text{si } 0 \leq x < 1/2 \\ 1/2 & \text{si } x = 1/2 \\ 1 & \text{si } 1/2 < x \leq 1 \end{cases}$$

se obtienen respectivamente los operadores EOWA

$$A_{Q_1} = Max, \quad A_{Q_2} = Min \quad \text{y} \quad A_{Q_3} = Mediana.$$

El triángulo correspondiente al operador $Max$ estaría formado por un 1 en la primera posición de cada fila y el resto ceros. El correspondiente al operador $Min$, igual pero el 1 de cada fila ocuparía la última plaza. Por último, el triángulo de pesos correspondiente a la $Mediana$ sería el siguiente:

$$\begin{array}{ccccccc} & & & 1 & & & \\ & & 1/2 & & 1/2 & & \\ & 0 & & 1 & & 0 & \\ 0 & & 1/2 & & 1/2 & & 0 \\ 0 & 0 & & 1 & & 0 & 0 \\ & & & \ldots & & & \end{array}$$

**Ejemplo 4** Si consideramos el cuantificador $Q(x) = x^2$, utilizado a menudo (ver [6],[7]) para representar el concepto borroso de "mayoría", se obtiene el operador EOWA con pesos $w_j^n = \frac{2j-1}{n^2}$ y entonces el triángulo correspondiente sería:

$$
\begin{array}{c}
1 \\
1/4 \quad 3/4 \\
1/9 \quad 3/9 \quad 5/9 \\
1/16 \quad 3/16 \quad 5/16 \quad 7/16 \\
1/25 \quad 3/25 \quad 5/25 \quad 7/25 \quad 9/25 \\
\ldots
\end{array}
$$

**Proposición 3** Sea $Q : [0,1] \to [0,1]$ un cuantificador. Entonces el operador EOWA $A_Q$ és una FAE.

**Dem.** Basta probar que el triángulo probabilístico generado por $Q$ es regular, es decir

$$\sum_{j=1}^{p} w_j^{n+1} \leq \sum_{j=1}^{p} w_j^n \leq \sum_{j=1}^{p+1} w_j^{n+1} \quad \forall\, p = 1, \ldots, n.$$

Pero esto equivale a probar que

$$Q(\frac{p}{n+1}) \leq Q(\frac{p}{n}) \leq Q(\frac{p+1}{n+1}) \quad \forall\, p = 1, \ldots, n$$

y esto es cierto ya que $Q$ es creciente, y $\frac{p}{n+1} \leq \frac{p}{n} \leq \frac{p+1}{n+1}$.

**Proposición 4** Sea $A : \bigcup_{n \geq 1} [0,1]^n \to [0,1]$ un operador EOWA con triángulo probabilístico $\wedge w_j^n$. Supongamos que $A$ es una FAE. Entonces existe un cuantificador $Q$ tal que $A = A_Q$ si y sólo si se cumple la condición siguiente:

$$k \cdot m = n \cdot r \Longrightarrow \sum_{j=1}^{k} w_j^n = \sum_{j=1}^{r} w_j^m. \qquad (2)$$

**Dem.** Supongamos primero que se cumple la condición (2). Sobre los racionales de $[0,1]$, definimos la aplicación $Q$ de la siguiente manera:

$Q(0) = 0$,

$Q(\frac{k}{n}) = \sum_{j=1}^{k} w_j^n \quad \forall n \geq 1 \quad \text{y} \quad \forall\, k = 1, \ldots, n$

Si ahora $\frac{k}{n} = \frac{r}{m}$, $n \neq m$, $0 \leq k \leq n$, $0 \leq r \leq m$, entonces

$$Q(\frac{k}{n}) = \sum_{j=1}^{k} w_j^n = \sum_{j=1}^{r} w_j^m = Q(\frac{r}{m}).$$

Por tanto $Q$ está bien definida sobre $\mathbb{Q} \cap [0,1]$ y consecuentemente la podemos extender a una aplicación $Q$ definida en todo el intervalo $[0,1]$.

Observemos que $A = A_Q$, ya que $Q(\frac{j}{n}) - Q(\frac{j-1}{n}) = \sum_{i=1}^{j} w_i^n - \sum_{i=1}^{j-1} w_i^n = w_j^n$.

Recíprocamente, supongamos ahora que $A = A_Q$, y vamos a probar la condición (2).

Tenemos: $\sum_{j=1}^{k} w_j^n = Q(\frac{k}{n})$, i $\sum_{j=1}^{r} w_j^m = Q(\frac{r}{m})$. Entonces si $k \cdot m = n \cdot r, Q(\frac{k}{n}) = Q(\frac{r}{m})$ por ser $Q$ aplicación.

**Nota 2** En la primera parte de la demostración, hemos visto que $Q$ queda definida sobre $\mathbb{Q} \cap [0,1]$. Veamos que si $A_Q = A_{Q'}$, entonces $Q = Q'$ sobre $\mathbb{Q} \cap [0,1]$.

Observemos que: $Q(\frac{j}{n}) - Q(\frac{j-1}{n}) = w_j^n = Q'(\frac{j}{n}) - Q'(\frac{j-1}{n})$.

Si ahora $j = 1$, resultará $Q(0) = Q'(0) = 1$, de donde se tiene $Q(\frac{1}{n}) = Q'(\frac{1}{n})$, y así sucesivamente hasta $Q(1) = Q'(1) = 1$.

**Nota 3** En el caso que $Q$ sea uniformemente continua sobre $\mathbb{Q} \cap [0,1]$, existe una única extensión de $Q$ a $[0,1]$ que también es uniformemente continua.

**Proposición 5** Sean $Q$ un cuantificador y $\wedge w_j^n$ el triángulo probabilístico generado por $Q$. Entonces $\wedge w_j^n$ es simétrico si y sólo si $Q$ es simétrico respecto a $(1/2, 1/2)$ sobre $\mathbb{Q} \cap [0,1]$ (es decir, $Q(1-x) = 1 - Q(x)$ para todo $x \in \mathbb{Q} \cap [0,1]$).

**Dem.** Supongamos primero que el triángulo probabilístico dado es simétrico, es decir:

$$w_j^n = w_{n-j+1}^n \quad \forall n \geq 2 \quad \text{y} \quad \forall j = 1, \ldots, n.$$

Sea $n \geq 2$. Sabemos que $w_j^n = Q(\frac{j}{n}) - Q(\frac{j-1}{n})$. Entonces

$$Q(\frac{j}{n}) - Q(\frac{j-1}{n}) = Q(\frac{n-j+1}{n}) - Q(\frac{n-j}{n}),$$

de donde

$$Q(\frac{j}{n}) + Q(1 - \frac{j}{n}) = Q(\frac{j-1}{n}) + Q(1 - \frac{j-1}{n})$$

para todo $j = 1, \ldots, n$.

Si ahora $j = n$, resulta:

$$Q(\frac{j}{n}) + Q(1 - \frac{j}{n}) = Q(1) + Q(0) = 1 \quad \forall j = 1, \ldots, n,$$

de donde

$$Q(1 - \frac{j}{n}) = 1 - Q(\frac{j}{n}) \quad \forall j = 1, \ldots, n,$$

es decir que $Q$ es simétrico respecto a $(1/2, 1/2)$ sobre $\mathbb{Q} \cap [0,1]$.

Recíprocamente, supongamos ahora que $Q$ es simétrico respecto a $(1/2, 1/2)$ sobre $\mathbb{Q} \cap [0,1]$, es decir

$$Q(1-x) = 1 - Q(x) \quad \forall x \in \mathbb{Q} \cap [0,1].$$

Sea $n \geq 2$. Entonces

$$\forall j = 1, \ldots, n, \quad Q(1 - \frac{j}{n}) = 1 - Q(\frac{j}{n}).$$

En particular

$$1 = Q(\frac{j}{n}) + Q(1 - \frac{j}{n}) = Q(\frac{j-1}{n}) + Q(1 - \frac{j-1}{n})$$

de donde

$$Q(\frac{j}{n}) - Q(\frac{j-1}{n}) = Q(\frac{n-j+1}{n}) - Q(\frac{n-j}{n}).$$

Con lo cual $w_j^n = w_{n-j+1}^n \quad \forall j = 1, \ldots, n$, y por tanto el triángulo es simétrico.

# 4 GENERACIÓN POR SUCESIONES

Consideremos una sucesión de números reales $\lambda_1, \lambda_2, \ldots \geq 0$ tal que $\lambda_1 > 0$. A partir de esta sucesión podemos generar un triángulo probabilístico de la forma siguiente:

$$\forall n \geq 1, \quad w_j^n = \frac{\lambda_j}{\lambda_1 + \cdots + \lambda_n}$$

Observemos que $\sum_{j=1}^{n} w_j^n = \frac{\sum_{j=1}^{n} \lambda_j}{\lambda_1 + \cdots + \lambda_n} = 1$, y por tanto obtenemos efectivamente un triángulo probabilístico.

**Proposición 6** Todo triángulo generado por una sucesión $\{\lambda_n\}$ es regular si y sólo si la sucesión $\left\{\frac{S_{n+1}}{S_n}\right\}$ es decreciente, donde $S_n = \lambda_1 + \cdots + \lambda_n$.

**Dem.** Supongamos primero que el triángulo es regular. Entonces

$$\sum_{j=1}^{p} w_j^{n+1} \leq \sum_{j=1}^{p} w_j^n \leq \sum_{j=1}^{p+1} w_j^{n+1} \quad \forall p = 1, \ldots, n.$$

En nuestro caso, $w_j^n = \frac{\lambda_j}{S_n}$, por lo que la condición anterior será

$$\sum_{j=1}^{p} \frac{\lambda_j}{S_{n+1}} \leq \sum_{j=1}^{p} \frac{\lambda_j}{S_n} \leq \sum_{j=1}^{p+1} \frac{\lambda_j}{S_{n+1}} \quad \forall p = 1, \ldots, n.$$

Si en la segunda desigualdad tomamos $p = n-1$ resulta

$$\frac{S_{n-1}}{S_n} \leq \frac{S_n}{S_{n+1}},$$

lo cual demuestra que la sucesión $\left\{\frac{S_{n+1}}{S_n}\right\}$ es decreciente.

Recíprocamente, supongamos ahora que $\left\{\frac{S_{n+1}}{S_n}\right\}$ es decreciente, y queremos probar que

$$\sum_{j=1}^{p} \frac{\lambda_j}{S_{n+1}} \leq \sum_{j=1}^{p} \frac{\lambda_j}{S_n} \leq \sum_{j=1}^{p+1} \frac{\lambda_j}{S_{n+1}} \quad \forall p = 1, \ldots, n,$$

o equivalentemente

$$\frac{S_p}{S_{n+1}} \leq \frac{S_p}{S_n} \leq \frac{S_{p+1}}{S_{n+1}}.$$

La primera desigualdad es evidente, ya que $S_n$ es creciente, y la segunda también por ser $\left\{\frac{S_{n+1}}{S_n}\right\}$ decreciente.

**Nota 4** Observemos que si un triángulo de pesos es regular, el operador EOWA correspondiente es una FAE.

**Ejemplo 5** Son regulares los triángulos generados por:

a) sucesiones decrecientes,

b) progresiones aritméticas,

c) progresiones geométricas.

**Ejemplo 6**
La sucesión de Fibonacci: $1, 1, 2, 3, 5, 8, \ldots$ genera el triángulo

$$\begin{array}{ccccc} & & 1 & & \\ & 1/2 & & 1/2 & \\ 1/4 & & 1/4 & & 2/4 \\ 1/7 & & 1/7 & 2/7 & 3/7 \\ 1/12 & 1/12 & 2/12 & 3/12 & 5/12 \\ & & \ldots & & \end{array}$$

Se demuestra fácilmente por inducción que dicha sucesión verifica:

a) $S_n = \lambda_{n+2} - 1$,

b) $(\lambda_{n+3} - 1)(\lambda_{n+1} - 1) \leq (\lambda_{n+2} - 1)^2$

y de esta última desigualdad se deduce que la sucesión $\left\{\frac{S_{n+1}}{S_n}\right\}$ es decreciente. Según la proposición (6), esto prueba que el triángulo generado por la sucesión de Fibonacci es regular.

En este trabajo se ha visto que se pueden obtener FAE a partir de cuantificadores y también de sucesiones. De

la proposición (4) también se deduce que existen FAE generadas a la vez por cuantificadores y por sucesiones. Veamos finalmente ejemplos que ilustran estos hechos.

**Ejemplo 7** Media aritmética.

Consideremos $Q(x) = x$. Entonces un sencillo cálculo prueba que $w_j^n = 1/n \quad \forall j = 1, \ldots, n, \quad \forall n \geq 1$.

Observemos que tomando la sucesión $\lambda_n = 1 \quad \forall n \geq 1$ se obtiene la misma FAE.

**Ejemplo 8** Mediana.

Según se ha visto en el ejemplo (3), esta FAE se obtiene a partir de un cuantificador. En cambio, como $w_1^3 = 0$, no puede venir generada por una sucesión.

**Ejemplo 9** Consideremos la sucesión $\lambda_n = 2^{n-1} \quad \forall n \geq 1$. Se ve fácilmente que el triángulo de pesos que genera dicha sucesión no proviene de un cuantificador.

**Agradecimientos**

Este trabajo ha sido, en parte, financiado por la CICYT, proyecto: TIC96-1393-C06-06.

**Referencias**

1. Carbonell, M., Mas, M., Mayor,G. "On a class of Monotonic Extended OWA Operators". FUZZ-IEEE'97. Barcelona.
2. Fodor, J., Marichal, J.L., Roubens, M:, "Characterization of the Ordered Weighted Averaging Operators", *IEEE Trans. on Fuzzy Systems*, 3:2 (1995) 236-240.
3. Mayor, G., Calvo, T. "On Extended Aggregation Functions". IFSA'97. Praga.
4. Torra, V., "The Weighted OWA operator", *Int. J. of Intel. Systems*, 12 (1997) 153-166.
5. Yager, R.R. "On ordered weighted averaging aggregation operators in multi-criteria decision making", *IEEE Trans. on SMC*, 18 (1988) 183-190.
6. Yager, R.R. "Connectives and Quantifiers in Fuzzy Sets". *Fuzzy Sets and Systems*, 40 (1991) 39-75.
7. Yager, R.R. "Quantifier Guided Aggregation Using OWA Operators" *Int. J. of Intel. Systems*, 11 (1996) 49-73.
8. Yager, R.R. "Constrained OWA aggregation". *Fuzzy Sets and Systems*,810 (1996) 89-101.

# TOMA DE DECISIONES COLECTIVAS CON PREFERENCIAS DIFUSAS RECÍPROCAS

**José Luis García Lapresta**
Dep. de Economía Aplicada (Matemáticas)
Facultad de CC EE y EE
Universidad de Valladolid
e-mail: lapresta@cpd.uva.es

**Bonifacio Llamazares Rodríguez**
Dep. de Economía Aplicada (Matemáticas)
Facultad de CC EE y EE
Universidad de Valladolid
e-mail: boni@eco.uva.es

## Resumen

En este trabajo se analizan algunos procedimientos de toma de decisiones colectivas, cuando los agentes muestran sus preferencias no necesariamente taxativas mediante relaciones binarias difusas recíprocas. En particular, caracterizamos las reglas de agregación de preferencias difusas recíprocas que son neutrales (aquéllas que dan un tratamiento igualitario a las opciones). Y de éstas, las que son descomponibles y verifican otras propiedades adicionales, como anonimato, unanimidad, aditividad, monotonía y reciprocidad.

**Palabras Clave:** preferencias difusas, reciprocidad, elección social, regla de agregación, anonimato, neutralidad, independencia de alternativas irrelevantes, unanimidad, monotonía.

## 1 Introducción

En los procedimientos convencionales de toma de decisiones colectivas se supone que los agentes manifiestan sus preferencias sobre los pares de opciones de forma binaria: tan solo señalan qué opción prefieren o bien se muestran indiferentes. Supongamos un agente que dispone de una unidad de preferencia para repartir entre cada par de opciones a comparar. En el caso clásico, donde las preferencias se muestran taxativamente, pueden darse dos casos. Si el agente es indiferente entre dos opciones, entonces reparte equitativamente la unidad de preferencia entre las mencionadas opciones, 0,5 en ambos casos. Si prefiere una a otra, entonces mostrará intensidad máxima de preferencia, 1, al comparar la mejor con la peor; a su vez, mostrará intensidad mínima, 0, al comparar la peor con la mejor.

En este trabajo partimos de una información más rica y realista en la que los agentes no solo informan de qué opciones prefieren, sino que evalúan numéricamente la intensidad de sus preferencias. Así, la unidad de preferencia podrá repartirse de cualquiera de las formas posibles, mediante la utilización de relaciones binarias difusas recíprocas.

El trabajo se organiza como sigue. Después de considerar en 1.1 la notación y los conceptos básicos, en 1.2 introducimos las reglas de agregación de preferencias difusas recíprocas y presentamos propiedades que reflejan diferentes aspectos normativos habitualmente tratados en la literatura de la elección social. Es el caso de las propiedades de respeto al anonimato, neutralidad, independencia de alternativas irrelevantes, respeto a la unanimidad y monotonía.

Con estas premisas en 2 estudiamos la agregación de preferencias difusas recíprocas, cuando se verifica el axioma de neutralidad. En 2.1 probamos que las reglas de agregación neutrales quedan definidas por una clase de funciones auxiliares que asignan, a cada vector de intensidades individuales de preferencia, el nivel correspondiente a la intensidad colectiva. En 2.2 introducimos las propiedades de aditividad y descomponibilidad para reglas de agregación neutrales y justificamos ciertos vínculos que existen entre estas propiedades y las ya mencionadas.

En 3 caracterizamos las reglas de agregación neutrales que son descomponibles y respetan el anonimato. Se trata de aquéllas que asocian como intensidad colectiva de preferencia la media aritmética de las imágenes, por una determinada aplicación, de las intensidades individuales de preferencia. Además, caracterizamos aquéllas que verifican otras propiedades adicionales, como respeto a la unanimidad, aditividad, monotonía y reciprocidad. La regla de la media aritmética, además de constituir un procedimiento sencillo y natural de agregación, es el único que cumple todas las propiedades consideradas.

### 1.1 Preferencias difusas recíprocas

**Definición.** Dado un conjunto ordinario $X$, una *relación binaria difusa sobre* $X$ es un subconjunto difuso de $X \times X$. Si $R$ es una relación binaria difusa sobre $X$ con función de pertenencia

$$\mu_R : X \times X \longrightarrow [0,1],$$

se entenderá que $\mu_R(x,y)$ es la *intensidad* con la que $x$ está relacionado con $y$. El conjunto de relaciones binarias difusas sobre $X$ vendrá denotado por $\mathcal{R}(X)$.

**Notación.** A partir de ahora consideraremos un conjunto finito de opciones, $X = \{x_1, \ldots, x_n\}$, sobre el que un agente, individual o colectivo, muestra de forma no necesariamente taxativa sus preferencias, a través de una relación binaria difusa $R$ sobre $X$. Con $r_{ij} = \mu_R(x_i, x_j)$ indicaremos el nivel de intensidad con el que el agente prefiere la opción $x_i$ a la opción $x_j$.

**Definición.** Una relación binaria difusa $R$ sobre $X = \{x_1, \ldots, x_n\}$ es *recíproca* si y sólo si

$$r_{ij} + r_{ji} = 1 \quad \forall i, j \in \{1, \ldots n\}.$$

El conjunto de relaciones binarias difusas recíprocas sobre $X$ vendrá denotado por $\mathcal{R}_r(X)$.

En adelante supondremos que toda relación binaria difusa que represente las preferencias de un agente es recíproca. Hemos de señalar que ninguna relación binaria difusa recíproca es ordinaria, ya que al ser $r_{ii} = \frac{1}{2}$ para cualquier $i \in \{1, \ldots, n\}$, se tiene $\frac{1}{2} \in \mu_R(X \times X)$, por lo que no se verifica $\mu_R(X \times X) \subseteq \{0, 1\}$.

### 1.2 Reglas de agregación

**Definición.** Sea $X = \{x_1, \ldots, x_n\}$ un conjunto de opciones ($n \geq 2$), sobre el que $m$ agentes ($m \geq 2$) muestran sus preferencias, a través de relaciones binarias difusas recíprocas.

Una *regla de agregación* es una aplicación

$$F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$$

que asigna a cada *configuración* de relaciones binarias difusas recíprocas, $(R^1, \ldots, R^m) \in \mathcal{R}_r(X)^m$, una relación binaria difusa $\overline{R} = F(R^1, \ldots, R^m)$, llamada *agregada*.

Dadas dos opciones $x_i, x_j \in X$, $r_{ij}^k = \mu_{R^k}(x_i, x_j)$ es el nivel de intensidad con el que el agente $k$ prefiere $x_i$ a $x_j$; con $\overline{r}_{ij} = \mu_{\overline{R}}(x_i, x_j)$ denotaremos el nivel de intensidad con el que $x_i$ es preferida colectivamente a $x_j$, mediante la relación agregada $\overline{R}$.

**Propiedades.** En la literatura sobre agregación de preferencias difusas existen diversas propiedades, a veces divergentes, para representar los principios normativos que han de satisfacer las reglas de agregación que pretendan respetar los valores individuales. En este sentido introducimos a continuación varias propiedades que puede tener una regla de agregación de preferencias difusas.

Sea $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ una regla de agregación.

1. $F$ *respeta el anonimato* si y sólo si para cualquier biyección $\sigma : \{1, \ldots, m\} \longrightarrow \{1, \ldots, m\}$ se verifica

   $$F\left(R^{\sigma(1)}, \ldots, R^{\sigma(m)}\right) = F(R^1, \ldots, R^m)$$

   para cualquier $(R^1, \ldots, R^m) \in \mathcal{R}_r(X)^m$.

   Parece lógico imponer esta propiedad a toda regla de agregación que intervenga en sistemas democráticos de votación, pues de lo contrario no sería igualitaria y los agentes tendrían diferente poder de influencia en el resultado final. Sin embargo, esta propiedad no es indispensable en la agregación de preferencias correspondientes a criterios, ya que éstos pueden tener diferentes pesos en la decisión multicriterio.

2. $F$ es *neutral* si y sólo si para cualesquiera $(R^1, \ldots, R^m)$, $(S^1, \ldots, S^m) \in \mathcal{R}_r(X)^m$ y $i, j, p, q \in \{1, \ldots, n\}$, se verifica

   $$\forall k \in \{1, \ldots, m\} \; r_{ij}^k = s_{pq}^k \quad \Rightarrow \quad \overline{r}_{ij} = \overline{s}_{pq}.$$

   Mientras que el respeto al anonimato expresa un tratamiento igualitario entre los agentes, la neutralidad lo representa para las opciones.

3. $F$ satisface el axioma de *independencia de alternativas irrelevantes* si y sólo si para cualesquiera $(R^1, \ldots, R^m)$, $(S^1, \ldots, S^m) \in \mathcal{R}_r(X)^m$ y $i, j \in \{1, \ldots, n\}$, se verifica

   $$\forall k \in \{1, \ldots, m\} \; r_{ij}^k = s_{ij}^k \quad \Rightarrow \quad \overline{r}_{ij} = \overline{s}_{ij}.$$

   Este axioma garantiza que la preferencia colectiva entre dos opciones sólo depende de las manifestaciones de los agentes sobre esas opciones y no de otras, consideradas irrelevantes.

4. $F$ *respeta la unanimidad* si y sólo si para cualesquiera $(R^1, \ldots, R^m) \in \mathcal{R}_r(X)^m$, $i, j \in \{1, \ldots, n\}$ y $c \in [0, 1]$, se verifica

   $$\forall k \in \{1, \ldots, m\} \; r_{ij}^k = c \quad \Rightarrow \quad \overline{r}_{ij} = c.$$

   Esta propiedad generaliza el principio débil de Pareto, que pasa a ser un caso particular del respeto a la unanimidad cuando las preferencias de los agentes son ordinarias y $c = 1$.

El respeto a la unanimidad puede parecer inoperante con la formulación que aquí establecemos para preferencias difusas, ya que sólo es aplicable cuando todos los agentes coinciden en el mismo nivel de intensidad. Sin embargo, tal como probaremos en la Prop. 1, resulta totalmente plausible cuando se da conjuntamente a la neutralidad y la monotonía.

5. $F$ es *monótona* si y sólo si para cualesquiera $(R^1,\ldots,R^m)$, $(S^1,\ldots,S^m) \in \mathcal{R}_r(X)^m$ y $i,j \in \{1,\ldots,n\}$, se verifica

$$\forall k \in \{1,\ldots,m\}\ r_{ij}^k \geq s_{ij}^k \quad \Rightarrow \quad \overline{r}_{ij} \geq \overline{s}_{ij}.$$

Esta propiedad refleja un principio incuestionable: si algún agente incrementa la intensidad de preferencia entre dos opciones, no puede disminuir la intensidad de la preferencia agregada entre dichas opciones.

6. $F$ es *recíproca* si y sólo si $\overline{R}$ es recíproca para cualquier $(R^1,\ldots,R^m) \in \mathcal{R}_r(X)^m$.

Obsérvese que toda regla de agregación neutral o monótona satisface el axioma de independencia de alternativas irrelevantes.

## 2 Neutralidad

La neutralidad garantiza un trato igualitario entre las opciones. En el Teorema 1 justificamos que la neutralidad equivale a la existencia de una función que asigne la intensidad colectiva de preferencia entre dos opciones a partir de las intensidades individuales entre estas opciones, independientemente de cuáles sean éstas. Este hecho permite analizar las reglas de agregación neutrales a través de funciones auxiliares de $[0,1]^m$ en $[0,1]$. Recíprocamente, a partir de una función de $[0,1]^m$ en $[0,1]$, puede definirse una regla de agregación neutral. En adelante, el estudio de las reglas de agregación neutrales quedará determinado por el análisis de las funciones auxiliares asociadas.

**Notación.** Dados $\bar{x}$, $\bar{y} \in [0,1]^m$, se entenderá que $\bar{x} = (x_1,\ldots,x_m)$, $\bar{y} = (y_1,\ldots,y_m)$; con $\bar{x} \geq \bar{y}$ se abreviará $x_i \geq y_i \ \ \forall i \in \{1,\ldots,m\}$. Si $c \in [0,1]$, con $\bar{c}$ se denotará el vector $(c,\ldots,c) \in [0,1]^m$.

### 2.1 Funciones auxiliares

**Teorema 1**

Si $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ es una regla de agregación, entonces son equivalentes:

1. $F$ es neutral.
2. Existe una única función $f : [0,1]^m \longrightarrow [0,1]$ tal que $\overline{r}_{ij} = f(r_{ij}^1,\ldots,r_{ij}^m)$ para cualesquiera $i,j \in \{1,\ldots,n\}$.

DEMOSTRACIÓN:

$1 \Rightarrow 2$: Por ser $F$ neutral, verifica el axioma de independencia de alternativas irrelevantes. Por tanto, dados $i,j \in \{1,\ldots,n\}$ con $i \neq j$, existe una función $f_{ij} : [0,1]^m \longrightarrow [0,1]$ que permite obtener la intensidad colectiva de preferencia entre las opciones $x_i$ y $x_j$ a partir del vector de intensidades individuales entre dichas opciones: $\overline{r}_{ij} = f_{ij}(r_{ij}^1,\ldots,r_{ij}^m)$, para cualquier configuración $(R^1,\ldots,R^m) \in \mathcal{R}_r(X)^m$. Veamos ahora que si $p,q \in \{1,\ldots,n\}$ son distintos entre sí, se tiene $f_{pq} = f_{ij}$. Dado $\bar{x} \in [0,1]^m$, existen $(R^1,\ldots,R^m)$, $(S^1,\ldots,S^m) \in \mathcal{R}_r(X)^m$, tales que $x_k = r_{ij}^k = s_{pq}^k$ para $k = 1,\ldots,m$. Por ser $F$ neutral se tiene

$$f_{ij}(\bar{x}) = f_{ij}(r_{ij}^1,\ldots,r_{ij}^m) = \overline{r}_{ij} = \\ = \overline{s}_{pq} = f_{pq}(s_{pq}^1,\ldots,s_{pq}^m) = f_{pq}(\bar{x}).$$

En consecuencia, existe $f : [0,1]^m \longrightarrow [0,1]$, tal que para cualesquiera $(R^1,\ldots,R^m) \in \mathcal{R}_r(X)^m$ y $i,j \in \{1,\ldots,n\}$, se tiene $\overline{r}_{ij} = f(r_{ij}^1,\ldots,r_{ij}^m)$, si $j \neq i$. Veamos a continuación que $\overline{r}_{ii} = f(\frac{1}{2},\ldots,\frac{1}{2})$. Por ser recíprocas las relaciones de preferencia individuales, se tiene $r_{ii}^k = \frac{1}{2}$ para todo $k \in \{1,\ldots,m\}$. Dados $p,q \in \{1,\ldots,n\}$ con $p \neq q$, sea $(S^1,\ldots,S^m) \in \mathcal{R}_r(X)^m$, tal que $s_{pq}^k = \frac{1}{2}$ para todo $k \in \{1,\ldots,m\}$. Entonces, por ser $F$ neutral, tenemos $\overline{r}_{ii} = \overline{s}_{pq} = f(\frac{1}{2},\ldots,\frac{1}{2})$. La unicidad de $f$ se desprende de la propia construcción.

$2 \Rightarrow 1$: Sean $(R^1,\ldots,R^m),(S^1,\ldots,S^m) \in \mathcal{R}_r(X)^m$ y $i,j,p,q \in \{1,\ldots,n\}$, tales que $r_{ij}^k = s_{pq}^k$ para todo $k \in \{1,\ldots,m\}$. Entonces se tiene

$$\overline{r}_{ij} = f(r_{ij}^1,\ldots,r_{ij}^m) = f(s_{pq}^1,\ldots,s_{pq}^m) = \overline{s}_{pq}. \qquad \blacksquare$$

**Notación.** De acuerdo con el teorema, diremos que $F$ es la regla de agregación neutral *asociada a* $f$ y que $f$ es la *función auxiliar asociada* a $F$.

**Observación.** A lo largo del trabajo sólo consideraremos reglas de agregación neutrales. Para este tipo de reglas, las diferentes propiedades quedan reflejadas de la siguiente forma sobre una función auxiliar arbitraria $f : [0,1]^m \longrightarrow [0,1]$:

1. Respeto al anonimato: cualquiera que sea la aplicación biyectiva $\sigma : \{1,\ldots,m\} \longrightarrow \{1,\ldots,m\}$ se verifica

$$\forall \bar{x} \in [0,1]^m \quad f(x_{\sigma(1)},\ldots,x_{\sigma(m)}) = f(\bar{x}).$$

2. Respeto a la unanimidad: $\forall c \in [0,1]\ \ f(\bar{c}) = c$.
3. Monotonía:

$$\forall \bar{x},\bar{y} \in [0,1]^m \quad \bar{x} \geq \bar{y} \ \Rightarrow\ f(\bar{x}) \geq f(\bar{y}).$$

4. Reciprocidad: $\forall \bar{x} \in [0,1]^m\ \ f(\bar{1} - \bar{x}) = 1 - f(\bar{x})$.

La propiedad de respeto a la unanimidad determina una intensidad colectiva de preferencia igual a la de las intensidades individuales, pero sólo cuando todos los agentes tienen la misma intensidad. En la Prop. 1 demostramos que, en las reglas monótonas, la propiedad de respeto a la unanimidad informa de cómo ha de ser la intensidad colectiva de preferencia, cuando no se exige la misma intensidad de preferencia a todos los agentes.

**Proposición 1**

Sea $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$. Si $F$ es monótona, son equivalentes:

1. $F$ respeta la unanimidad.
2. $\min\{x_1,\ldots,x_m\} \leq f(\bar{x}) \leq \max\{x_1,\ldots,x_m\}$ para cualquier $\bar{x} \in [0,1]^m$.

DEMOSTRACIÓN:

$1 \Rightarrow 2$: Dado $\bar{x} = (x_1,\ldots,x_m) \in [0,1]^m$, sean $a = \min\{x_1,\ldots,x_m\}$ y $b = \max\{x_1,\ldots,x_m\}$. De $\bar{a} \leq \bar{x} \leq \bar{b}$ se sigue, por ser $F$ monótona y respetar la unanimidad, $a = f(\bar{a}) \leq f(\bar{x}) \leq f(\bar{b}) = b$.

$\Leftarrow$) Por hipótesis, se tiene $c \leq f(\bar{c}) \leq c$ para todo $c \in [0,1]$. ∎

## 2.2 Aditividad y descomponibilidad

A continuación introducimos dos propiedades para las reglas de agregación neutrales: la aditividad y la descomponibilidad y mostramos ciertas relaciones existentes entre estas propiedades y otras que son casos particulares del respeto a la unanimidad.

**Definición.** Sea $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$.

1. $F$ es *aditiva* si y sólo si se verifica

$$f(\bar{x}+\bar{y}) = f(\bar{x}) + f(\bar{y})$$

para cualesquiera $\bar{x},\bar{y} \in [0,1]^m$ que satisfagan $\bar{x}+\bar{y} \in [0,1]^m$.

2. $F$ es *descomponible* si y sólo si para cualquier $\bar{x} \in [0,1]^m$ se verifica

$$f(\bar{x}) = f(x_1,0,\ldots,0) + \cdots + f(0,\ldots,0,x_m).$$

**Proposición 2**

Sea $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$.

1. $F$ aditiva $\Rightarrow$ $F$ descomponible

   $F$ descomponible $\not\Rightarrow$ $F$ aditiva.

2. $F$ aditiva $\Rightarrow$ $F$ monótona

   $F$ monótona $\not\Rightarrow$ $F$ aditiva.

3. $F$ descomponible $\not\Rightarrow$ $F$ monótona

   $F$ monótona $\not\Rightarrow$ $F$ descomponible.

DEMOSTRACIÓN: La justificación de las implicaciones verdaderas es puramente rutinaria. En cuanto a las implicaciones falsas, damos a continuación los correspondientes contraejemplos.

La regla de agregación neutral asociada a

$$f(\bar{x}) = \frac{1}{m}\sum_{i=1}^{m} x_i(1-x_i)$$

es descomponible; sin embargo, no es monótona y, por tanto, tampoco aditiva.

La regla de agregación neutral asociada a

$$f(\bar{x}) = x_1 \cdot \cdots \cdot x_m$$

es monótona, pero no descomponible y, por tanto, tampoco aditiva. ∎

Las propiedades $f(\bar{0}) = 0$ y $f(\bar{1}) = 1$ corresponden a debilitaciones del respeto a la unanimidad para reglas neutrales, relativas a los casos extremos. Así, $f(\bar{0}) = 0$ equivale a que la intensidad colectiva de preferencia sea nula cuando sean nulas las intensidades individuales. Y $f(\bar{1}) = 1$ equivale a que la intensidad colectiva de preferencia sea máxima cuando lo sean las intensidades individuales. En la siguiente proposición mostramos las relaciones existentes entre estas dos propiedades y las de descomponibilidad, reciprocidad y aditividad.

**Proposición 3**

Sea $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$.

1. Si $F$ es descomponible, entonces $f(\bar{0}) = 0$.
2. Si $F$ es recíproca, entonces

$$f(\bar{0}) = 0 \Leftrightarrow f(\bar{1}) = 1.$$

3. Si $F$ es aditiva, entonces

$$F \text{ es recíproca} \Leftrightarrow f(\bar{1}) = 1.$$

DEMOSTRACIÓN:

1. Al considerar la descomponibilidad para $\bar{x} = \bar{0}$, se tiene $f(\bar{0}) = mf(\bar{0})$, luego $f(\bar{0}) = 0$.
2. $\Rightarrow$) $f(\bar{1}) = f(\bar{1}-\bar{0}) = 1 - f(\bar{0}) = 1$.
   $\Leftarrow$) $f(\bar{0}) = f(\bar{1}-\bar{1}) = 1 - f(\bar{1}) = 0$.
3. $\Rightarrow$) Se sigue de los apartados anteriores y de que toda regla aditiva es descomponible.

   $\Leftarrow$) Dado $\bar{x} \in [0,1]^m$ se tiene $\bar{1}-\bar{x} \in [0,1]^m$ y $\bar{1} = \bar{x} + (\bar{1}-\bar{x})$; luego

   $1 = f(\bar{1}) = f(\bar{x}+(\bar{1}-\bar{x})) = f(\bar{x})+f(\bar{1}-\bar{x})$. ∎

## 3 Resultados de caracterización

A continuación caracterizamos todas las reglas de agregación neutrales que son descomponibles y respetan el anonimato (Teorema 2). Son todas aquéllas en las que la intensidad colectiva de preferencia es la media aritmética de los valores que asigna una aplicación del intervalo unidad a los niveles individuales de intensidad de preferencia. A medida que se añaden propiedades a las reglas de agregación, éstas tienen su reflejo en la aplicación del intervalo unidad. Así, la regla de la media aritmética es la única que verifica el respeto a la unanimidad (Corolario 1). Además, caracterizamos las reglas de agregación que satisfacen el resto de propiedades consideradas en este trabajo.

**Teorema 2**

Si $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ es la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$, son equivalentes:

1. $F$ es descomponible y respeta el anonimato.
2. Existe una única aplicación $\varphi : [0,1] \longrightarrow [0,1]$ con $\varphi(0) = 0$, tal que

$$f(\bar{x}) = \frac{1}{m}\sum_{i=1}^{m}\varphi(x_i) \quad \forall \bar{x} \in [0,1]^m.$$

DEMOSTRACIÓN:

$1 \Rightarrow 2$: Sea $\varphi : [0,1] \longrightarrow [0,1]$ la aplicación definida por $\varphi(c) = f(\bar{c})$ para todo $c \in [0,1]$. Entonces, por el primer apartado de la Prop. 3, tenemos $\varphi(0) = f(\bar{0}) = 0$. Además, para cualquier $\bar{x} \in [0,1]^m$, se verifica

$$mf(\bar{x}) = mf(x_1, 0, \ldots, 0) + \cdots +$$
$$+mf(0, \ldots, 0, x_m) = f(\bar{x}_1) + \cdots + f(\bar{x}_m) =$$
$$= \varphi(x_1) + \cdots + \varphi(x_m).$$

Respecto a la unicidad, sea $\psi : [0,1] \longrightarrow [0,1]$ una aplicación que verifica lo requerido. Entonces, para cualquier $c \in [0,1]$ se tiene

$$\varphi(c) = f(\bar{c}) = \frac{1}{m}m\psi(c) = \psi(c).$$

$2 \Rightarrow 1$: Obviamente, $F$ respeta el anonimato. Por otro lado, para cualquier $\bar{x} \in [0,1]^m$, se verifica

$$f(x_1, 0, \ldots, 0) + \cdots + f(0, \ldots, 0, x_m) =$$
$$= \frac{1}{m}\Big(\varphi(x_1) + \varphi(0) + \cdots + \varphi(0)\Big) + \cdots +$$
$$+\frac{1}{m}\Big(\varphi(0) + \cdots + \varphi(0) + \varphi(x_m)\Big) =$$
$$= \frac{1}{m}\Big(\varphi(x_1) + \cdots + \varphi(x_m)\Big) = f(\bar{x}). \quad \blacksquare$$

**Definición.** La función $f : [0,1]^m \longrightarrow [0,1]$, definida por

$$f(x_1, \ldots, x_m) = \frac{x_1 + \cdots + x_m}{m},$$

induce una regla de agregación neutral que denominaremos *media aritmética*.

Hemos de observar que la media aritmética ha sido utilizada en la agregación de preferencias difusas, entre otros por Blin–Whinston [3], Bezdek–Spillman–Spillman [2] y Nurmi [6].

**Corolario 1**

Si $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ es la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$, son equivalentes:

1. $F$ es descomponible y respeta la unanimidad y el anonimato.
2. $F$ es la media aritmética.

DEMOSTRACIÓN:

$1 \Rightarrow 2$: Si en la demostración del Teorema 2 consideramos que $F$ respeta la unanimidad, se tiene $\varphi(c) = f(\bar{c}) = c$ para cualquier $c \in [0,1]$, es decir $\varphi$ es la aplicación identidad y, por tanto, $F$ es la regla de la media aritmética.

$2 \Rightarrow 1$: Obvio. $\blacksquare$

Hemos de señalar que un resultado similar al del Corolario 1 ha sido probado por Candeal–Induráin–Uriarte [4] fuera del marco de las preferencias difusas, para conjuntos de opciones dotados de la estructura de espacio vectorial topológico.

**Corolario 2**

Si $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ es la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$, son equivalentes:

1. $F$ es aditiva y respeta el anonimato.
2. Existe una única aplicación $\varphi : [0,1] \longrightarrow [0,1]$ que satisface $\varphi(x+y) = \varphi(x) + \varphi(y)$ para cualesquiera $x, y \in [0,1]$ tales que $x + y \in [0,1]$, de forma que

$$f(\bar{x}) = \frac{1}{m}\sum_{i=1}^{m}\varphi(x_i) \quad \forall \bar{x} \in [0,1]^m.$$

DEMOSTRACIÓN: Consideramos el Teorema 2 y su demostración.

$1 \Rightarrow 2$: Para aplicar el Teorema 2 tenemos en cuenta que toda regla de agregación aditiva es descomponible. Sean $x, y \in [0,1]$ tales que $x+y \in [0,1]$. Entonces $\overline{x+y} \in [0,1]^m$ y

$$\varphi(x+y) = f(\overline{x+y}) = f(\bar{x}+\bar{y}) =$$
$$= f(\bar{x}) + f(\bar{y}) = \varphi(x) + \varphi(y).$$

$2 \Rightarrow 1$: De la aditividad de $\varphi$ se sigue $\varphi(0) = 0$. Entonces, por el Teorema 2, $F$ respeta el anonimato. La aditividad de $F$ es consecuencia inmediata de la de $\varphi$. ∎

**Corolario 3**

Si $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ es la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$, son equivalentes:

1. $F$ es descomponible, respeta el anonimato y es monótona.
2. Existe una única aplicación $\varphi : [0,1] \longrightarrow [0,1]$ que es creciente y verifica $\varphi(0) = 0$, de manera que

$$f(\bar{x}) = \frac{1}{m}\sum_{i=1}^{m} \varphi(x_i) \quad \forall \bar{x} \in [0,1]^m.$$

DEMOSTRACIÓN: Consideramos el Teorema 2 y su demostración.

$1 \Rightarrow 2$: Sólo falta comprobar que $\varphi$ es creciente. Sean $x, y \in [0,1]$ tales que $x \leq y$; entonces $\bar{x} \leq \bar{y}$ y $\varphi(x) = f(\bar{x}) \leq f(\bar{y}) = \varphi(y)$.

$2 \Rightarrow 1$: Sólo es necesario probar que $F$ es monótona. Sean $\bar{x}, \bar{y} \in [0,1]^m$ tales que $\bar{x} \leq \bar{y}$; entonces $x_i \leq y_i$ y $\varphi(x_i) \leq \varphi(y_i)$ para todo $i \in \{1, \ldots, m\}$. Luego

$$f(\bar{x}) = \frac{1}{m}\sum_{i=1}^{m} \varphi(x_i) \leq \frac{1}{m}\sum_{i=1}^{m} \varphi(y_i) = f(\bar{y}). \quad ∎$$

Otra consecuencia inmediata del Teorema 2 es la siguiente.

**Corolario 4**

Si $F : \mathcal{R}_r(X)^m \longrightarrow \mathcal{R}(X)$ la regla de agregación neutral asociada a $f : [0,1]^m \longrightarrow [0,1]$, son equivalentes:

1. $F$ es descomponible, respeta el anonimato y es recíproca.
2. Existe una única aplicación $\varphi : [0,1] \longrightarrow [0,1]$ que verifica $\varphi(0) = 0$ y $\varphi(1-x) = 1 - \varphi(x)$ para todo $x \in [0,1]$, de manera que

$$f(\bar{x}) = \frac{1}{m}\sum_{i=1}^{m} \varphi(x_i) \quad \forall \bar{x} \in [0,1]^m. \quad ∎$$

## Referencias


[1] ARROW, K.J. (1963): *Social Choice and Individual Values*. Second edition. Yale University Press, New Hagen.

[2] BEZDEK, J.C. – SPILLMAN, B. – SPILLMAN, R. (1979): "Fuzzy relation spaces for group decision theory: an application". *Fuzzy Sets and Systems* 2, pp. 5-14.

[3] BLIN, J.M. – WHINSTON, A.B. (1974): "Fuzzy sets and social choice". *Journal of Cybernetics* 3, pp. 28-36.

[4] CANDEAL, J.C. – INDURÁIN, E. – URIARTE, J.R. (1992): "Some issues related to the topological aggregation of preferences". *Social Choice and Welfare* 9, pp. 213-227.

[5] MAY, K.O. (1952): "A set of necessary and sufficient conditions for simple majority decision". *Econometrica* 20, pp. 680-684.

[6] NURMI, H. (1981): "Approaches to collective decision making with fuzzy preference relations". *Fuzzy Sets and Systems* 6, pp. 249-259.

[7] OVCHINNIKOV, S.V. (1990): "Means and social welfare function in fuzzy binary relation spaces". En KACPRZYK, J. – FEDRIZZI, M. (editores) (1990): *Multiperson Decision Making Using Fuzzy Sets and Possibility Theory*, pp. 143-154. Kluwer Academic Publishers, Dordrecht.

# Una solución al problema de la División Relacional Difusa

**José Galindo Gómez**
Dpto. Lenguajes y Ciencias de la Computación.
Universidad de Málaga
ppgg@lcc.uma.es

**Juan Miguel Medina Rodríguez**
Dpto. Ciencias de la Computación e I.A.
Universidad de Granada

**Mª Carmen Aranda Garrido**
Dpto. Lenguajes y Ciencias de la Computación.
Universidad de Málaga

## Resumen

En este artículo presentamos una posible forma para calcular el operador División Relacional en Bases de Datos Relacionales Difusas, partiendo del modelo GEFRED. Para ello, presentamos una ampliación del Álgebra Relacional Difuso para Bases de Datos Relacionales Difusas. Esta ampliación se basa en dos nuevos operadores de consulta que, por sus características, resultan muy útiles: La Intersección Difusa Cualificada y la Proyección Difusa con Funciones de Grupo. El primer operador mide el grado con el que cada tupla de una relación pertenece a la intersección de esa relación con otra. El segundo operador nos permite utilizar funciones de agregación en los atributos de una relación. De cada operador presentado se incluye un ejemplo ilustrativo.

**Palabras Clave:** Almacenamiento y Recuperación de Información, Consultas Fuzzy, Bases de Datos Fuzzy, Álgebra Relacional Fuzzy, División Relacional Fuzzy.

## 1 Introducción

El modelo GEFRED [2] [3], representa una síntesis entre los diferentes modelos aparecidos para tratar el problema de representación y tratamiento de información difusa mediante bases de datos relacionales. Una de las principales ventajas de este modelo es que consiste en una abstracción general que permite tratar diferentes enfoques, incluso aunque estos puedan parecer muy dispares. GEFRED se basa en la definición de lo que llama *Dominio Difuso Generalizado* ($D$) y *Relación Difusa Generalizada* ($R$), que incluyen los dominios y las relaciones clásicas respectivamente:

**Definición 1.1** *Si $U$ es el universo o dominio de discurso, $\widetilde{\mathcal{P}}(U)$ el conjunto de todas las distribuciones de posibilidad definidas sobre $U$ (incluyendo los tipos* Unknown *y* Undefined*) y* NULL *es otro tipo, entonces, definimos* ***Dominio Difuso Generalizado*** *como $D \subseteq \widetilde{\mathcal{P}}(U) \cup$ NULL.*

*Los tipos* Unknown, Undefined *y* NULL *son definidos en el sentido de Umano, Fukami et al. de [1] y [4].*

**Definición 1.2** *Una* ***Relación Difusa Generalizada****, $R$, viene dada por un par de conjuntos: "Cabecera" ($\mathcal{H}$) y "Cuerpo" ($\mathcal{B}$). La "Cabecera" es un conjunto fijo de ternas atributo-dominio-compatibilidad (donde el último es opcional),*

$$\mathcal{H} = \{(A_1 : D_1\ [, C_1]), \ldots, (A_n : D_n\ [, C_n])\}$$

*donde a cada atributo $A_j$, le subyace un dominio difuso generalizado, no necesariamente distinto, $D_j$ ($j = 1, 2, \ldots, n$). $C_j$ es un "atributo de compatibilidad" que toma valores en el intervalo [0,1].*

*El "Cuerpo" consiste en un conjunto de tuplas difusas generalizadas distintas, donde cada tupla está compuesta por un conjunto de ternas atributo-valor-grado (el grado es opcional),*

$$\mathcal{B} = \{(A_1 : \widetilde{d}_{i1}\ [, c_{i1}]), \ldots, (A_n : \widetilde{d}_{in}\ [, c_{in}])\}$$

*con $i = 1, 2, \ldots, m$, siendo $m$ el número de tuplas de la relación, y donde $\widetilde{d}_{ij}$ representa el valor de dominio que toma la tupla $i$ sobre el atributo $A_j$, y $c_{ij}$ el grado de compatibilidad asociado a este valor.*

Los operadores de comparación también son redefinidos para adaptarse a la naturaleza difusa de los datos que manejamos. Con estos comparadores podremos establecer una relación entre distintos valores de los atributos de una relación. Por ejemplo, el comparador "aproximadamente igual" ($\Theta^{=}$) se podría modelar como:

$$\Theta^{=}(\widetilde{p}, \widetilde{p'}) = \sup_{d \in U} \min\ (\pi_{\widetilde{p}}(d), \pi_{\widetilde{p'}}(d)) \qquad (1)$$

donde $\widetilde{p}, \widetilde{p'} \in D$, y sus distribuciones de posibilidad asociadas son $\pi_{\widetilde{p}}$ y $\pi_{\widetilde{p'}}$ respectivamente. $U$ es el dominio de discurso sobre el que se construye el dominio difuso generalizado $D$ (ver Definición 1.1).

Sobre estas definiciones GEFRED redefine los operadores del álgebra relacional en el llamado **Álgebra Relacional Difuso Generalizado** [2][3]: Unión, Intersección, Diferencia, Producto cartesiano, Proyección, Selección y Reunión. El **grado de compatibilidad** del valor de un atributo concreto (en una tupla) se obtiene como consecuencia de los procesos de manipulación realizados sobre esa relación y expresa el grado en el que ese valor ha satisfecho la operación realizada en él. Para calcular División Relacional Difusa presentamos un método en el cual usamos dos nuevos operadores que definimos a continuación y que también son de gran utilidad para otras cuestiones:

# 2 Intersección Difusa Cualificada: $\cap_Q$

En el álgebra relacional clásico, si suponemos que tenemos dos relaciones clásicas $R$ y $R'$ compatibles respecto a la unión (con igual número y tipo de atributos), la intersección relacional, notada por $R \cap R'$, representa las tuplas de $R$ que pertenecen también a $R'$ (entendiendo la pertenencia desde un punto de vista crisp).

En una base de datos relacional difusa los conceptos anteriores ya no están tan claramente definidos: La igualdad de tuplas y la pertenencia a un conjunto son conceptos difusos. En este contexto, definimos la Intersección Difusa Cualificada de $R$ y $R'$, notada por $R \cap_Q R'$, que expresa la posibilidad de que existan las tuplas de $R$ en la relación $R'$ o, dicho de otro modo, la posibilidad de que las tuplas de $R$ estén en la intersección.

**Definición 2.1** *Sean $R$ y $R'$ dos relaciones difusas generalizadas, compatibles con respecto a la unión y ambas con grado $n$, dadas por:*

$$R = \begin{cases} \mathcal{H} = \{(A_1 : D_1\ [, C_1]), \ldots, (A_n : D_n\ [, C_n])\} \\ \mathcal{B} = \{(A_1 : \widetilde{d}_{i1}\ [, c_{i1}]), \ldots, (A_n : \widetilde{d}_{in}\ [, c_{in}])\} \end{cases}$$

$$R' = \begin{cases} \mathcal{H}' = \{(A'_1 : D_1\ [, C'_1]), \ldots, (A'_n : D_n\ [, C'_n])\} \\ \mathcal{B}' = \{(A'_1 : \widetilde{d}'_{k1}\ [, c'_{k1}]), \ldots, (A_n : \widetilde{d}'_{kn}\ [, c'_{kn}])\} \end{cases}$$

*con $i = 1, \ldots, m$ y $k = 1, \ldots, m'$, siendo $m$ y $m'$ las cardinalidades de $R$ y $R'$ respectivamente.*

*Entonces, la **Intersección Difusa Cualificada** de $R$ y $R'$, será igual a $R$, modificando los grados de compatibilidad de un atributo en todas sus tuplas:*

$$R \cap_Q R' = \begin{cases} \mathcal{H}_{\cap_Q} = & \{(A_1 : D_1, C_1), (A_2 : D_2, [C_2]), \\ & \ldots, (A_n : D_n, [C_n])\} \\ \mathcal{B}_{\cap_Q} = & \begin{cases} \mathcal{B}^v_{\cap_Q} = \mathcal{B}^v \\ \mathcal{B}^c_{\cap_Q} = \{c''_{i1}, [c_{i2}], \ldots, [c_{in}]\} \end{cases} \end{cases} \quad (2)$$

*con $i = 1, \ldots, m$ y donde $\mathcal{B}^v$ es la componente de valor (sin grados de compatibilidad) del cuerpo, $\mathcal{B}$, de $R$. Los grados de compatibilidad de la componente de compatibilidad $\mathcal{B}^c_{\cap_Q}$ se dejan igual salvo el del primer atributo (por definición) que se calcula de la siguiente forma: $\forall i = 1, \ldots, m$*

$$c''_{i1} = \max_{w=1,\ldots,m'} \{ \min_{q=1,\ldots,n} \{\Theta^{=}(\widetilde{d}_{iq}, \widetilde{d}'_{wq})\}\} \quad (3)$$

*Los $c''_{i1}$ almacenan el grado de posibilidad de que la tupla $i$ de $R$ exista en $R'$.*

Se puede mejorar mucho la eficiencia del proceso, si entre los atributos de $R$ y $R'$ existen $n' \geq 1$ con dominios crisp (numéricos o escalares simples), ya que calcular $\Theta^{=}$ en esos atributos es una simple igualdad y si los atributos son distintos (resultado 0) no hay que continuar comparando con dicha tupla. Observe que este operador no posee la propiedad conmutativa.

**Ejemplo 1** Sean las relaciones difusas generalizadas $R$ y $R'$ dadas por las tablas 1 y 2 respectivamente. Supondremos que el atributo `A` es un atributo con dominio crisp y que los atributos `B` y `C` son atributos difusos con las etiquetas lingüísticas de la figura 1. Si calculamos la Intersección Difusa Cualificada $R \cap_Q R'$ el resultado es el que se encuentra en la tabla 3.

Tabla 1: Ejemplo 1: Relación $R$.

| $\mathcal{H}$ | A | B | C |
|---|---|---|---|
| $\mathcal{B}$ | A1 | B1 | C1 |
| | A2 | B2 | C2 |
| | A2 | B3 | C2 |
| | A3 | B3 | C2 |
| | A3 | B1 | C3 |

Tabla 2: Ejemplo 1: Relación $R'$.

| $\mathcal{H}$ | A | B | C |
|---|---|---|---|
| $\mathcal{B}$ | A1 | B1 | C1 |
| | A1 | B2 | C1 |
| | A1 | B2 | C3 |
| | A2 | B3 | C3 |
| | A2 | B1 | C1 |
| | A3 | B3 | C3 |
| | A3 | B2 | C1 |

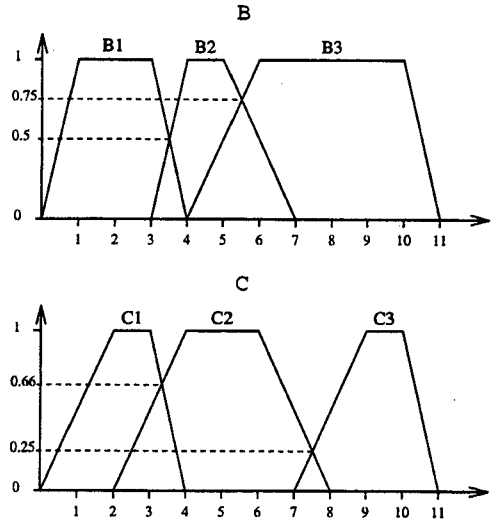



Figura 1: Etiquetas sobre los atributos B y C.

Tabla 3: Ejemplo 1: Relación $R \cap_Q R'$.

| $\mathcal{H}$ | A | $C_A$ | B | C |
|---|---|---|---|---|
| $\mathcal{B}$ | A1 | 1 | B1 | C1 |
| | A2 | 0.5 | B2 | C2 |
| | A2 | 0.25 | B3 | C2 |
| | A3 | 0.66 | B3 | C2 |
| | A3 | 0 | B1 | C3 |

El valor del atributo $C_A$ para una tupla concreta expresa la posibilidad de que esa tupla se encuentre en la relación $R'$. □

# 3 Proyección Difusa con Funciones de Grupo $\mathcal{F}$: $\mathcal{P}^{\mathcal{F}}$

**Definición 3.1** *Supongamos que tenemos los siguientes cuatro elementos, definidos como sigue:*

*1) Una relación difusa generalizada $R$ dada por:*

$$R = \begin{cases} \mathcal{H} = \{(A_1 : D_1\ [, C_1]), \ldots, (A_n : D_n\ [, C_n])\} \\ \mathcal{B} = \{(A_1 : \widetilde{d}_{r1}\ [, c_{r1}]), \ldots, (A_n : \widetilde{d}_{rn}\ [, c_{rn}])\} \end{cases}$$

*con $r = 1, 2, \ldots, m$, siendo $m$ el número de tuplas (la cardinalidad) de la relación.*

*2) Una lista de elementos de $\mathcal{H}$, $X$, con $\alpha$ elementos, que notaremos como: $X = \{x_1, \ldots, x_\alpha\} : x_i \in \mathcal{H}$, $\forall i = 1, \ldots, \alpha$. En $X$ puede haber elementos repetidos en posiciones distintas.*

*3) Un subconjunto de $\mathcal{H}$, $X'$, con atributos con dominios crisp. Si al número de elementos de $X'$ le llamamos $\beta$, a este subconjunto lo notaremos como: $X' \subseteq \mathcal{H} : X' = \{x'_1, \ldots, x'_\beta\}$. En $X'$, como en cualquier conjunto, no se admiten elementos duplicados. Tanto en $X$ como en $X'$ sus elementos pueden ser atributos normales o atributos de compatibilidad.*

*4) Una lista de Funciones de Grupo[1], $\mathcal{F}$, definidas sobre los elementos de $X$. Por tanto, la lista $\mathcal{F}$ tiene $\alpha$ elementos, donde cada elemento es una función de grupo (no necesariamente distinta) definida sobre cada uno de los atributos de $X$ respectivamente: $\mathcal{F} = \{\mathtt{gfunc}_1, \ldots, \mathtt{gfunc}_\alpha\}$.*

*Ahora, calculamos $R'$ como la Proyección Difusa de $R$ sobre el conjunto de atributos $X'$:*

$$\mathcal{P}(R; X') = \begin{cases} \mathcal{H}' = X' \\ \mathcal{B}' = \{(x'_1 : \widetilde{d}'_{t1}), \ldots, (x'_\beta : \widetilde{d}'_{t\beta})\} \end{cases} \quad (4)$$

*con $t = 1, \ldots, m'$, siendo $m'$ el número de tuplas de la proyección. Esta proyección no plantea problemas ya que los atributos de $X'$ tienen dominios crisp.*

*Entonces, con los elementos anteriores, definimos la* ***Proyección Difusa de $R$ sobre $X'$, con Funciones de Grupo $\mathcal{F}$ sobre*** *$X$, como una relación difusa generalizada dada por: $\mathcal{P}^{\mathcal{F}}(R; X'; X) =$*

$$= \begin{cases} \mathcal{H}^{\mathcal{F}}_{\mathcal{P}} = & \{X'.x'_1, \ldots, X'.x'_\beta, X.x_1, \ldots, X.x_\alpha\} \\ \mathcal{B}^{\mathcal{F}}_{\mathcal{P}} = & \{(X'.x'_1 : \widetilde{d}'_{t1}), \ldots, (X'.x'_\beta : \widetilde{d}'_{t\beta}), \\ & (X.x_1 : \widetilde{d}^{\mathcal{F}}_{t1}), \ldots, (X.x_\alpha : \widetilde{d}^{\mathcal{F}}_{t\alpha})\} \end{cases}$$

*con $t = 1, \ldots, m'$, siendo $m'$ el número de tuplas de esa proyección y donde los $\widetilde{d}'_{tj}$ con $j = 1, \ldots, \beta$ son los mismos valores que los que se obtienen de la proyección $\mathcal{P}(R; X')$ expresada anteriormente (por eso los notamos igual). Ambas proyecciones tienen $m'$ tuplas.*

*Los $\widetilde{d}^{\mathcal{F}}_{ti}$ se obtienen aplicando el siguiente algoritmo:*

```
FOR t := 1 TO m' DO
   FOR i := 1 TO α DO
      G := ∅
      FOR r := 1 TO m DO
         IF (R'.d'_t1 = R.d'_r1 AND ···
            AND R'.d'_tβ = R.d'_rβ)
            G := G ∪ R.d_ri
         END IF
      END FOR
      d^F_ti := gfunc_i{G}
   END FOR
END FOR
```

[1]Por Funciones de Grupo entendemos aquellas funciones que se aplican a un grupo finito de valores dando como resultado un único valor. Los ejemplos más típicos de funciones de grupo son las que se usan para contar el número de elementos del grupo (`count`), calcular el valor máximo (`max`), el mínimo (`min`), la suma (`sum`), la media (`avg`), la varianza (`variance`), la desviación estándar (`stddev`)... Todas las funciones anteriores están definidas en el lenguaje SQL.

*donde los* $R.\widetilde{d'}_{rj}$ *con* $j = 1, \ldots, \beta$ *son los valores de* $X'$ *en la* $r$*-éxima tupla de* $R$.

Por comodidad, $\mathcal{H}^{\mathcal{F}}_{\mathcal{P}}$ puede también expresarse como:

$$\mathcal{H}^{\mathcal{F}}_{\mathcal{P}} = \{x'_1, \ldots, x'_\beta, \mathtt{gfunc}_1(x_1), \ldots, \mathtt{gfunc}_\alpha(x_\alpha)\} \quad (5)$$

**Ejemplo 2** Sean los siguientes 4 elementos: 1) Una relación difusa generalizada $R$: tabla 4. Hemos considerado todos sus atributos con dominios crisp para simplificar el ejemplo. 2) Una lista de elementos de $\mathcal{H}$, $X$, con 3 elementos: $X = \{\mathtt{B}, \mathtt{B}, \mathtt{C}\}$. 3) Un subconjunto de $\mathcal{H}$, $X'$, con un elemento: $X' = \{\mathtt{A}\}$. 4) Una lista de Funciones de Grupo, $\mathcal{F}$, definidas sobre los elementos de $X$. Por tanto, la lista $\mathcal{F}$ tendrá 3 elementos (las funciones de grupo mínimo, media y suma de un grupo): $\mathcal{F} = \{\mathtt{min}, \mathtt{avg}, \mathtt{sum}\}$.

Tabla 4: Ejemplo 2: Relación $R$.

| $\mathcal{H}$ | A | B | C |
|---|---|---|---|
| $\mathcal{B}$ | A1 | 0.5 | 1 |
| | A1 | 0.6 | 2 |
| | A1 | 0.7 | 3 |
| | A2 | 0.3 | 4 |
| | A2 | 0.5 | 5 |

Tabla 5: Ejemplo 2: Resultado de $\mathcal{P}^{\mathcal{F}}(R; X; X')$.

| $\mathcal{H}$ | A | min(B) | avg(B) | sum(C) |
|---|---|---|---|---|
| $\mathcal{B}$ | A1 | 0.5 | 0.6 | 6 |
| | A2 | 0.3 | 0.4 | 9 |

Si con los datos anteriores hallamos la Proyección:

$$\mathcal{P}^{\mathcal{F}}(R; X'; X) = \mathcal{P}^{\{\mathtt{min},\mathtt{avg},\mathtt{sum}\}}(R; \{\mathtt{A}\}; \{\mathtt{B}, \mathtt{B}, \mathtt{C}\})$$

obtendremos la relación que se expresa en la tabla 5. La cabecera está en el formato de la ecuación 5. □

La potencia de la Proyección Difusa con Funciones de Grupo nos permite resolver fácilmente cuestiones que, de otra forma, serían más complicadas. Por ejemplo, nos permite solventar cuestiones como la siguiente: *Qué alumnos son Buenos (con grado mínimo 0.8) en 2 o más asignaturas.*

La Proyección Difusa de $R$ sobre el conjunto de atributos $X'$, con Funciones de Grupo $\mathcal{F}$ sobre los atributos de $X$, $\mathcal{P}^{\mathcal{F}}(R; X'; X)$, modela en álgebra relacional difuso generalizado lo que en SQL se efectúa con la cláusula GROUP BY de una sentencia SELECT. Así, la proyección $\mathcal{P}^{\mathcal{F}}(R; X'; X)$ podría ser fácilmente traducida a una sentencia SELECT de SQL, actuando cada cláusula sobre los siguientes elementos:

- **SELECT**: Conjunto $X'$ y cada función de $\mathcal{F}$ sobre cada atributo de $X$ respectivamente.
- **FROM**: La relation $R$.
- **GROUP BY**: Atributos de $X'$.

# 4 División Relacional Difusa: ÷

**Definición 4.1** *Sean* $R$ *y* $R'$ *dos relaciones:*

$$R = \begin{cases} \mathcal{H} = & \{(A_1 : D_1\ [, C_1]), \ldots, \\ & (A_{n'} : D_{n'}\ [, C_{n'}]), \ldots, (A_n : D_n\ [, C_n])\} \\ \mathcal{B} = & \{(A_1 : \widetilde{d}_{i1}\ [, c_{i1}]), \ldots, \\ & (A_{n'} : \widetilde{d}_{in'}\ [, c_{in'}]), \ldots, (A_n : \widetilde{d}_{in}\ [, c_{in}])\} \end{cases}$$

$$R' = \begin{cases} \mathcal{H}' = & \{(A'_{n'+1} : D_{n'+1}\ [, C_{n'+1}]), \ldots, \\ & (A'_n : D_n\ [, C_n])\} \\ \mathcal{B}' = & \{(A'_{n'+1} : \widetilde{d'}_{kn'+1}\ [, c'_{kn'+1}]), \ldots, \\ & (A_n : \widetilde{d'}_{kn}\ [, c'_{kn}])\} \end{cases}$$

*con* $i = 1, \ldots, m$ *y* $k = 1, \ldots, m'$, *siendo* $m$ *y* $m'$ *las cardinalidades de* $R$ *y* $R'$ *respectivamente y donde* $n$ *y* $n - n'$ *son los respectivos grados. Además, tenemos que* $1 \leq n' < n$, *tal que los* $D_c$, *con* $c = 1, \ldots, n'$, *son dominios crisp (numéricos o escalares simples).*

*Entonces, definimos la* ***División Relacional Difusa Generalizada*** *de* $R$ *entre* $R'$, *notada por* $R \div R'$, *como otra relación difusa generalizada obtenida por las siguientes operaciones:*

1. *Calculamos una relación* $R''$*:*

$$R'' = \mathcal{P}(R; A) \times R' \quad (6)$$

*con* $A = \{A_1, \ldots, A_{n'}\}$. *La proyección sobre* $A$ *no plantea problemas, ya que todos los atributos de* $A$ *tienen dominios crisp. El producto cartesiano difuso se efectúa de la forma tradicional.*

2. *Calculamos* $R'''$ *como*

$$R''' = R'' \cap_Q R \quad (7)$$

3. *Por último, tenemos que*

$$R \div R' = \mathcal{P}^{\mathcal{F}}(R'''; A; C_1) \quad (8)$$

*donde* $C_1$ *es el primer atributo de compatibilidad de* $A$, *y* $\mathcal{F}$ *es una lista con una función: La función de grupo "mínimo",* $\mathcal{F} = \{\mathtt{min}\}$.

*Esta Proyección con Funciones de Grupo no plantea ninguna problemática, ya que tanto* $A$ *como* $C_1$ *son atributos con dominios crisp.*

La División Relacional Difusa Generalizada es una extensión de la División Relacional clásica sobre atributos crisp y, por tanto, incluye a esta. O sea, sobre relaciones clásicas (sin atributos difusos), ambas obtienen idénticos resultados (aunque de distinta forma). En otras palabras, en bases de datos clásicas esta sería también otra forma de calcular la división relacional.

**Ejemplo 3** Supongamos que tenemos una base de datos Relacional Difusa sobre jugadores de baloncesto. Una relación de la base de datos podría tener los atributos (`JUGADOR`, `EQUIPO`, `ALTURA`, `CALIDAD`, `NUM_CAMISETA`...). Los campos `ALTURA` (que almacena la altura del jugador) y `CALIDAD` (que mide la calidad del jugador según el número de puntos medio por partido), permiten almacenar valores difusos. Nosotros, para el ejemplo, usaremos las etiquetas lingüísticas de la figura 2. En este contexto, vamos a encontrar los equipos de baloncesto que tienen en sus fichajes todos los tipos de jugadores (en `ALTURA` y `CALIDAD`) que el equipo de Córdoba.

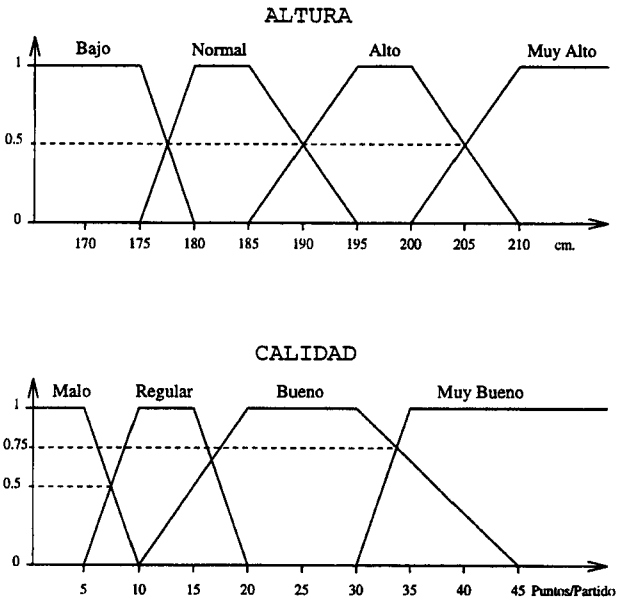



Figura 2: Etiquetas sobre los atributos `ALTURA` y `CALIDAD`.

Para encontrar dichos equipos, primero tomaremos una proyección de la relación anterior sobre los atributos que nos interesan (`EQUIPO`, `ALTURA` y `CALIDAD`), quedando una relación $R$ que ejemplarizamos en la tabla 6. Esta tabla no tiene atributos de compatibilidad puesto que sólo hemos efectuado una proyección de una relación base. La segunda relación $R'$ será el resultado de la proyección sobre los atributos `ALTURA` y `CALIDAD` después de haber efectuado una selección con la condición `EQUIPO`=Córdoba. En nuestro ejemplo, $R'$ contendrá sólo las dos tuplas de la tabla 7.

Tabla 6: Ejemplo 3: Relación $R$.

| $\mathcal{H}$ | EQUIPO | ALTURA | CALIDAD |
|---|---|---|---|
| $\mathcal{B}$ | Córdoba | Bajo | Muy Bueno |
| | Córdoba | Muy Alto | Malo |
| | Granada | Bajo | Muy Bueno |
| | Granada | Muy Alto | Malo |
| | Granada | Alto | Regular |
| | Málaga | Bajo | Muy Bueno |
| | Málaga | Alto | Malo |
| | Málaga | Muy Alto | Muy Bueno |
| | Sevilla | Bajo | Bueno |
| | Sevilla | Muy Alto | Malo |
| | Sevilla | Normal | Bueno |
| | Cádiz | Muy Alto | Muy Bueno |
| | Cádiz | Bajo | Bueno |
| | Huelva | Alto | Muy Bueno |

Tabla 7: Ejemplo 3: Relación $R'$.

| $\mathcal{H}$ | ALTURA | CALIDAD |
|---|---|---|
| $\mathcal{B}$ | Bajo | Muy Bueno |
| | Muy Alto | Malo |

Ahora, lo único que tenemos que hacer es calcular la División Difusa Generalizada $R \div R'$.

Si observamos ambas relaciones, podemos ver que en $R$ tenemos que el equipo de Granada cumple con las condiciones de dicha división. Obviamente, el equipo de Córdoba también. Ambos equipos tienen dos tuplas con los mismos valores que $R'$ (en `ALTURA` y `CALIDAD`).

El equipo de Málaga tiene una tupla que coincide con una de $R'$, (Bajo, Muy Bueno): Con compatibilidad 1 en `CALIDAD` y 1 en `ALTURA`. La otra tupla de $R'$, (Muy Alto, Malo), no empareja exactamente con ninguna otra tupla de Málaga. Sin embargo, vemos que no hay muchas diferencias: La compatibilidad con la tupla (Málaga, Alto, Malo) es de 1 en `CALIDAD` y de 0.5 en `ALTURA` (Alto y Muy Alto "se parecen" con grado 0.5). Por tanto, parece lógico que el equipo de Málaga debe aparecer en el resultado de la división, pues cumple "en parte" con los requisitos.

Igualmente, el equipo de Sevilla tiene una tupla que coincide con una de $R'$, (Muy Alto, Malo). La otra tupla de $R'$, (Bajo, Muy Bueno), no empareja exactamente con ninguna otra tupla de Sevilla, pero vemos que hay similitudes: La compatibilidad con la tupla (Sevilla, Bajo, Bueno) es de 1 en `ALTURA` y de 0.75 en `CALIDAD` (Bueno y Muy Bueno "se parecen" con grado 0.75), y la compatibilidad con la tupla (Sevilla, Normal, Bueno) es de 0.5 en `ALTURA` y de 0.75 en `CALIDAD`. Así, vemos que el equipo de Sevilla también cumple "en parte" con los requisitos de la división.

Después de este breve estudio informal de lo que bus-

camos, pasemos a ver cómo calcular $R \div R'$:

1. **Ecuación 6**: Hallar $R'' = \mathcal{P}(R; \texttt{EQUIPO}) \times R'$. El resultado es la tabla 8 sin $C_{\texttt{EQUIPO}}$.

Tabla 8: Ejemplo 3: Relación (parcial) $R''' = R'' \cap_Q R$.

| $\mathcal{H}$ | EQUIPO | $C_{\texttt{EQUIPO}}$ | ALTURA | CALIDAD |
|---|---|---|---|---|
| $\mathcal{B}$ | Córdoba | 1 | Bajo | Muy Bueno |
| | Córdoba | 1 | Muy Alto | Malo |
| | Granada | 1 | Bajo | Muy Bueno |
| | Granada | 1 | Muy Alto | Malo |
| | Málaga | 1 | Bajo | Muy Bueno |
| | Málaga | 0.5 | Muy Alto | Malo |
| | Sevilla | 0.75 | Bajo | Muy Bueno |
| | Sevilla | 1 | Muy Alto | Malo |
| | Cádiz | 0.75 | Bajo | Muy Bueno |
| | Cádiz | 0 | Muy Alto | Malo |
| | Huelva | 0 | Bajo | Muy Bueno |
| | Huelva | 0 | Muy Alto | Malo |

2. **Ecuación 7**: Calculamos la Intersección Difusa Cualificada, $R'' \cap_Q R$. Así, obtenemos la relación $R'''$ de la tabla 8. En el campo $C_{\texttt{EQUIPO}}$ tenemos la posibilidad de que las tuplas de $R''$ estén en $R$.

3. **Ecuación 8**: Para terminar:

$$R \div R' = \mathcal{P}^{\{\min\}}(R'''; \texttt{EQUIPO}; C_{\texttt{EQUIPO}}) \qquad (9)$$

Tabla 9: Ejemplo 3: Relación resultante de $R \div R'$.

| $\mathcal{H}$ | EQUIPO | $C_{\texttt{EQUIPO}}$ |
|---|---|---|
| $\mathcal{B}$ | Córdoba | 1 |
| | Granada | 1 |
| | Málaga | 0.5 |
| | Sevilla | 0.75 |
| | Cádiz | 0 |
| | Huelva | 0 |

El resultado final de la división está en la tabla 9. □

# 5 Conclusiones

Los operadores presentados son muy útiles para conseguir ejecutar la División Relacional en una base de datos Difusa. Además, sus resultados pueden añadirse a los de otras operaciones, como las presentadas en [5], [6] y [7] para obtener más información de una base de datos. Por ejemplo, podemos saber en qué medida es cierta la siguiente afirmación y, además, saber qué tuplas cumplen esa afirmación y en qué grado la cumple cada una: *La mayoría de los alumnos son Buenos (o Muy Buenos) en al menos una asignatura y Malos (o Muy Malos) en al menos otra asignatura.*

Hemos estudiado aquí la División Relacional en Bases de Datos Relacionales Difusas cuando los atributos no comunes a ambas relaciones son crisp. Para poder incluir ahí atributos con dominios difusos es necesario adoptar previamente un criterio para saber cuando dos valores difusos pueden ser considerados iguales.

Actualmente, estamos estudiando una propuesta que contribuya a reducir, en lo posible, el problema de la igualdad de conjuntos difusos y el problema de la redundancia en Bases de Datos Difusas. No obstante, la eliminación total de este problema, de forma única para todos los contextos, es imposible, por la naturaleza difusa de los datos en cuestión. ¿Cómo podemos solucionar automaticamente problemas que ni siquiera sabemos solucionar manualmente?

# Referencias


[1] S. Fukami, M. Umano, M. Muzimoto, H. Tanaka: "Fuzzy Database Retrieval and Manipulation Language", IEICE Technical Reports, Vol. 78, N° 233, pp. 65–72, AL-78-85 (Automata and Language) (1979).

[2] J.M. Medina, O. Pons, M.A. Vila: "GEFRED. A Generalized Model of Fuzzy Relational Data Bases". Information Sciences, 76(1–2), 87–109 (1994).

[3] J.M. Medina: "Bases de Datos relacionales difusas. Modelo teórico y aspectos de su implementación". Ph. Doctoral Thesis, University of Granada, May 1994.

[4] M. Umano: "Freedom-O: A Fuzzy Database System". Fuzzy Information and Decision Processes. Gupta-Sanchez edit. North-Holand Pub. Comp. (1982).

[5] M.A. Vila, J.C. Cubero, J.M. Medina, O. Pons: "Using OWA Operator in Flexible Query Processing". Volume "The Ordered Weighted Averaging Operators: Theory, Methodology and Applications". Eds. R.R. Yager and J. Kacprzyk.

[6] R.R. Yager: "Fuzzy Quotient Operators". Proc. of IPMU 1992, pp. 317–322.

[7] L.A. Zadeh: "A Computational approach to fuzzy quantifiers in natural languages". Computing and Mathematics with Applications. Vol. 9, pp. 149–184.

# Redes de Restricciones Temporales Disyuntivas Borrosas*

Isabel Navarrete, Roque Marín, Mercedes Balsa
Departamento de Informática y Sistemas
Universidad de Murcia
30.071-Espinardo (Murcia)
e-mail: inava@dif.um.es

## Resumen

En este trabajo generalizamos el modelo de restricciones temporales borrosas para permitir la introducción de restricciones binarias dadas por un conjunto finito de distribuciones de posibilidad normalizadas y convexas. Con esto se aumenta considerablemente la expresividad del modelo. Aunque las redes disyuntivas en general no son tratables, se estudia un caso especial como son las redes disyuntivas con estructura de árbol, para las que podemos encontrar la red mínima y obtener una solución en tiempo polinomial. Motivados por este hecho, introducimos un método para encontrar una posible descomposición en árbol de una red mínima y comprobar la minimalidad y descomponibilidad de una red camino consistente.

## 1 Introducción

El modelo de redes de restricciones temporales temporales (FTCN) introducido en trabajos previos [1,6,7] sólo permite expresar restricciones simples, representadas cada una mediante una distribución de posibilidad convexa. En ocasiones, especialmente para tareas de planificación, es necesario considerar un caso más general en el que las restricciones temporales sean no convexas.

La ampliación que haremos sobre el modelo FTCN, en la sección 2, consistirá en permitir que tanto las restricciones unarias como las binarias vengan determinadas, no por una única distribución, sino por un conjunto finito de distribuciones de posibilidad normalizadas y convexas. Con esto se aumenta considerablemente la expresividad del modelo FTCN, en cuanto que permite expresiones como: "$X_1$ está mucho antes o poco después que $X_2$". Además incluye al álgebra de puntos (PA) de Vilain y Kautz [12]. El problema de este tipo de redes es su ineficiencia, debido a que, en general, son redes no descomponibles [11], con lo que necesitamos un algoritmo de búsqueda con retroceso para encontrar una solución. Además, los problemas de determinar la consistencia y encontrar la red mínima no pueden ser resueltos en tiempo polinomial. Sin embargo, en la sección 3, mostramos cómo para redes disyuntivas con topología de árbol estos problemas sí son tratables. Proponemos un algoritmo que calcula la red mínima equivalente a una red con estructura de árbol en $O(n^2R^2)$ y mostramos cómo puede obtenerse una solución con un cierto grado de posibilidad en $O(nR)$.

Debido al buen comportamiento computacional de las redes disyuntivas con topología de árbol, abordamos en la sección 4 el problema de la descomponibilidad en árbol de una red disyuntiva. El método consiste en eliminar restricciones redundantes en la red original hasta encontra (si es posible) un árbol que represente a la red de forma exacta y sin pérdida de información. El algoritmo de descomposición en árbol, que tiene un coste $O(n^3R^2)$, cuando se aplica a redes disyuntivas mínimas, determina si la red es o no descomponible. En caso afirmativo, encuentra el árbol equivalente. Cuando partimos de redes camino consistentes el algoritmo no es completo pero puede servir de test para decidir cuando una red disyuntiva camino consistente es mínima y descomponible.

## 2 Redes de restricciones temporales disyuntivas borrosas

En este apartado vamos a generalizar el modelo de redes de restricciones temporales borrosas (FTCN) introducido en trabajos previos [1,6,7]. Recordemos que una FTCN es un par $N = \langle X, L \rangle$ donde $X$ es un conjunto finito de $n$ variables y $L$ un conjunto finito de restricciones unarias y binarias borrosas entre ellas. En ausencia de restricciones cada variable $X_i$ puede tomar cualquier valor numérico de un conjunto de instantes temporales precisos que llamaremos $\tau$. Una restric-

*Este trabajo ha sido realizado gracias al soporte de la CICyT a través del proyecto de investigación TIC95-0604

ción unaria $L_i$ sobre una variable temporal $X_i$ se define mediante una distribución de posibilidad $\pi_i$ sobre $\tau$, normalizada y unimodal, que restringe el dominio de valores que pueden ser asignados a $X_i$, a aquellos instantes de tiempo $t_i \in \tau$ tales que $\pi_i(t_i) > 0$. Cada restricción binaria $L_{ij}$ se define mediante una distribución de posibilidad $\pi_{ij}$, también normalizada y unimodal, que limita conjuntamente los dominios de los valores asignables a las variables $X_i$ y $X_j$. Para poder realizar las asignaciones $X_i = x_i$ y $X_j = x_j$ será necesario que $\pi_{ij}(x_j - x_i) > 0$.

Ahora vamos a ampliar el modelo FTCN permitiendo que tanto las relaciones unarias como las binarias estén determinadas por una unión finita de distribuciones de posibilidad normalizadas y convexas, con lo cual, las restricciones temporales serán no convexas.

**Definición 1** *Llamamos red de restricciones temporales disyuntivas borrosas (DFTCN) a un par* $N^d = \langle X, L^d \rangle$ *donde* $X = \{X_0, X_1, \ldots, X_{n-1}\}$ *es un conjunto finito de n variables temporales cuyo dominio es un conjunto de instantes precisos* $\tau$ *y* $L^d = \{L_i^d \mid 1 \leq i, \leq n-1\} \cup \{L_{ij}^d \mid 0 \leq i, j \leq n-1\}$ *es un conjunto finito de relaciones unarias y binarias sobre las variables de* $X$. *Cada relación unaria* $L_i^d$ *restringe el dominio de la variable* $X_i$ *y viene dada por* $L_i^d = \{\pi_i^1, \pi_i^2, \ldots, \pi_i^k\}$ *donde cada* $\pi_i^q$ *es una distribución de posibilidad normalizada y unimodal sobre* $\tau$. *Una relación binaria* $L_{ij}^d = \{\pi_{ij}^1, \pi_{ij}^2, \ldots, \pi_{ij}^m\}$ *describe los posibles valores de la diferencia entre las variables* $X_i$ *y* $X_j$ *mediante unión o disyunción de distribuciones de posibilidad normalizadas y convexas.*

Supondremos que las restricciones $L_{ij}^d$ y $L_{ji}^d$ se definen de forma simétrica, para ello hacemos que,

$$L_{ji}^d = \left(L_{ij}^d\right)^{-1} = \{\pi_{ji}^1, \pi_{ji}^2, \ldots, \pi_{ji}^k\}$$
$$\forall\, 1 \leq q, \leq k \text{ debe ser } \pi_{ij}^q(m) = \pi_{ji}^q(-m)$$

Por otra parte, para cada $L_{ii}^d$ debe cumplirse $L_{ii}^d = \pi_0$ donde $\pi_0(0) = 1$, y $\pi_0(m) = 0$ para $m \neq 0$.

En lo que sigue supondremos varias simplificaciones que redundarán en un mejor tratamiento computacional:

- El dominio de las variables temporales se considera proyectado sobre un eje unidimensional discreto $\tau = \{t_0, t_1, t_2, \ldots\}$, donde $\forall i \in \mathbb{N}$, $t_i$ representa un instante preciso de tiempo y $t_0$ hace referencia al origen de tiempos, que por simplicidad tomaremos $t_0 = 0$. Suponemos que los sucesivos instantes de tiempo son equidistantes, esto es, $\forall i \in \mathbb{N}$, $t_{i+1} - t_i = \Delta t$ y cada $t_i$ reflejará una distancia $i \times \Delta t$ al origen de tiempos. A la cantidad constante $\Delta t$ la llamaremos *factor de máxima precisión* y para una aplicación concreta se elegirá de forma que consideremos simultáneos cualesquiera dos hechos que ocurran en instantes de tiempo que disten entre sí menos de $\Delta t$. Esto no resulta una limitación frente a considerar un eje de tiempos continuo, siempre que se pueda aumentar el nivel de precisión tanto como sea necesario. Con esta simplificación, cada restricción binaria $L_{ij}^d$ tendrá como universo el conjunto de los enteros $\mathbb{I}$. Además, podemos transformar cualquier restricción unaria $L_i^d$ sobre $\tau$ a una restricción binaria equivalente $L_{0i}^d$ sobre $\mathbb{I}$ y considerando que a $X_0$ le asignamos el valor $t_0 = 0$, tenemos que $L_{0i}^d = L_i^d$.

- Cada distribución de posibilidad $\pi_{ij}$ sobre $\mathbb{I}$ la aproximaremos mediante una distribución trapezoidal que coincide con la función de pertenencia $\mu_A$ asociada al subconjunto borroso $A = (\alpha, \beta, \gamma, \delta)$ de $\mathbb{I}$, donde $\beta$ y $\gamma$ representan, respectivamente el inicio y el final del núcleo, y $\alpha$ y $\delta$ el principio y el final del soporte. Cuando trabajamos con distribuciones de posibilidad convexas y normalizadas, como es nuestro caso, la imprecisión queda prácticamente caracterizada por la posición y extensión del núcleo y el soporte. Por tanto, asumir distribuciones trapezoidales no supone nuevas restricciones [4].

- Suponemos que las restricciones $L_{ij}^d$ están expresadas en *forma canónica*, de manera que para cualesquiera dos conjuntos borrosos $A_1 = (\alpha_1, \beta_1, \gamma_1, \delta_1)$ y $A_2 = (\alpha_2, \beta_2, \gamma_2, \delta_2)$ correspondientes a dos distribuciones de $L_{ij}^d$ se cumple que $A_1$ no es subconjunto difuso de $A_2$, ni $A_2$ es subconjunto de $A_1$.

Una DFTCN puede representarse mediante un grafo de restricciones dirigido $G = (V, E)$, donde los nodos corresponden a las variables temporales y los arcos corresponden a restricciones sobre las variables que conectan. Por simplicidad, en el grafo no se representan los lazos correspondientes a las restricciones $L_{ii}^d$ y si incluimos un arco $e$ etiquetado con $L_{ij}^d$, que parte del nodo $X_i$ y llega al nodo $X_j$, entonces no se introduce el arco correspondiente a la restricción simétrica $L_{ji}^d$. Además, si no existe ninguna restricción entre dos variables, o lo que es lo mismo, su distribución asociada es la restricción universal $\pi_U(m) = 1, \forall\, m \in \mathbb{I}$, entonces, los nodos correspondientes no estarán conectados por un arco.

**Definición 2** *Sean* $L_{ik}^d = \{A_1, \ldots, A_p\}$, $L_{kj}^d = \{B_1, \ldots, B_q\}$ *dos restricciones disyuntivas dadas por los conjuntos difusos asociados a las correspondientes distribuciones de posibilidad trapezoidales. Se define la composición de las restricciones* $L_{ik}^d$, $L_{kj}^d$ *como una nueva restricción disyuntiva* $L_{ik}^d \odot L_{kj}^d$, *sobre las variables* $X_i$, $X_j$ *que llamaremos restricción inducida por*

$L_{ik}^d$, $L_{kj}^d$ *y se calcula como*

$$L_{ik}^d \odot L_{kj}^d = \bigcup_{\substack{x=1\ldots p \\ y=1\ldots q}} A_x \oplus B_y \qquad (1)$$

*donde* $A_x \oplus B_y$ *es la suma difusa de los conjuntos* $A_x$ *y* $B_y$.

**Definición 3** *Sean* $H_{ij}^d = \{A_1, \ldots, A_p\}$, $L_{ij}^d = \{B_1, \ldots, B_q\}$ *dos restricciones disyuntivas. Se define la intersección de restricciones disyuntivas como una nueva restricción disyuntiva* $H_{ij}^d \sqcap L_{ij}^d$, *sobre las variables* $X_i$, $X_j$ *que se calcula como*

$$H_{ij}^d \sqcap L_{ij}^d = \bigcup_{\substack{x=1\ldots p \\ y=1\ldots q}} A_x \cap B_y \qquad (2)$$

*donde* $A_x \cap B_y$ *es la intersección difusa de los conjuntos* $A_x$ *y* $B_y$.

Puede probarse que $\left|L_{ik}^d \odot L_{kj}^d\right| \leq p \times q$ y $\left|H_{ij}^d \sqcap L_{ij}^d\right| \leq p+q$. Una observación a tener en cuenta es que en (1), al calcular $L_{ik}^d \odot L_{kj}^d$ puede que la restricción resultante no esté en forma canónica. En ese caso conviene obtener la representación canónica con objeto de que el número de conjuntos borrosos en la restricción disyuntiva sea mínino.

**Definición 4** *El rango [2] de una red DFTC* $N^d$, *es un valor entero* $R$ *donde*

$$R = \max_{0\leq i,j\leq n-1} R_{ij} \text{ donde } R_{ij} = \max_{A_p, A_q \in L_{ij}} (\delta_q - \alpha_p)$$

**Definición 5** *Decimos que una asignación* $X_i = x_i$, $X_j = x_j$ *donde* $x_i, x_j \in \tau$ *satisface la restricción* $L_{ij}^d = \{\pi_{ij}^1, \pi_{ij}^2, \ldots, \pi_{ij}^k\}$ *entre las variables* $X_i$ *y* $X_j$ *si y sólo si*

$$\exists\, \pi_{ij}^q \in L_{ij}^d \text{ tal que } \pi_{ij}^q(x_j - x_i) > 0 \qquad (3)$$

*El máximo grado de posibilidad con que la asignación* $X_i = x_i$, $X_j = x_j$ *satisface* $L_{ij}^d$ *es*

$$\sigma_{ij}^{\max}(x_i, x_j) = \max_{1\leq q\leq k} \pi_{ij}^q(x_j - x_i)$$

La razón de definir un máximo grado de posibilidad se justifica porque en (3) la distribución $\pi_{ij}^q$ no tiene por qué ser única, ya que puede darse el caso de que exista solapamiento entre núcleos o soportes en los conjuntos borrosos asociados a las distribuciones de $L_{ij}^d$.

**Definición 6** *Una solución* $\sigma$*-posible para una DFTCN es una n-tupla* $s = (x_0, x_1, \ldots, x_{n-1}) \in \tau^n$ *que verifica* $\pi_s(s) = \sigma$, *donde* $\pi_s$ *es la distribución de posibilidad que define el conjunto de posibles soluciones de la red y viene dada por*

$$\pi_s(s) = \min_{0\leq i,j\leq n-1} \sigma_{ij}^{\max}(x_i, x_j)$$

**Definición 7** *Decimos que una DFTCN* $N^d$ *es* $\alpha$*-consistente si existe al menos una solución* $\alpha$*-posible. En particular diremos que una DFTCN es consistente si existe al menos una solución* $s$ *completamente posible, esto es, debe ser* $\pi_s(s) = 1$ *y por tanto la distribución* $\pi_s$ *está normalizada.* $N^d$ *es inconsistente si no hay solución* ($\alpha = 0$).

Podemos definir un orden parcial $\sqsubseteq$ entre restricciones binarias disyuntivas, de forma que una restricción binaria $H_{ij}^d$ es *menos imprecisa* que otra $L_{ij}^d$ y lo notamos como $H_{ij}^d \sqsubseteq L_{ij}^d$ si y sólo si para cada conjunto borroso $A_p$ en $H_{ij}^d$ existe un conjunto borroso $B_q$ en $L_{ij}^d$ tal que $A_p$ esta incluido en $B_q$. El orden $\sqsubseteq$ entre restricciones disyuntivas induce un orden parcial $\subseteq$ entre redes disyuntivas de forma que podemos decir que una red disyuntiva $N_1^d$ es *más reducida* que otra $N_2^d$ (con el mismo conjunto de variables) y lo notamos $N_1^d \subseteq N_2^d$, si todas las correspondientes restricciones satisfacen el orden parcial $\sqsubseteq$, esto es, $\forall\, i, j \quad H_{ij}^d \sqsubseteq L_{ij}^d$.

Al igual que con redes FTC, podemos definir una relación de equivalencia entre redes DFTC con el mismo conjunto de variables y diremos que dos redes $N_1^d$ y $N_2^d$ son *equivalentes* si y sólo si toda solución $\sigma$-posible de una de ellas es también solución $\sigma$-posible de la otra. De todas las redes equivalentes a $N^d$ nos interesa la más reducida respecto al orden $\subseteq$, que llamaremos *red mínima* asociada a $N^d$. Para el caso de redes FTC, en otros trabajos [8] hemos demostrado que la red minima es única, descomponible [11] y puede obtenerse en tiempo polinomial. En el modelo DFTCN podemos asegurar la unicidad de la red disyuntiva mínima, sin embargo, el problema de determinar la consistencia en una red DFTC es NP-completo y la red mínima no puede obtenerse en tiempo polinomial. Además, otro hecho poco deseable, es que se pierde la descomponibilidad de la red mínima, con lo cual, para obtener una solución de la red es necesario aplicar un algoritmo de búsqueda con retroceso. Todos estos resultados pueden demostrarse fácilmente a partir del trabajo de Dechter et al. [2]

Las redes DFTC pueden tratarse como un conjunto de redes simples FTC, que son generadas seleccionando para cada arco una de las distribuciones de posibilidad que componen la restricción disyuntiva correspondiente. Y la red DFTC mínima puede obtenerse segun nos indica el siguiente teorema.

**Teorema 1** *La red mínima,* $M^d$, *correspondiente a una DFTCN,* $N^d$, *satisface* $M^d = \cup_i M_i$, *donde* $M_i$

*es la red mínima equivalente a una red FTC extraída de $N^d$, y la unión se toma entre todas las FTCN generadas a partir $N^d$.*

Si tenemos como mucho $k$ distribuciones de posibilidad en cada disyunción y existen $p$ restricciones disyuntivas en una red $N^d$ de $n$ variables, la red mínima $M^d$ equivalente a $N^d$ puede obtenerse en $O(n^3k^p)$, ya que tendríamos que calcular $k^p$ minimizaciones de redes simples FTC y cada una requiere $n^3$ pasos. Debido al tiempo exponencial que se requiere para obtener una red disyuntiva mínima, debemos pensar en cómo podemos resolver este problema en la práctica. Esto lo veremos en las próximas secciones.

# 3 Redes DFTC relativas

En esta sección ampliamos el concepto de redes de restricciones temporales relativas imprecisas (RFCN) introducido en [9] y veremos cómo a partir de un grafo de restricciones, correspondiente a una DFTCN que tenga topología de árbol, puede obtenerse la red mínima y una solución en tiempo polinomial.

**Definición 8** *Una red DFTC relativa $N^d$ es aquella cuyo grafo de restricciones tiene estructura de árbol.*

Dada una red DFTC $N^d$ siempre se cumple que su red mínima $M^d$ equivalente verifica que $\forall\, 0 \leq i,j,k \leq n-1$, $M^d_{ij} \sqsubseteq M^d_{ik} \odot M^d_{kj}$, donde $M^d_{ij}$, $M^d_{ik}$, $M^d_{kj}$ son relaciones disyuntivas mínimas de la red $M^d$. Si tenemos que, en concreto, $M^d_{ij} = M^d_{ik} \odot M^d_{kj}$ entonces decimos que la restricción mínima $M^d_{ij}$ es *redundante*. Se puede comprobar que para una red DFTC relativa $N^d$ su red mínima $M^d$ cumple que para cada par de variables $X_i$, $X_j$ no restringidas existe una variable $X_r$ tal que

$$M^d_{ij} = M^d_{ir} \odot M^d_{rj} \qquad (4)$$

donde $X_r$ es la raiz del subárbol más pequeño que conecta las variables $X_i$ y $X_j$. Al calcular la red mínima se obtiene un grafo de restricciones completo, pero las restricciones originales no universales se mantienen exactamente iguales y las restantes son redundantes.

**Teorema 2** *La red mínima $M^d$ equivalente a una red DFTC relativa $N^d$ puede obtenerse en $O(n^2R^2)$.*

*Dem.*- Según Montanari [11] si una red de restricciones binarias tiene estructura de árbol entonces la red camino consistente equivalente es mínima. Por tanto, para redes DFTC relativas el problema de encontrar la red mínima sí es tratable. Bastaría con aplicar un algoritmo de camino consistencia [5] que revisa todos los caminos de longitud tres propagando las restricciones de forma adecuada hasta que se consigue una relajación de la red. Este proceso requiere como mínimo un tiempo $O(n^3)$, pero aprovechando la topología de la red DFTC relativa, proponemos un algoritmo (fig. 1) que calcula la red mínima de forma más eficiente.

---

1. begin
2. sea $T$ el árbol correspondiente a una DTCN relativa $N^d$;
3. sea $\langle X_0, \ldots, X_{n-1}\rangle$ una lista con las vbles. de $N^d$ desde nivel 0 de $T$ a nivel máximo;
4. desde $i = 0$ hasta $n-2$
5. desde $j = i+1$ hasta $n-1$
6. si $(i,j) \notin T$ entonces
7. $M^d_{ij} = M^d_{ipadre(i)} \odot M^d_{padre(i)j}$;
8. $M^d_{ji} \leftarrow \left(M^d_{ij}\right)^{-1}$;
9. en otro caso $M^d_{ij} \leftarrow L^d_{ij}$ y $M^d_{ji} \leftarrow \left(L^d_{ij}\right)^{-1}$;
10. end

---

Figura 1: Red mínima relativa

El algoritmo anterior sólo calcula las relaciones disyuntivas mínimas entre variables no restringidas inicialmente. El resto de restricciones, correspondientes a los arcos del árbol, se dejan como están porque ya son mínimas. Al recorrer los nodos del árbol por niveles, comenzando por la raíz, nos aseguramos de que $M^d_{ij}$ se calcula en función de restricciones mínimas ya calculadas previamente y lo hacemos teniendo en cuenta la condición (4). Como cada operación $\odot$ requiere como mucho $p \times q$ operaciones aritméticas, que en el peor de los casos será $R^2$, y se calculan un máximo de $n^2$ operaciones de este tipo, tenemos que el tiempo de ejecución es $O(n^2R^2)$. □

En [9] proponíamos varios métodos para obtener soluciones en tiempo polinomial en una red FTC. Los algoritmos partían de la precondición de minimalidad de la FTCN. Ahora proponemos un algoritmo (fig.2) que, partiendo de una DFTCN relativa $N^d$ con n variables (no necesariamente mínima), determina si existe una solución $\sigma$-posible. En concreto, tomando $\sigma = \alpha$, podemos usar el algoritmo para determinar la $\alpha$-consistencia de una DFTCN relativa. El método propuesto está basado en el algoritmo de Dechter y Pearl [3] para búsqueda de soluciones en problemas de satisfacción de restricciones (CSP) cuyo grafo tiene estructura de árbol.

El algoritmo parte de una asignación arbitraria para la raíz del árbol correspondiente a $X_0$, que por simpli-

1. begin
2. sea $T$ el árbol correspondiente a una DFTCN relativa $N^d$;
3. sea $\langle X_1, \ldots, X_{n-1} \rangle$ una lista con las variables de $N^d$ en preorden respecto a $T$;
4. $s = \{X_0 = 0\}$
5. desde $i = 1$ hasta $n - 1$
6. $F = x_{padre(i)} \odot L^d_{padre(i)i}$;
7. Si existe $x_i$ tal que $\pi_F(x_i) \geq \sigma$ entonces $s = s \cup \{X_i = x_i\}$;
8. en otro caso return ( "No existe solución $\sigma$-posible");
9. return $(s)$;
10. end

Figura 2: Solución para una red relativa

cidad tomamos $X_0 = 0$. Después obtenemos una lista de las variables a partir de un recorrido en preorden del árbol. Cuando seleccionamos una variable $X_i$ tenemos que asignarle una valor consistente con las asignaciones previas y esto se hace construyendo un conjunto difuso F en el paso 6. Si no es posible encontrar un valor $x_i$ que nos de un grado de posibilidad $\pi_F(x_i) \geq \sigma$ entonces no existe una solución $\sigma$−posible. Debido a la topología de árbol, cada variable $X_i$ está restringida como mucho por otra variable $X_j$ anterior a ella, por eso, para asignarle un valor a $X_i$ no necesitamos ningún tipo de retroceso que revise las asignaciones previas. Por tanto el orden de ejecución del algoritmo es $O(nR)$.

# 4 Descomposición en árbol de redes DFTC

Como dijimos anteriormente, las redes DFTC en general no son tratables, no podemos obtener la red mínima en tiempo polinomial y aun en el caso de obtenerla segun el teorema 1 en tiempo exponencial, recesitaríamos de todas formas un algoritmo de búsqueda con retroceso para obtener una solución. Sin embargo, en la sección anterior mostramos que ambos problemas son tratables en el caso concreto de que la red DFTC tenga estructura de árbol. Esto motiva la conveniencia de examinar la posibilidad de eliminar redundancias en una DFTC para extraer un posible árbol que represente a una DFTC relativa equivalente a la original.

Primero veremos que para cualquier red DFTC podemos obtener una red DFTC equivalente que sea camino consistente. Dechter et al. [2] afirman que para redes TCSP no siempre es posible obtener una red camino consistente debido a que un algoritmo de camino consistencia puede no terminar en tiempo finito. Sin embargo, para el caso de redes DFTC donde las distribuciones de posibilidad se definen sobre el conjunto de los números enteros, podemos aplicar, por ejemplo, el algoritmo PC-2 de Mackworth [5]. Con este algoritmo revisaríamos todas las ternas $(i, k, j)$ del grafo de restricciones para comprobar si $L^d_{ij} \sqsubseteq L^d_{ik} \odot L^d_{kj}$. En caso afirmativo $L^d_{ij}$ debe ajustarse según la fórmula

$$L^d_{ij} = L^d_{ij} \sqcap L^d_{ik} \odot L^d_{kj} \tag{5}$$

y deberíamos volver a revisar las ternas que se ven afectadas por el cambio en $L^d_{ij}$. El algoritmo termina cuando no se produce ningún cambio en las restricciones disyuntivas y la red resultante por consiguiente es camino consistente. La clave para demostrar que el algoritmo termina está en que cada vez que se aplica la operación de intersección según (5) se reduce la restricción $L^d_{ij}$ en una cantidad entera y por tanto el proceso de revisión de ternas acaba en tiempo finito. Se puede probar que el orden de complejidad de este algoritmo es $O(n^3R^3)$, donde R es el *rango* de la red DFTC.

A partir de ahora, adaptaremos los resultados del trabajo de Meiri et al. [10] sobre descomposición en árbol de CSP's definidos sobre variables con dominios finitos al caso de redes DFTC, donde los dominios son infinitos y las restricciones disyuntivas.

**Definición 9** *Decimos que una red DFTC $N^d$ es descomponible en árbol si existe una red DFTC relativa $T^d$ con el mismo conjunto de variables equivalente a $N^d$.*

Para obtener una posible descomposición en árbol necesitamos un procedimiento que elimine restricciónes disyuntivas en la red original sin que ello suponga una pérdida de información. La forma natural de hacerlo es mediante la eliminación de restricciones redundantes en la red DFTC mínima equivalente a la original.

**Definición 10** *Sea $N^d$ una red DFTC. Un etiquetado $w$ de $N^d$ es una asignación de pesos a los arcos de su grafo de restricciones $G = (V, E)$. Un etiquetado triangular es un etiquetado $w$ tal que para cada terna o triangulo $t = (e_1, e_2, e_3)$, donde $L^d_{ij}$, $L^d_{ik}$, $L^d_{kj}$ son las restricciones correspondientes a $e_1, e_2, e_3$ respectivamente, se se cumple lo siguiente*

1. *Si $L^d_{ij} = L^d_{ik} \odot L^d_{kj}$ ($e_1$ es redundante en $t$) entonces $w(e_1) \leq w(e_2)$ y $w(e_1) \leq w(e_3)$*

2. *Si $e_1$ es redundante en $t$ y $e_2$ no es redundante en $t$ entonces $w(e_1) < w(e_2)$*

*A las condiciones 1. y 2. las llamaremos restricciones triangulares.*

**Definición 11** *Sea $N^d$ una red DFTC y $G = (V, E)$ su grafo de restricciones. Llamamos grafo de restricciones triangulares asociado a $N^d$, a un grafo dirigido $G_T = (V_T, E_T)$ donde cada nodo representa un arco del grafo de restricciones y un arco $e_1 \rightarrow e_2$ en $E_T$ representa la restricción triangular $w(e_1) < w(e_2)$ o $w(e_1) \leq w(e_2)$ .*

Para una red DFTC cualquiera no siempre es posible encontrar un etiquetado triangular ya que podríamos tener dos triángulos de forma que para un par $(e_1, e_2)$ un triangulo indique $w(e_1) \leq w(e_2)$ y el otro $w(e_1) > w(e_2)$, resultando dos restricciones triangulares en conflicto. Sin embargo si la red DFTC es camino consistente, se puede probar que siempre es posible encontrar un etiquetado triangular. En la figura 3 presentamos un algoritmo que calcula dicho etiquetado triangular en $O(n^3R^2)$. Una vez encontrado el

1. begin
2. sea $N^d$ una red DFTC camino consistente y $G = (V, E)$ su grafo de restricciones;
3. sea $G_T = (V_T, E_T)$ con $V_T = E$ y $E_T = \emptyset$;
4. Para cada $t = (e_i, e_j, e_k)$ en $G$
5. si $e_i$ es redundante en $t$ entonces
6. añadir $e_i \rightarrow e_j$ y $e_i \rightarrow e_k$ a $E_T$;
7. obtener un DAG $G_R = (V_R, E_R)$ a partir de las componentes fuertes de $G_T$;
8. obtener un orden topológico $w$ para $V_R$;
9. para cada componente conexa $C_i$ en $V_R$
10. para cada arco e en $C_i$
11. $w(e) \leftarrow w(C_i)$;
12. return $(w)$
13. end

Figura 3: Etiquetado triangular

etiquetado triangular $w$, podemos usarlo para encontrar una árbol de expansión de coste máximo (MWST) que usaremos para comprobar si existe una descomposición en árbol de una red DFTC. El algoritmo de la

1. begin
2. sea $M^d$ una red DFTC mínima y $G$ su grafo de restricciones;
3. $w \leftarrow$ etiquetado triangular de $M^d$;
4. $T \leftarrow$ MWST de $G$ segun $w$;
5. si $G$ es descomponible por $T$ entonces
6. return $(T)$;
7. en otro caso return ("no existe descomposición en árbol");
8. end

Figura 4: Descomposición en árbol

figura 4 lleva a cabo esta tarea en $O(n^3R^2)$. Para que el algoritmo de descomposición de la figura 4 funcione correctamente es necesario que la red que le damos de entrada sea mínima. Si no, el hecho de que G no sea descomponible por T no nos asegura que la red no sea descomponible. Esto es debido a que el algoritmo no encontraría todas las restricciones redundantes que sólo podrían detectarse en la red mínima.

Aunque la obtención de la red mínima sea no polinomial, si somos capaces de encontrar un árbol de restricciones que la represente, podremos obtener soluciones en la red en tiempo polinomial, según el algoritmo mostrado en la sección anterior. Por tanto, puede ser de interés aplicar el algoritmo de descomposición en árbol. Además, aun en el caso de que el algoritmo nos diga que la red mínima no es descomponible, podemos usar el árbol $T$ obtenido como una buena aproximación a la red mínima.

Otro hecho destacable es que podemos usar el algoritmo de descomposición para detectar si una red DFTC $N^d$ camino consistente es mínima. Puesto que podemos encontrar un etiquetado triangular, segun dijimos antes, el algoritmo de descomposición en principio es aplicable a $N^d$. Si el árbol de expansión $T$ resulta ser una descomposición de $G$ entonces podemos deducir que la red camino consistente es también mínima y $T$ es árbol equivalente a la red mínima. En caso contrario, no podemos decidir nada acerca de la descomponibilidad de $N^d$.

# 5 Conclusiones

En este trabajo hemos ampliado el modelo FTCN permitiendo que tanto las relaciones unarias como las binarias estén determinadas por una unión finita de dis-

tribuciones de posibilidad normalizadas y convexas, con lo cual, las restricciones temporales serán no convexas. Este tipo de restricciones aumenta la expresividad del modelo y pueden ser de utilidad en tareas como planificación. En general, en el modelo ampliado los problemas de determinar la consistencia y encontrar la red mínima no pueden ser resueltos en tiempo polinomial. Sin embargo, hemos demostrado cómo para redes disyuntivas con topología de árbol estos problemas sí son tratables y a partir de aquí hemos propuesto un método de descomposición en árbol de redes disyuntivas basado en la eliminación de restricciones redundantes. El algoritmo propuesto además sirve como test para comprobar la minimalidad de una red disyuntiva camino consistente. Todo esto puede resolver en parte el problema de la ineficiencia de las redes disyuntivas.

# Referencias


[1] S. Barro, R. Marín, A.R Patón, A model and a language for the fuzzy representation and handling of time. Fuzzy Sets and Systems, 61, 1994, 153–175.

[2] R. Dechter, I. Meiri, J. Pearl, Temporal constraint networks. Artif. Intell., 49, 1991, 61-95.

[3] R. Dechter, J. Pearl, Network-based heuristics for constraint satisfaction problems. Artif. Intell., 34 (1), 1987, 1-38.

[4] D. Dubois, J. Prade, Processing fuzzy temporal knowledge. IEEE Trans. on Systems, Man and Cybernetics, 19, 4, 1988, 729-744.

[5] A. K. Mackworth and E. C. Freuder, The complexity of some polynomial network consistency algorithms for constraint satisfaction problems. Artif. Intell., 25 (1), 1985, 65-74.

[6] R. Marín, S. Barro, A. Bosch, J. Mira, Modeling time representation from a fuzzy perspective. Proc. of EUROCAST'93, 1993, 81-84.

[7] R. Marín, S. Barro, A. Bosch, I. Navarrete, M. Cárdenas, A. Bugarín, J. Mira, Un modelo para la representación de información temporal borrosa. Actas del Tercer Congreso Español sobre Tecnologías y Lógica Fuzzy, 1993, 209-216.

[8] R. Marín, S. Barro, A. Bosch, J. Mira, Modeling time representation from a fuzzy perspective. Cybernetics and Systems 25, 2, 1994, 207-215.

[9] R. Marín, M. Cárdenas, M. Balsa, J. Sánchez, Obtaining solutions in fuzzy constraint networks. International Journal of Approximate Reasoning (aceptado y pendiente de publicación).

[10] I. Meiri, R. Dechter, J. Pearl, Uncovering trees in constraint networks. Artif. Intell., 86, 1996, 245-267.

[11] U. Montanari, Networks of constraints: fundamental properties and applications to picture processing. Information Science, 7, 1974, 95-132.

[12] M. Vilain and H.Kautz, Constraint propagation algorithms for temporal reasoning. In Proc. AAAI-86, pages 377-382, 1986.

# Restricciones de diferente importancia en problemas de satisfacción de restricciones difusas

**Vicenç Torra i Reventós**
Departament d'Enginyeria Informàtica (ETSE), Universitat Rovira i Virgili
Carretera de Salou s/n, E-43006 Tarragona (Spain)
vtorra@etse.urv.es

## Resumen

Muchas aplicaciones, como las de planificación temporal o configuración, pueden formularse como problemas de satisfacción de restricciones. En estos problemas, un conjunto de variables debe fijarse a un valor que satisfaga un conjunto de restricciones. Los problemas de satisfacción de restricciones clásicos se definen unicamente a partir de restricciones nítidas (*crisp contraints*). Sin embargo, como las restricciones booleanas son normalmente demasiado estrictas para expresar las necesidades que tiene que cumplir una solución, se han introducido las restricciones difusas.

Cuando se consideran restricciones con satisfación parcial, el razonamiento humano realiza normalmente algún tipo de compensación entre las diferentes alternativas. Por tanto, en el caso de utilizar restricciones difusas son necesarios operadores diferentes que las t-normas. Además, no todas las restricciones son consideradas con igual importancia. En este trabajo se muestra que el operador WOWA permite considerar ambos aspectos: compensación entre restricciones y restricciones de diferente importancia.

**Palabras clave**: Satisfacción de restricciones, operadores de agregación.

## 1 INTRODUCCIÓN

Muchas de las aplicaciones existentes en la actualidad (por ejemplo en planificación temporal [6], configuración [5], [7]) pueden formularse como problemas de satisfacción de restricciones. En esta clase de problemas una solución consiste en asignar a cada variable un valor de un conjunto de valores. Además la solución debe satisfacer un conjunto de restricciones.

Como [6] afirma, estas restricciones pueden verse como objetos matemáticos usados para hacer explícito los aspectos lógicos subyacentes a un problema. Por ejemplo, en la configuración de un sistema de ayuda a la compilación en entornos paralelos [7] las restricciones expresan la incompatibilidad entre las diferentes estrategias de paralelización. Por tanto, estas estrategias corresponden a un conocimiento básico de las soluciones del problema.

En los problemas de satisfacción de restricciones clásicos, las soluciones solo tienen restricciones nítidas (restricciones booleanas). Por tanto, una asignación de un conjunto de valores a un conjunto de variables solo puede ser solución (si los valores asignados satisfacen las restricciones) o ser no-solución (si alguna de las restricciones es violada). En este caso, las restricciones se pueden representar mediante relaciones nítidas $R(S)=1$ si la solución S satisface la restricción R y $R(S)=0$ si no se satisface la restricción.

Sin embargo, algunas veces las restricciones booleanas son demasiado estrictas en relación al razonamiento humano. No todas las soluciones son igualmente buenas, y algunas no soluciones pueden ser casi-soluciones. Para resolver este problema, se pueden introducir las restricciones difusas. Estas se representan mediante relaciones difusas de forma que cualquier solución puede evaluarse a partir de una restricción difusa R y $R(S) \in [0,1]$.

Como se ha dicho, cuando se consideran las restricciones en un sistema clásico de CSP, todas las relaciones deben satisfacerse para una asignación de valores a variables. En el caso de disponer de restricciones difusas, esto puede modelizarse mediante t-normas. Sin embargo, las personas acostumbran a compensar las diferentes relaciones. Esto es, a veces se escoge una solución que satisface en menor grado una restricción cuando otra es satisfecha con mayor grado. Por tanto, para obtener una buena solución global, se realiza un compromiso entre varias restricciones. Para modelizar este compromiso, es adecuado el operador OWA [12],[13],[14].

Además, de forma frecuente, no todas las restricciones tienen la misma importancia. Por tanto, es razonable utilizar una función que considere la importancia de las restricciones. En [6] se revisa un conjunto de funciones que consideran una evaluación de la importancia.

En este trabajo se muestra como puede ser aplicado el operador WOWA [9], [10] en los problemas de satisfacción de restricciones difusas. En particular, se muestra como modelizar los dos aspectos mencionados anteriormente: la compensación entre restricciones y la importancia de las mismas. Además se muestra la aplicación al ejemplo de [6].

## 2 COMPENSACIÓN Y IMPORTANCIA

Consideremos el ejemplo de [6] adaptado de [3].

**Ejemplo 1.** [6] Los profesores A y B deben impartir un curso de acuerdo con las siguientes restricciones.

$C_1$. El profesor A dará la parte teórica del curso y puede impartir entre 2 y 4 sesiones, siendo 3 el número ideal.

$C_2$. El profesor B dará la parte práctica del curso. Puede impartir alrededor de 2 sesiones siendo 1 y 3 medianamente acceptable.

$C_3$. Idealmente el curso debe estar compuesto de 6 sesiones, pero 5 o 7 son igualmente acceptables.

$C_4$. Los dos profesores A y B deberían impartir aproximadamente el mismo número de sesiones (idealmente el mismo número, pero una diferencia de 1 o 2 sesiones es medianamente acceptable).

Definiendo un vector de variables $\mathbf{x}=(x_1, x_2)$ donde $x_1$ es el número de sesiones teóricas y $x_2$ es el número de sesiones prácticas podemos reescribir estas restricciones como relaciones sobre el vector $\mathbf{x}$. De este modo cada restricción $C_i$ se traduce a una relación $R_i(\mathbf{x})$ donde $R_i$ captura el significado de $C_i$. Cada relación puede ser nítida ($R_i(\mathbf{x}) \in \{0,1\}$) o difusa ($R_i(\mathbf{x}) \in [0,1]$) de acuerdo con el significado de la restricción correspondiente.

En el ejemplo dado anteriormente, cada restricción puede formalizarse mediante una relación difusa. Por ejemplo, $C_1$ puede traducirse a una relación difusa que se satisface completamente cuando $x_1=3$ (esto es, $R_1(\mathbf{x})=1$ si $x_1=3$), que disminuye su satisfacción cuando $\mathbf{x}$ incrementa y mayor que 3 (esto es, $R_1(\mathbf{x_1})<R_2(\mathbf{x_2})$ si $x_{11}>x_{21}>3$) y que también decrementa su satisfacción cuando x decrementa y es menor que 3 (esto es, $R_1(\mathbf{x_1})<R_2(\mathbf{x_2})$ si $x_{11}<x_{21}<3$). Definiendo un conjunto difuso sobre $x_1$, por ejemplo $\mu_{About_3}$, con las propiedades descritas anteriormente, es posible formalizar la restricción $C_1$ como:

$$R_1(\mathbf{x})=\mu_{About_3}(x_1)$$

De forma parecida, definiendo los conjuntos difusos $\mu_{About_2}$, $\mu_{About_6}$, $\mu_{About_0}$ es posible formalizar las propiedades $C_2$, $C_3$ y $C_4$. De este modo, obtenemos las relaciones siguientes que corresponden a las restricciones $C_1$, $C_2$, $C_3$ y $C_4$:

$$R_1(\mathbf{x}) = \mu_{About_3}(x_1)$$

$$R_2(\mathbf{x}) = \mu_{About_2}(x_2)$$

$$R_3(\mathbf{x}) = \mu_{About_6}(x_1+x_2)$$

$$R_4(\mathbf{x}) = \mu_{About_0}(|x_1-x_2|)$$

Para completar esta formalización, se requiere una definición de los conjuntos difusos mencionados anteriormente. Utilizamos en este ejemplo la definición de [6] que se reproduce en la figura 1.

Dado un conjunto de restricciones, es posible evaluar una solución en relación a cada restricción. Esto se consigue aplicando la relación correspondiente a la solución ($R_i(\mathbf{x})$). Cuando las restricciones son nítidas, la evaluación es nítida: una solución o bien *satisface* o *no satisface* la restricción. Cuando las restricciones son difusas, la evaluación es un valor difuso. Consideremos ahora 6 posibles soluciones del ejemplo dado anteriormente:

$$\mathbf{u}_1=(2,3),\ \mathbf{u}_2=(3,2),\ \mathbf{u}_3=(3,3),\ \mathbf{u}_4=(4,2),$$

$$\mathbf{u}_5=(4,3),\ \mathbf{u}_6=(3.5,2.5)$$

La evaluación de las cuatro restricciones dadas anteriormente aparece en la tabla 1. Es necesario subrayar que ninguna de las soluciones consideradas satisface completamente todas las restricciones (de hecho, no existe ninguna solución que las satisfaga todas cuando los conjuntos difusos utilizados son los de la figura 1).

| Soluciones posibles | $R_1(\mathbf{u})$ | $R_2(\mathbf{u})$ | $R_3(\mathbf{u})$ | $R_4(\mathbf{u})$ | $\mathbb{C}$=Min | $\mathbb{C}$=Prod | $\mathbb{C}$=$OWA_2$ | $\mathbb{C}$=$OWA_3$ | $\mathbb{C}$=$OWA_*$ |
|---|---|---|---|---|---|---|---|---|---|
| $\mathbf{u}_1$=(2,3) | .5 | .5 | .5 | .5 | .5 | .0625 | .5 | .5 | .5 |
| $\mathbf{u}_2$=(3,2) | 1 | 1 | .5 | .5 | .5 | .25 | 1 | .6333 | .65 |
| $\mathbf{u}_3$=(3,3) | 1 | .5 | 1 | 1 | .5 | .5 | 1 | 1 | .8 |
| $\mathbf{u}_4$=(4,2) | .5 | 1 | 1 | .5 | .5 | .25 | 1 | .6333 | .65 |
| $\mathbf{u}_5$=(4,3) | .5 | .5 | .5 | .5 | .5 | .0625 | .5 | .5 | .5 |
| $\mathbf{u}_6$=(3.5,2.5) | .75 | .75 | 1 | .5 | .5 | .28125 | .875 | .6333 | .675 |

Tabla 1. Algunas funciones de agregación. $OWA_2$ corresponde a un OWA con pesos w=(.5, .5, 0, 0), $OWA_3$ a los pesos w=(1/3, 1/3, 1/3, 0) y $OWA_*$ a w=(.1, .2, .3, .4).

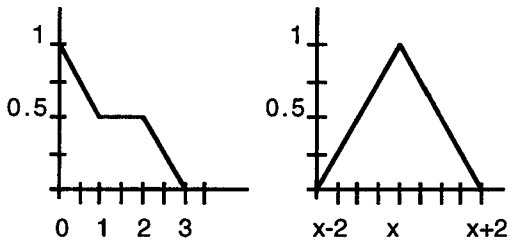



Figura 1. Funciones de pertenencia: About_0 (izquierda), About_x (derecha)

Para seleccionar la mejor solución no es suficiente tener una medida de satisfacción para cada restricción, es necesario tener una medida general que englobe a todas las restricciones. Si unicamente consideramos la satisfacción de las restricciones es posible ordenar todas las soluciones posibles: una solución $\mathbf{u}_1$ es mejor que otra solución $\mathbf{u}_2$ si para todas las relaciones $R_i$ la solución $\mathbf{u}_1$ es mejor que la de $\mathbf{u}_2$. Esto es, $\mathbf{u}_1 \geq \mathbf{u}_2$ si y solo si para todas las restricciones $C_i$, $R_i(\mathbf{u}_1) \geq R_i(\mathbf{u}_2)$. Sin embargo, esta ordenación $\geq$ es un orden parcial porque para algunos pares de soluciones $\mathbf{u}_1$, $\mathbf{u}_2$, ni se satisface $\mathbf{u}_1 \geq \mathbf{u}_2$ ni $\mathbf{u}_1 \leq \mathbf{u}_2$. Si, encambio, consideramos una medida general R la relación $\mathbf{u}_1 \geq \mathbf{u}_2$ puede definirse como cierta si y solo si $R(\mathbf{u}_1) \geq R(\mathbf{u}_2)$. Esto es un orden total.

La medida general $R(\mathbf{u})$ se define como una agregación de las medidas de cada restricción. En general, cuando $R_i(\mathbf{u})$ corresponde a la medida de satisfacción de la restricción $C_i$ de la solución $\mathbf{u}$, podemos definir la medida general $R(\mathbf{u})$ como:

$$R(\mathbf{u}) = \mathbb{C}(R_1(\mathbf{u}), R_2(\mathbf{u}), R_3(\mathbf{u}), R_4(\mathbf{u})) \quad (1)$$

donde $\mathbb{C}$ es una función de agregación definida sobre el rango de $R_i$.

De hecho, es posible tener una definición más general:

$$R(\mathbf{u}) = \mathbb{C}_R(R_1, R_2, R_3, R_4)\,(\mathbf{u}) \quad (2)$$

donde $\mathbb{C}_R$ es una función de agregación sobre relaciones. Sin embargo, por razones de simplicidad consideramos (1) en lugar de (2). La condición (1) se obtiene de (2) cuando se exige la condición de independencia de las alternativas irrelevantes [1], [8]:

$$\mathbb{C}_R(R_1, R_2, R_3, R_4)\,(\mathbf{u}) = \mathbb{C}(R_1(\mathbf{u}), R_2(\mathbf{u}), R_3(\mathbf{u}), R_4(\mathbf{u}))$$

Esta condición significa, en nuestro caso, que la medida global de satisfacción de una solución depende unicamente de las medidas parciales de esa solución, pero que no influyen otras soluciones.

Cuando unicamente se consideran restricciones nítidas, $R(\mathbf{u})$ se define como cierto (la posible solución $\mathbf{u}$ es una buena solución) si y solo si todas las $R_i(\mathbf{u})$ son ciertas. En este caso tenemos que $R_i(\mathbf{u})$=cierto si la restricción $C_i$ se satisface. Para modelizar esta condición, la función de agregación $\mathbb{C}$ se define como la conyunción de todas las $R_i(\mathbf{u})$.

Cuando se consideran las restricciones difusas, la extensión natural de la conjunción es definir $\mathbb{C}$ como una t-norma. Por ejemplo, $\mathbb{C}$ puede definirse como el minimo o el producto de las $R_i(\mathbf{u})$. Los resultados obtenidos mediante estas dos t-normas para el ejemplo anterior se muestran en la tabla 2. Cuando se quiere usar una t-norma como función de agregación es importante subrayar que la única t-norma que satisface la idempotencia (esto es, $\mathbb{C}(v,v,...,v)=v$ para todo v en [0,1]) es el minimo. Por ejemplo, en la tabla 2 podemos observar que $\mathbb{C}(.5,.5,.5,.5)=.0625$ cuando $\mathbb{C}$ es el producto (soluciones $\mathbf{u}_1$ y $\mathbf{u}_5$).

Además de la falta de idempotencia en la mayoria de t-normas, otro inconveniente de usar las t-normas como agregación es que no permiten compensar las diferentes restricciones. Como [6] y [15] apuntan, se ha demostrado

que las personas son capaces de compensar los objetivos en conflicto. Este puede ser el caso del ejemplo anterior. Como no todas las restricciones pueden satisfacerse a la vez, podemos compensar en la solución $\mathbf{u}_3$=(3,3) el hecho que el profesor B imparte 3 sesiones con el hecho que las restricciones $C_2$ y $C_4$ se satisfacen completamente.

Un operador que permite la compensación es el operador OWA. Definimos a continuación este operador para la agregación de n restricciones:

**Definición 1** [12],[13]. Sea **w** un vector de pesos de dimensión n ($\mathbf{w}= [w_1\ w_2\ ....\ w_n]$) tal que

$$w_i \in [0,1] \quad y \quad \Sigma_i\ w_i = 1$$

En este caso, la función $f_{owa}: \mathbb{R}^n \rightarrow \mathbb{R}$ es un **Ordered Weighted Averaging (OWA) operator** de dimensión n si:

$$f_{owa}(a_1,...,a_n) = \Sigma_i\ w_i\ a_{\sigma(i)}$$

donde $\{\sigma(1),...,\sigma(n)\}$ es una permutación de $\{1,..,n\}$ tal que $a_{\sigma(i-1)} \geq a_{\sigma(i)}$ para todo i=2, ..., n. (esto es, $a_{\sigma(i)}$ es el elemento i-ésimo más grande de la colección $a_1,..., a_n$).

El operador OWA permite la compensación a causa de la ordenación introducida en la permutación $\sigma$. Cuando los pesos asociados a los valores mayores son más grandes que los pesos asociados a los valores pequeños, tenemos mucha compensación. La compensación máxima se consigue cuando los pesos $w_i$ se definen como sigue: $w_1$=1 y $w_i$=0 para todo $i \neq 1$. En este caso, la medida de satisfacción de una solución corresponde a la medida de satisfacción de la restricción que se satisface mejor.

Cuando es necesario, el OWA permite también no compensar. Cuando los pesos asociados a los valores pequeños son mayores que los pesos asociados a los valores grandes, no tenemos compensación. El caso de compensación mínima se consigue cuando $w_n$=1 y $w_i$=0 para todo $i \neq n$. En este caso, la medida de satisfacción de una solución corresponde a la medida de satisfacción de la restricción que se satisface menos.

En la tabla 1 se muestran tres casos con el operador OWA. $OWA_2$ corresponde al OWA con un vector de pesos **w**=(1/2, 1/2, 0 ,0) y $OWA_3$ corresponde a un vector de pesos de la forma **w**=(1/3, 1/3, 1/3, 0). Se puede observar que la compensación del $OWA_2$ es mayor que la del $OWA_3$, y como las soluciones $\mathbf{u}_2$, $\mathbf{u}_4$ y $\mathbf{u}_6$ son mejor evaluadas en el $OWA_2$ que en el $OWA_3$ (porque en el primer caso se consideran unicamente los 2 valores más grandes). La tabla 1 también incluye (columna $OWA_*$) los resultados con el vector de pesos **w**=(.1, .2, .3, .4) que Slany [6] incluye en su trabajo. Este vector corresponde a una compensación parcial porque los pesos asociados a los valores menores son mayores que los asociados a los valores mayores.

En la tabla 1 puede observarse que dadas dos posibles soluciones $\mathbf{u}_i$ y $\mathbf{u}_j$ el OWA no puede compensar de forma suficiente para llegar a obtener $OWA(\mathbf{u}_i) > OWA(\mathbf{u}_j)$, cuando tenemos $\mathbb{C}(\mathbf{u}_i) < \mathbb{C}(\mathbf{u}_j)$ utilizando el producto. Sin embargo, esta compensación si que es posible en general. Damos un ejemplo de ello en la tabla 2. En este caso, se consideran dos soluciones posibles $\mathbf{v}_1$ y $\mathbf{v}_2$ en relación a cuatro restricciones $R_1$, $R_2$, $R_3$ y $R_4$. La satisfacción de cada restricción se muestra en la tabla 2. Puede observarse que el operador OWA con los pesos **w**=(.4,.3,.2,.1) satisface que $\mathbb{C}(\mathbf{v}_1) > \mathbb{C}(\mathbf{v}_2)$ mientras que para el mínimo y el producto $\mathbb{C}(\mathbf{v}_1) < \mathbb{C}(\mathbf{v}_2)$. Notese, sin embargo, que con un vector de pesos **w**=(0,0,0,1) restaura la relación $\mathbb{C}(\mathbf{v}_1) < \mathbb{C}(\mathbf{v}_2)$ puesto que en este caso el OWA es igual al mínimo.

Una de las propiedades que satisface el operador OWA (véase [4],[13],[10] para una revisión de las propiedades) es que el resultado siempre permanece entre el mínimo y el máximo de los valores a combinar. Esto significa en nuestro caso:

$$\text{Min}\ \{R_1(\mathbf{u}), ..., R_n(\mathbf{u})\} \leq \mathbb{C}(R_1(\mathbf{u}), ..., R_n(\mathbf{u})) \leq$$

$$\text{Max}\ \{R_1(\mathbf{u}), ..., R_n(\mathbf{u})\}$$

El mínimo se obtiene para **w**=(0,0,...,0,1) y el máximo para **w**=(1,0,...,0).

| Soluciones posibles | $R_1(\mathbf{u})$ | $R_2(\mathbf{u})$ | $R_3(\mathbf{u})$ | $R_4(\mathbf{u})$ | $\mathbb{C}$=Min | $\mathbb{C}$=Prod | $\mathbb{C}=OWA_2$ | $\mathbb{C}=OWA_3$ | $\mathbb{C}=OWA^*$ |
|---|---|---|---|---|---|---|---|---|---|
| $\mathbf{v}_1$ | .8 | .6 | .4 | .2 | .2 | .0384 | .7 | .6 | .62 |
| $\mathbf{v}_2$ | .5 | .5 | .5 | .5 | .5 | .25 | .5 | .5 | .5 |

Tabla 2. $\mathbb{C}=OWA^*$ con **w**=(.4,.3,.2,.1)

Otra de las propiedades que satisfacen los operadores OWA es la commutatividad. El operador, como también las t-normas, es invariante en relación a las permutaciones de los parámetros de entrada.

Esta propiedad no es adecuada cuando no todas las restricciones tienen la misma importancia. Como señala [6] "*an intuitive requirement is that as a constraint becomes more important it should play a more significant role in determining the overall decision function*".

Para considerar la importancia de una restricciones, la función de agregación en (1) debe considerar los pesos para cada restricción. En [6] se consideran varias funciones de agregación junto a sus propiedades. La función más simple de todas ellas es la media ponderada. Esto es:

$$\mathbb{C}(R_1(\mathbf{x}), \ldots, R_n(\mathbf{x})) = \sum_{i=1}^{n} p_i R_i(\mathbf{x})$$

con el vector de pesos $\mathbf{p}=(p_1,...,p_n)$ definido de forma que $p_i \geq 0$ y $\Sigma_i p_i = 1$.

En este caso, la importancia de la restricción $C_i$ es dada por $p_i$. Por ejemplo, cuando una restricción $C_i$ no tiene importancia, esto es, $p_i=0$, su satisfacción no afecta al resultado final.

Utilizando la media aritmética como operador de agregación con el vector de pesos **p** definido como (1/16, 9/16, 5/16, 1/16) obtenemos la evaluación mostrada en la tabla 3. En este ejemplo podemos ver que la solución $\mathbf{u}_4$ es ahora la mejor evaluada porque las dos restricciones más importantes se satisfacen completamente ($R_2(\mathbf{u}_4)=R_3(\mathbf{u}_4)=1$). En su lugar, la solución $\mathbf{u}_3$ que era la mejor cuando el operador de agregación utilizado era el $OWA_*$ se encuentra ahora en la tercera posición de la nueva ordenación porque la única restricción que no se satisface ($C_2$) es la que tiene una mayor importancia.

| Soluciones posibles | $\mathbb{C}$=Min | $\mathbb{C}$=Prod | $\mathbb{C}$=$OWA_2$ | $\mathbb{C}$=$OWA_3$ | $\mathbb{C}$=$OWA_*$ | $\mathbb{C}$=W.M. | $\mathbb{C}$=WOWA |
|---|---|---|---|---|---|---|---|
| $\mathbf{u}_1$=(2,3) | .5 | .0625 | .5 | .5 | .5 | .5 | .5 |
| $\mathbf{u}_2$=(3,2) | .5 | .25 | 1 | .6333 | .65 | .8125 | .719 |
| $\mathbf{u}_3$=(3,3) | .5 | .5 | 1 | 1 | .8 | .719 | .621 |
| $\mathbf{u}_4$=(4,2) | .5 | .25 | 1 | .6333 | .65 | .9375 | .892 |
| $\mathbf{u}_5$=(4,3) | .5 | .0625 | .5 | .5 | .5 | .5 | .5 |
| $\mathbf{u}_6$=(3.5,2.5) | .5 | .28125 | .875 | .6333 | .675 | .8125 | .757 |

Tabla 3. Algunas funciones de agregación. $OWA_2$ corresponde a un OWA con pesos **w**=(.5, .5, 0, 0), $OWA_3$ a unos pesos **w**=(1/3, 1/3, 1/3, 0) y $OWA_*$ a **w**=(.1, .2, .3, .4), WM a **p**=(1/16, 9/16, 5/16, 1/16), y WOWA a **p**=(1/16, 9/16, 5/16, 1/16) y **w**=(.1, .2, .3, .4).

## 3 CONSIDERANDO COMPENSACIÓN Y IMPORTANCIA

Los operadores considerados en la sección previa permiten considerar en la evaluación de una solución los dos aspectos de importancia de las restricciones y la necesidad de compensación entre restricciones. Sin embargo, ambos aspectos no pueden considerarse a la vez en las funciones presentadas. El operador WOWA introducido en [10] permite considerar los dos.

El operador WOWA es una generalización de las dos funciones de agregación: media aritmética y operador OWA. Permite agregar un conjunto de valores utilizando dos vectores de pesos. Uno corresponde al vector en la media aritmética, y por tanto mide la importancia de las restricciones. El otro corresponde a los pesos en el operador OWA, y por tanto los pesos corresponden a la compensación. El operador WOWA con ambos pesos permite considerar la importancia de las restricciones y la compensación entre ellas.

**Definición 2.** [10] Sean **p** y **w** dos vectores de pesos de dimensión n ($\mathbf{p}=[p_1 .... p_n]$, $\mathbf{w}=[w_1, ... w_n]$) tales que:

$$\text{i) } p_i \in [0,1] \quad \text{y} \quad \Sigma_i p_i = 1$$

$$\text{ii) } w_i \in [0,1] \quad \text{y} \quad \Sigma_i w_i = 1$$

En este caso, una función $f_{wowa}:\mathbb{R}^n \rightarrow \mathbb{R}$ es un operador **Weighted Ordered Weighted Averaging (WOWA)** de dimensión n si

$$f_{wowa}(a_1,...,a_n) = \Sigma_i \omega_i a_{\sigma(i)}$$

donde $\{\sigma(1),...,\sigma(n)\}$ es una permutación de $\{1,..,n\}$ tal que $a_{\sigma(i-1)} \geq a_{\sigma(i)}$ para todo i=2, ..., n. (esto es, $a_{\sigma(i)}$ es el

i-ésimo elemento en la colección $a_1,..., a_n$), y cada peso $\omega_i$ se define como

$$\omega_i = W^*(\Sigma_{j\le i} p_{\sigma(j)}) - W^*(\Sigma_{j<i} p_{\sigma(j)})$$

con $W^*$ siendo una función monótona creciente que interpola los puntos (i/n, $\Sigma_{j\le i} w_j$) y el punto (0,0). Además es necesario que $W^*$ sea una linea recta en el caso de que los puntos puedan ser interpolados de este modo. Esta condición es exigida para que la función de agregación sea idempotente [10]. En adelante, $\omega$ representa el conjunto de pesos $\{\omega_i\}$, esto es, $\omega=[\omega_1 \ldots \omega_n]$.

Para calcular la medida general de satisfacción con el operador WOWA es necesario determinar los dos conjuntos de pesos **p** y **w**. Además, debe definirse un método para construir la función de interpolación $W^*$. En [11] se describe la adaptación del método de interpolación definido en [2] de forma que la función $W^*$ satisfaga las condiciones requeridas por el operador WOWA.

En la tabla 3, damos los resultados para el ejemplo 1 cuando utilizamos el operador WOWA como función de agregación. Como vector de pesos **p** se ha utilizado el vector de la media aritmética y como vector **w** el de pesos en el $OWA_*$. Además el método de interpolación es el descrito en [11]. Debe subrayarse que al considerar ambos pesos, el nuevo orden es diferente a los anteriores. Las tres ordenaciones son las siguientes:

| | |
|---|---|
| $OWA^*$ | $\mathbf{u}_3 \ge \mathbf{u}_6 \ge \{\mathbf{u}_2,\mathbf{u}_4\} \ge \{\mathbf{u}_1,\mathbf{u}_5\}$ |
| WM | $\mathbf{u}_4 \ge \{\mathbf{u}_2,\mathbf{u}_6\} \ge \mathbf{u}_3 \ge \{\mathbf{u}_1,\mathbf{u}_5\}$ |
| WOWA | $\mathbf{u}_4 \ge \mathbf{u}_6 \ge \mathbf{u}_2 \ge \mathbf{u}_3 \ge \{\mathbf{u}_1,\mathbf{u}_5\}$ |

En el operador WOWA la mejor solución es $\mathbf{u}_4$ porque las restricciones que se satisfacen menos tienen poca importancia (1/16 cada una). Además, aunque la solución $\mathbf{u}_2$ tiene la misma evaluación que $\mathbf{u}_6$ cuando se usa el OWA, no es así en el caso del WOWA. Esto es así porque $\mathbf{u}_2$ satisface $C_1$ en lugar de $C_3$ y el peso del segundo es mayor que el del primero.

Además, ninguno de los métodos permiten la discriminación entre las soluciones $\mathbf{u}_1$ y $\mathbf{u}_5$ puesto que todas satisfacen del mismo modo todas las restricciones.

# 4 CONCLUSIONES

En este trabajo se ha mostrado la adecuación del operador WOWA en los problemas de satisfacción de restricciones.Hemos visto que el operador de agregación permite considerar restricciones de diferente importancia y compensación entre ellos.

# 5 AGRADECIMIENTOS

Se agradece la ayuda de la Comunidad Europea con el contrato VIM: CHRX-CT93-0401 y la CICYT, proyecto SMASH: TIC-96-1038-C04-04.

# 6 REFERENCIAS

[1] Arrow, K.J., (1951), Social choice and individual values, Wiley, New York, (2nd edition 1963).

[2] Chen, J. E., Otto, K. N., (1995), Constructing membership functions using interpolation and measurement theory, *Fuzzy Sets and Systems*, 73:3 313-327.

[3] Fargier, H., Lang, J., Schiex, T., (1993), Selecting preferred solutions in fuzzy constraint satisfaction problems, in: H.-J. Zimmwemann, ed., EUFIT'93, *First European Cong. on Fuzzy and Intelligent Technologies*, Aachen, Germany (Augustinus Buchhandlung, September 1993) 1128-1134.

[4] Fodor, J., Marichal, J.-L., Roubens, M., Characterization of the Ordered Weighted Averaging Operators, *IEEE Trans. on Fuzzy Systems*, 3:2, 236-240 (1995).

[5] Sebastian, H.-J., (1996), Intelligent Systems for Configuration Problems, 89-232, in H.-J. Sebastian, E. K. Antonsson, *Fuzzy Sets in Engineering Design and Configuration*, Kluwer Academic Publishers, USA.

[6] Slany, W., (1996), Scheduling as a fuzzy multiple criteria optimization problem, *Fuzzy Sets and Systems*, 78 197-222.

[7] Sodan, A., Torra, V., (1997), Configuration Decisions for Mapping by Fuzzy Inference, Technical Report, GMD-FIRST, Berlin.

[8] Torra, V., (1995), Combining Fuzzy Sets: The Geometric Consensus Function Family, *Fuzzy Sets and Systems*, 74 335-342.

[9] Torra, V., (1996), Weighted OWA operators for synthesis of information, actas del *Fifth IEEE International Conference on Fuzzy Systems (IEEE-FUZZ'96)*, 966-971, New Orleans, USA, Setiembre 8-11.

[10] Torra, V., (1997), The Weighted OWA operator, *International Journal of Intelligent Systems*, 12 153-166.

[11] Torra, V., (1997b), The WOWA operator and the interpolation function $W^*$: Chen and Otto's interpolation method revisited, Submitted.

[12] Yager, R.R., (1988), On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Trans. on SMC*, 18 183-190.

[13] Yager, R.R., (1993), Families of OWA operators", *Fuzzy Sets and Systems*, 59 125-148.

[14] Yager, R.R., (1996), Quantifier Guided Aggregation Using OWA operators, *International Journal of Intelligent Systems*, 11 49-73.

[15] Zimmermann, H,-J., (1991), *Fuzzy Set Theory - and Its Applications*, Kluwer, Dordrecht, 2nd revised edition.

# PRINCIPIOS MUY GENERALES PARA LA EDUCACIÓN EN LÓGICA FUZZY. Algunas reflexiones personales

**Alejandro Sobrino**
Dpto. de Lógica y Filosofía de la Ciencia. Campus Universitario, s. n. 15706 Santiago de Compostela
e-mail:lflgalex@usc.es

## Resumen

En este trabajo se ofrecen algunas reflexiones personales acerca de la conveniencia de educar en lógica fuzzy, sugiriendo, a grandes rasgos, algunas propuestas de cómo esta actividad se podría incardinar en el sistema educativo español, desde la infancia del individuo hasta el período adulto. Mas allá de algunas propuestas concretas, el trabajo pretende propiciar una discusión acerca de este tema entre todos aquellos que, desde alguna perspectiva, nos dedicamos a estudiar y desarrollar la lógica fuzzy.

**Palabras clave**: Educación en lógica fuzzy, conjuntos fuzzy y educación matemática.

## 1 GENERALIDADES

Las disciplinas sujeto de enseñanza se van conformando como tales a lo largo del tiempo. Existen ámbitos bien asentados desde la antigüedad, como la Matemática, la Filosofía, la Lingüística o la Física, y otros más advenedizos, como la Psicología o la Lógica. La Lógica, aún con relativamente pocos años de autonomía, posee parcelas de conocimiento con cierta tradición y otras más novedosas. Un desarrollo reciente de la Lógica es la lógica fuzzy, que desde el año 1965 ha alcanzado cada vez mayor credibilidad y difusión. Institucionalmente, la Lógica formal se ha enseñado en España en las Facultades de Filosofía, Matemáticas y, más recientemente, en la de Informática.

Aun cuando la lógica fuzzy tiene un bautismo nominal reciente (el rótulo 'lógica fuzzy' lo utilizó por primera vez Zadeh en un trabajo escrito en 1974), ésta se relaciona, no sólo con ámbitos disciplinares de tardía aparición, sino también con saberes tradicionales. En efecto, la lógica fuzzy no sólo tiene que ver con el control de sistemas, por más que este fuera el marco en el que la concibió Zadeh. Aristóteles, p. ej., ya avisó de la posible dimensión filosófica del estudio formal de la vaguedad al vincular los futuros contingentes, enunciados de significado indeterminado, con reflexiones sobre el libre albedrío. Son habituales también las borrosificaciones de diferentes parcelas de la Matemática que gozan de gran tradición. Pero indudablemente, la relación más estrecha de la lógica fuzzy es la que mantiene con la Informática avanzada. Por tanto, es una temática que tiene que ver, al menos, con dos disciplinas cuya enseñanza está institucionalizada desde antiguo en España; la Matemática y la Filosofía, asi como con otra más reciente y de creciente implantación: la Informática.

Pero la lógica fuzzy no se ha quedado en una mera teoría, sino que ha dado lugar a productos manufacturados de utilización industrial y doméstica. Por tanto, y este no es un caso demasiado frecuente en la historia de la ciencia, es una teoría que ha logrado su tecnología. Desde este punto de vista, constituye no sólo una saber científico, sino también una materia aplicada, con la que se hacen cosas, lo que le da una dimensión utilitaria.

La lógica fuzzy tiene que ver, como cualquier lógica, con la argumentación. Pero a diferencia de otras lógicas, como la clásica, que se fija preferentemente en la argumentación probatoria de la Matemática o de la Física, la lógica fuzzy atiende al razonamiento de sentido común. Mientras la argumentación clásica es una argumentación de contenidos cognitivos propios de etapas maduras, la argumentación ordinaria recorre la vida de un individuo desde su infancia.

Así, un niño de 3 años seguramente razona con dificultad a propósito de un enunciado elemental de la geometría, que involucra razonamiento preciso, pero es capaz de decidir razonablemente y sin aparente esfuerzo (usando una argumentación más bien implícita que explícita) sobre si quiere o no más comida o sobre si quiere o no muchas más galletas, siendo 'más' y 'mucho más' cuantificadores aproximados.

En la adolescencia se encontrará con los primeros estudios cerrados de un montón de disciplinas y tendrá, para afrontar con éxito esta tarea, que desarrollar aptitudes que involucran gestión de imprecisión: heurísticas de aprendizaje y recuerdo de

datos (reglas mnemotécnicas, p. ej.); estrategias para correlacionar informaciones diferentes; razonamiento por analogía para encontrar soluciones a problemas en virtud de casos similares ya resueltos, etc. Todas estas tareas comportan razonamiento implícito más bien que explícito; razonamiento aproximado en vez de preciso. Además, en cada materia concreta, se pretenderá que muchos conocimientos factuales, no metodológicos, sean elucidados por argumentaciones que involucren datos imprecisos e información incompleta. Este tipo de argumentación será abundante en las materias de Sociales y Humanidades y también en algunas asignaturas de Ciencias, como la Física. En Matemáticas, el estudio de la Teoría de la Probabilidad le puede hacer sentir la utilidad de 'precisar' lo impreciso para hacer predicciones exitosas, así como el beneficio de este ejercicio en muchas actividades de la vida ordinaria.

Finalmente, en la madurez universitaria, el estudiante puede desear conocer la lógica fuzzy para incrementar su conocimiento de lógica, su conocimiento en general o para adquirir un instrumento que le permita resolver problemas o realizar tareas relacionadas con otras disciplinas; esto es, como un saber instrumental.

La lógica fuzzy ha conocido recientemente una popularización extraordinaria en medios de divulgación y de difusión científica. En los medios de divulgación se ha propagado en dos ámbitos diferentes: el de reportaje o noticia y en el publicitario. A esta última forma de 'propaganda' ha contribuido el incremento constante de productos de consumo masivo, como electrodomésticos, cámaras de vídeo, tensímetros, etc. que incorporan el sello 'lógica fuzzy' como un sinónimo de fiabilidad e innovación, lo que ha hecho que las marcas comerciales incluyan en la publicidad de sus aparatos etiquetas similares a: 'hecho con lógica fuzzy' o 'con control fuzzy'.

La creciente difusión científica ha sido favorecida por el incremento paulatino de la credibilidad en esta propuesta teórica, muy denostada en sus inicios, pero cada vez más respetada. Esto ha propiciado que cada vez sea mayor el número de libros de texto, no sólo de lógica, sino también de Informática, Filosofía (analítica) o Lingüística que incorporan capítulos dedicados a la lógica fuzzy. También empiezan a ser habituales las referencias a la lógica fuzzy en los libros de Ciencias Sociales (Pedagogía, Psicología, Sociología), sobre todo en aquellos que tratan aspectos metodológicos, donde esta lógica se suele ver como un instrumento útil para el análisis riguroso, adaptativo y eficiente de aspectos típicos de su objeto de estudio, tales como la subjetividad y naturaleza cambiante del juicio o comportamiento humano, lo que conlleva tratamiento de incertidumbre e imprecisión.

La presencia cada vez mayor de la lógica fuzzy en el panorama científico-cultural, tecnológico-comercial y educativo demanda una alfabetización en los principios básicos del razonamiento aproximado; es decir, una asistencia en la instrucción en lógica fuzzy.

Los apartados que siguen pretenden ser sólo una serie de pautas y recomendaciones muy generales y absolutamente opinables sobre la enseñanza de la lógica fuzzy, en qué períodos escolares se recomienda, qué tareas se proponen, y qué beneficios se esperan de su instrucción. También, se señalan algunas precauciones muy genéricas que se deben adoptar en su enseñanza, sobre todo a niveles avanzados.

## 2 ALGUNAS PAUTAS

Decíamos antes que el razonamiento ordinario está presente en los diferentes ámbitos de nuestro recorrido formativo, desde la educación escolar hasta la universitaria. Parece, por tanto, que la lógica fuzzy, en tanto lógica del sentido común, debería tener un papel en las diferentes etapas educativas. Veamos algunas de las posibles tareas que le reservamos en cada una de esas etapas, así como la forma de introducir a los alumnos en las mismas.

### 2.1 En la educación preescolar ( y quizás en el inicio de la escolar)

En esta etapa el niño se inicia en el conocimiento usualmente a través de la manipulación y el juego. Los programas de educación lógica (clásica) a través del juego conocieron cierto desarrollo en la década de los 70. Piaget y Beth hicieron interesantes propuestas teóricas a este respecto e influenciaron toda una serie de experiencias sobre la ejercitación de un pensamiento prelógico. Respecto a la lógica fuzzy, en el ámbito de la educación preescolar y en el inicio de la etapa escolar el juego debería servir para empezar a familiarizar al infante con el concepto de regla imprecisa, ayudándole a comprender como en los juegos existen reglas que a veces son de dudosa aplicabilidad; esto es, que hay casos conflictivos donde la elección 'si' o 'no' de la regla no es inmediata, sino que requiere de interpretación y consenso acerca de un 'estado de hechos'.

*Tareas*: Adiestrarles en la ejercitación o invención de juegos y en la explicitación y acuerdo sobre las reglas que los definen, así como en imaginar situaciones conflictivas donde la aplicación de la regla sea dudosa.

*Propósito*:

- Familiarizarles con el concepto de regla imprecisa
- Distinguir, de modo incipiente, entre conocimiento explícito y conocimiento implícito.

## 2.2 En la educación escolar.

En la etapa escolar, al niño se le empieza a educar en contenidos concretos, pero además, también debe formársele en aspectos metodológicos que favorezcan el aprendizaje: en efecto, debe enseñársele cómo estructurar la información para que le sea más fácil de aprender, o cómo enfrentarse a la resolución de problemas -sobre todo si estos son de naturaleza imprecisa- de una manera rigurosa. Este aprendizaje metodológico, que pretende preparar al individuo para que formule y verifique hipótesis por si mismo -esto es, para que sea creativo-, es de naturaleza típicamente heurística y requiere de una formación en un tipo de razonamiento que será, en la mayoría de las ocasiones, más bien implícito que explícito, más bien analógico que algorítmico.

Pero en la etapa escolar, la lógica fuzzy, además de tener una utilidad formativa, debería tener también un papel propedéutico y moralizador, ayudando a conformar el sentimiento en el individuo de que la cautela y la prudencia son virtudes humanas y científicas. Debería servir para enseñar las limitaciones y el peligro de usar generalizaciones absolutas, haciendo ver como en muchas situaciones de la vida cotidiana es menos comprometedor y más ético usar oraciones de significado impreciso que de significado preciso.

*Tareas*: Enseñar a distinguir oraciones de significado preciso e impreciso, separando las unidades léxicas que generan esa imprecisión. Distinguir argumentos precisos e imprecisos y consecuencias precisas e imprecisas. Enseñar a distinguir conjuntos o colecciones de cosas que tienen límites precisos y otras que tienen límites imprecisos. Enseñar a hacer razonamientos muy elementales con enunciados de significado impreciso, así como a verificar grados de corrección de consecuencias aproximadas.

*Propósito*:

- Vincular la cautela a una actitud ética interesante como forma de aproximarse al estudio de un problema o al diagnóstico del mismo.
- Entender la imprecisión como una característica del lenguaje a veces evitable, otras veces inevitable, pero siempre natural.
- Asociar la imprecisión con la prudencia en el juicio, usual en las relaciones humanas ordinarias, pero nunca con la parálisis.
- Asociar la imprecisión a la negociación, al acuerdo sobre lo que entendemos que es el significado de una oración. Esto implica ver la construcción del significado, para muchos términos del lenguaje natural, no como algo dado, sino como algo a determinar, a pactar.
- Mostrar el papel del lenguaje natural en la construcción y justificación del conocimiento humano.

En resumen, un objetivo de esta etapa es dignificar un tipo de razonamiento que está presente en nuestra vidas y que empleamos constantemente en multitud de toma de decisiones.

## 2.3 En la educación secundaria.

En el período secundario, el alumno debería ser capaz de aislar, en sus diferentes materias curriculares, razonamientos que involucren el uso de oraciones de significado vago, así como captar la utilidad del razonamiento impreciso para justificar conclusiones de modo adecuado. La introducción de la lógica fuzzy debería hacerse a propósito de sus diferentes materias de estudio, así como respecto al perfeccionamiento y aprendizaje de pautas metodológicas que le ayuden a incrementar sus destrezas en la adquisición del conocimiento y en la resolución de problemas.

*Tareas:* Enseñarles a justificar racional-mente conclusiones aproximadas. Enseñarles la importancia del razonamiento analógico en la resolución de problemas; en particular, de aquellos que admiten un alto grado de formalización. Enseñarles la utilidad de resolver acertijos lógicos. En aquellos acertijos que incluyan el uso de probabilidades, los alumnos podrán capturar la importancia de la imprecisión en la toma de decisiones y del éxito de la lógica probabilística en la formalización de esta tarea.

*Propósito.*

- Entender al razonamiento impreciso como un tipo de razonamiento útil, al cual es posible acercarse de un modo riguroso con el lenguaje natural.
- Adquirir destrezas metodológicas vinculadas al análisis racional de la argumentación o a la resolución de problemas que involucran razona-miento impreciso.

## 2.4 En la enseñanza profesional.

Algunos alumnos habrán optado, en su adolescencia, por una enseñanza profesional, orientada sobre todo al aprendizaje de técnicas específicas para resolver problemas que se presentan en tecnologías de uso corriente y actual. En este apartado la enseñanza de algunos principios de la lógica fuzzy para el control parece interesante, porque empieza a haber bastantes aparatos de uso doméstico que funcionan con control fuzzy.

*Tareas.* Enseñarles, de modo práctico, los principios del control fuzzy, su aplicación y su utilidad. Animarles a que replanteen problemas habituales con esta nueva metodología (en aquellos casos en los que le parezca adecuado).

*Propósito.* Tener técnicos peritos en el control de procesos fuzzy.

## 2.5. En la enseñanza superior.

Ya que en este período la capacidad de elección del individuo es mayor, las propuestas deben ser necesariamente diversas. A continuación, consideramos diferentes tipos de alumnos a los que le puede interesar aprender lógica fuzzy en su última etapa formativa. Constituye la respuesta a la pregunta:

**¿A quién le interesa?**

En la enseñanza superior, las personas pueden acercarse al estudio de la lógica fuzzy al menos desde una triple perspectiva:

(a) para ampliar conocimiento:

**(a1)** específico, para aquellos que habiendo cursado estudios de lógica deseen extender su formación en esta materia aprendiendo una lógica nueva, o

**(a2)** genérico, para aquellos que aún no habiendo estudiado cursos previos de lógica sientan curiosidad intelectual por este nuevo ámbito, por un saber que se está expandiendo con profusión y rigor.

**(b)** para disponer de herramientas que ayuden a estudiar problemas teóricos nuevos, hasta ahora inéditos o por los que había una escasa preocupación y , por último,

**(c)** para diseñar productos que incorporen tecnología fuzzy.

A continuación, para cada uno de los alumnos tipo considerados, responderemos a tres preguntas:

- **¿Cómo enseñarla?**
- **¿Quién y dónde debería enseñarla?**
- **¿En qué modalidad de asignatura?**

***(Alumno tipo a1) Para ampliar conocimiento lógico***

**¿Cómo enseñarla?** Para (a1) debería plantearse un programa que incluyese los temas clásicos de la lógica borrosa, contextualizándolos en la tradición lógica; esto es, mostrando a la lógica fuzzy como heredera en algunos aspectos de la lógica clásica y de la lógica polivalente y justificando su estudio en tanto que responde al análisis de un tipo de consecuencia muy común en el lenguaje ordinario.

Capítulos del texto de Trillas, Alsina y Terricabras, *Introducción a la lógica borrosa* o del de J. P. Cleave, *A Study of Logics*, podrían ser manuales adecuados para este propósito.

También se podría incluir, quizás como un apartado final del programa, algún tema relacionado con aspectos críticos de esta lógica: qué diferencia a la lógica fuzzy de las lógicas multivaloradas; hasta dónde llega su carácter lingüístico; qué conflictos hay entre su disposición lingüística y su gestión numérica, etc. Una reflexión sobre el contenido de los libros citados podría ser suficiente para atender algunos de estos puntos, aunque eventualmente quizás convendría recomendar alguna lectura complementaria.

**¿Quién y dónde debería enseñarla?** Atendiendo a que las Facultades que imparten cursos regulares de Lógica son las de Filosofía y Matemáticas, en ambas titulaciones podría ser enseñada por profesores de esas Facultades formados en lógica fuzzy.

**¿En qué modalidad?** Quizás como una materia optativa, ubicada en algún curso que garantizase que los alumnos hubiesen estudiado ya, al menos, Lógica y Teoría de conjuntos clásica.

***(Alumno tipo a2) Para ampliar conocimiento en general***

**¿Cómo enseñarla?** Para (a2) debería plantearse un temario en el que apareciera una introducción muy genérica a los diferentes ámbitos de la lógica borrosa y de sus aplicaciones. Se trataría de que se le hiciese al alumno un poco de historia del análisis lógico de la vaguedad y se le situase en el marco de preocupaciones en el que surgió la lógica fuzzy. Las entrevistas hechas a Zadeh y Trillas deberían ser lecturas recomendadas. También podrían ser útiles libros de divulgación sobre lo fuzzy, como *Fuzzy logic* de D. McNeill y P. Freiberger o *Fuzzy Thinking* de B. Kosko. Algunas partes de estos libros, que para el neófito podrían ser desorientadoras o fuente de confusión, deberían ser específicamente comentadas por el profesor.

A esta introducción debería seguirle una exposición de los aspectos fundamentales de la lógica fuzzy, a ser posible con ejemplificaciones constantes y pegada a ejemplos del lenguaje natural, como ocurre con algunos trabajos de Zadeh, que parecen una forma recomendable de acercarse al estudio de la lógica fuzzy desde esta perspectiva. Algún libro elemental, que incluye frecuentes ejemplos, como *Fuzzy sets, fuzzy logic, applications* de G. Bojadziev y M. Bojadziev puede servir también como apoyo para este fin.

Finalmente, debería incluirse un capítulo en el que se estudiasen algunas aplicaciones sencillas e ilustrativas de la lógica fuzzy. Alguna descripción divulgativa de MILORD podría ser adecuada para este propósito.

**¿Quién y dónde debería enseñarla?** Como materia de 'formación cultural', sería oportuno que fuese impartida de manera interdisciplinar o, sino, por un profesor de formación generalista, como los que se encuentran en las Facultades de Filosofía. Ahora bien, en este último caso, debería garantizarse que la persona encargada tuviese vocación de hacerlo y mantuviese colaboración o contactos con personas que trabajan en ámbitos teóricos o aplicados diferentes al suyo.

**¿En qué modalidad?** Quizás como una materia optativa, orientada fundamentalmente a que los estudiantes la escojan en la modalidad de créditos de libre elección, con los que completan conocimientos o satisfacen inquietudes intelectuales alejadas de su formación específica.

*(Alumno tipo b) Para estudiar problemas nuevos*

**¿Cómo enseñarla?** La enseñanza de la lógica fuzzy debería estar dirigida hacia el aprendizaje de aquellas técnicas teóricas que permiten analizar problemas hasta hace poco inéditos. Se orientaría fundamentalmente hacia disciplinas en las que la lógica fuzzy haya tenido aplicaciones importantes.

El temario debería comprender una parte de contenidos de lógica fuzzy y, a la par, aspectos de esas ciencias que esta lógica permite clarificar o poner en limpio. Sus contenidos serían, por tanto, diversos, según la disciplina en la que se estudiasen las aplicaciones. Podría responder a títulos sonoros como: 'Todo lo que un ...(especialista en una materia concreta) quisiera saber de lógica fuzzy pero no se atreve a preguntarlo' o, quizás, en una orientación un poco distinta, 'qué importancia tiene la lógica fuzzy para la ... (materia especifica)'. El libro de T. Williamson, *Vagueness*, sería un posible texto a seguir en el caso de Filosofía.

**¿Quién y dónde debería enseñarla?** Deberían enseñarla profesores de cada una de las materias, con conocimientos de lógica fuzzy. Debería enseñarse en cada una de las Facultades en las que se propusiese.

**¿En qué modalidad?** Quizás como una materia optativa, orientada fundamentalmente a los estudiantes que deseen ampliar alguna de las asignaturas ya cursadas como troncales u obligatorias. En la licenciatura de Filosofía, p. ej., como un complemento a la Filosofía de la Lógica o a la Filosofía del Lenguaje, para entender mejor algunos problemas que, como la paradoja del Sorites o la identidad en el primer caso, o las sinonimia en el segundo, requieren para su análisis y comprensión de algunos rudimentos en lógica fuzzy.

*(Alumno tipo c) Para diseñar tecnología moderna*

**¿Cómo enseñarla?** La enseñanza de la lógica fuzzy debería estar dirigida hacia el aprendizaje de aquellas técnicas que permitan la obtención de productos tecnológicos que incorporen sentido común. Se orientaría fundamentalmente hacia las disciplinas tecnológicas, como ingeniería informática e ingeniería industrial.

En esta orientación, el temario debería incluir una primera parte de bases teóricas de la lógica fuzzy y una segunda de aplicaciones originadas por las mismas. Los textos de H. J. Zimmermann, *Fuzzy set theory and its applications* y Terano, Asai y Sugeno, *Fuzzy systems theory and its applications*, podrían ser útiles en este cometido. Una introducción a las tecnologías fuzzy en el contexto de la historia de las tecnologías sería también conveniente para una visión histórico-cultural de este ámbito.

**¿Quién y dónde debería enseñarla?** Deberían enseñarla profesores de las Facultades de Informática e Ingeniería Informática especializados en lógica y tecnologías fuzzy. Debería impartirse en las Facultades de Informática e Ingeniería Informática y quizás en otras Facultades Politécnicas, como Ingeniería Industrial o Telecomunicaciones.

**¿En qué modalidad?** Quizá como materia obligatoria en Informática e Ingeniería Informática. Debido a la cantidad de productos de consumo que incorporan lógica fuzzy y a la comercialización de paquetes de software con herramientas especializadas en esta lógica, parece imprescindible que los alumnos de esta titulación tengan algunas nociones básicas de esta lógica.

Como materia optativa en otras titulaciones técnicas interesadas en aplicar las tecnologías fuzzy a sus ámbitos de trabajo.

## 3 A MODO DE RECAPITULACIÓN

En este trabajo hemos querido suscitar un debate en torno a la oportunidad y conveniencia de una alfabetización en lógica fuzzy. En cada una de las etapas educativas hemos sugerido, de modo justificado, algunas tareas relacionadas con el aprendizaje de aspectos relativos al razonamiento aproximado y hemos apuntado los posibles beneficios que ello tendría en la formación del individuo. Entre las tareas pendientes está la elaboración de programas más detallados de aquellos aspectos de la lógica fuzzy que en cada etapa fuese conveniente enseñar, así como una propuesta de la metodología más conveniente para hacer efectiva esa enseñanza. Esto debería resultar de una discusión en la que participase toda la 'comunidad fuzzy' de España y tendría el efecto benefactor de contar con una propuesta normalizada de cómo educar en lógica fuzzy, conveniente para convencer a individuos y autoridades académicas del interés de esta acción.

## AGRADECIMIENTOS

Con el profesor Enric Trillas he discutido algunas de las propuestas que aqui se hacen. De ello no debe inferirse que las comparta.

## BIBLIOGRAFIA

ALSINA, C. & TRILLAS, E. 'Fuzzy Sets and Mathematical Education'. *For the Learning of Mathematics*, 11, 3, (November, 1991), 16-19.

'Guidelines for Logic Education', The ASL Commitee on Logic and Education. *The Bulletin of Symbolic Logic*, vol. 1, n° 1, March, 1995.

TRILLAS, E. & SOBRINO, A., 'Nota sobre la enseñanza de la lógica en el Bachillerato', *Suma, Enseñanza y aprendizaje de las matemáticas*, nº 7, 1990,17-22.

# SOFTWARE DE COMPILACIÓN, SIMULADOR Y EMULACIÓN BORROSOS

**Gerardo Aranguren, Miguel Rodríguez, Joseba López de Arroyabe y Nuria Altarriba**
Dpto. Electrónica y Telecomunicaciones, E.T.S. Ing. Ind. y Telecom., Universidad del País Vasco
Alameda Urquijo s/n, 48013 Bilbao
jtpararg@bi.ehu.es

## Resumen

En el congreso Estylf'96 los autores presentaron una serie de simplificaciones a los algoritmos borrosos que permitían la realización de un controlador secuencial borroso en un circuito integrado programable (FPGA). En el presente artículo se comenta la realización de un software que permite la edición de los términos lingüísticos, las reglas borrosas y la aplicación automática de las reglas de simplificación enunciadas de cara a la compilación del programa del controlador. Además se comentan otras características como la introducción de máscaras y la edición en jerarquía. El software cuenta además con un simulador y un emulador.

**Palabras Clave:** Software simulación, Control borroso, Simplificaciones borrosas.

## 1 INTRODUCCIÓN

Desde que Zadeh [1] dio origen a la lógica borrosa han sido muchos los enunciados matemáticos que se han realizado de los algoritmos de control borrosos [2], [3], [4]. Todos ellos tienen su ámbito de validez y es difícil pronunciarse sobre cual es el mejor o el que incluye todas las posibilidades. En algunas ocasiones, además, el óptimo no coincide con el mejor, sino con el más simple dentro de lo aceptablemente correcto. Con este objetivo en el congreso ESTYLF'96 [5] se presentaron tres simplificaciones al algoritmo borroso más estandarizado que permitían la implementación de un controlador borroso mediante un circuito con paralelismo segmentado (pipe-line) en un circuito integrado programable del tipo FPGA (Field Programmable Gate Array).

Un enunciado breve de estas tres simplificaciones es el siguiente:

1. Descomposición de términos. Las funciones de pertenencia de los subconjuntos borrosos sólo pueden adoptar las formas de S-términos o Z-términos. Las funciones con forma de trapecio o triángulo también son aceptables pero en el proceso de compilación se descompone cada término en dos términos uno del tipo S-término y otro del tipo Z-término. Con esto se logra una simplicidad y uniformidad de los términos borrosos que quedan definidos únicamente por dos puntos.

2. Heurística. En cualquier controlador que actúe de forma secuencial se puede prescindir de reglas o antecedentes cuyo grado de cumplimiento sea siempre nulo o igual a la unidad. En la mayoría de los controladores hardware esto no es posible por su actuación no secuencial. Un estudio interesante hecho al respecto es el publicado por Samoladas [6].

3. Reducción de reglas y subpremisas. En un enunciado matemático de las variables borrosas procuramos crear un número suficiente de subconjuntos borrosos que nos permita emitir todas las reglas borrosas que sean necesarias. Pero si el enunciado es lingüístico no respetamos, habitualmente, todos los subconjuntos borrosos realizando simplificaciones de gran interés. Por ejemplo, podemos decir que no circulamos rápido, incluyendo en esta afirmación que no circulamos ni rápido ni muy rápido. De la misma forma, dos reglas con antecedentes vecinos que concluyan en un mismo consecuente se pueden simplificar.

Estas tres simplificaciones permiten que los antecedentes que se deben examinar sean simples y que el número de antecedentes y consecuentes a examinar sea lo más reducido posible, dos condiciones muy útiles para la realización de un controlador borroso.

## 2 SOFTWARE DE DEFINICIÓN DE SUBCONJUNTOS

Para la definición de los subconjuntos borrosos se ha realizado un software con todas las opciones de edición de variables de entrada de los tipos permitidos y variables de salida de tipo simple. Este software escrito en C++ es muy similar a otros de definición de sistemas de control borrosos.

En la figura 1 se puede ver la definición de una variable borrosa mediante 4 subconjuntos borrosos. El programa permite definir hasta 32 variables de entrada cada una de ellas especificada hasta con 32 términos lingüísticos, lo que en la práctica es suficiente para cualquier aplicación.

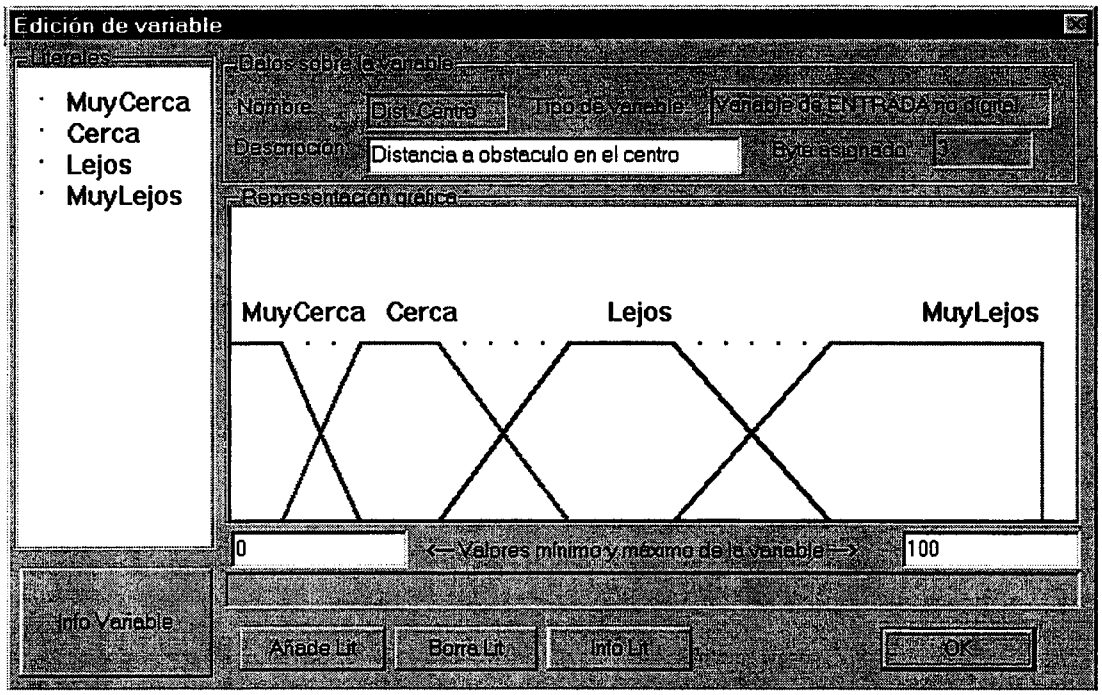



Figura 1. Ventana de edición de las variables y conjuntos borrosos de entrada.

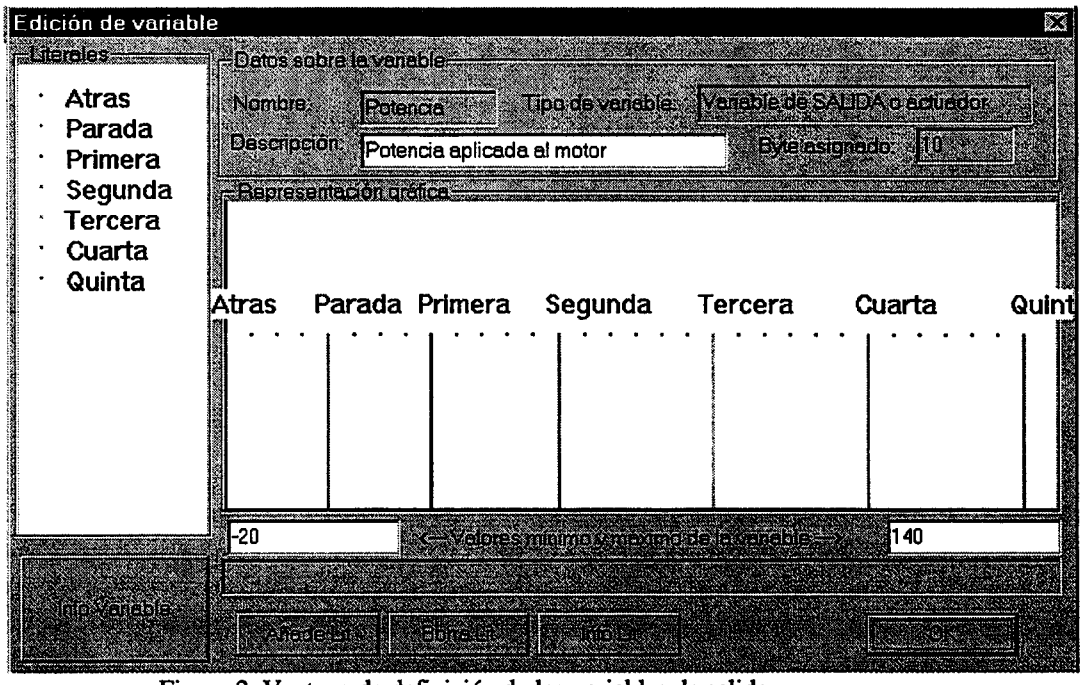



Figura 2. Ventana de definición de las variables de salida.

Para la definición de los términos lingüísticos correspondientes a las variables de salida se utiliza una pantalla similar (figura 2). Como el algoritmo que se va a emplear es el de Media de los Máximos basta con definir cada literal por un único valor, semejante a indicar el punto máximo de un triángulo o un trapecio.

En estas dos últimas pantallas se puede observar que existe un procedimiento completo de edición de nuevos términos, eliminación de términos, información, etc.

## 3. EDICIÓN DE REGLAS

La edición de reglas se realiza sobre una tabla de dos entradas como la mostrada en la figura 3. En cada tabla intervienen dos variables de entrada, una de salida y una máscara. Sobre esta pantalla se pueden mostrar las simplificaciones de antecedentes y de reglas según el enunciado de las simplificaciones.

Al pulsar el botón de mostrar simplificaciones aparecen los cuadros de la figura señalando que reglas se van a asociar en una única regla.

## 4 MENÚ PRINCIPAL

En la figura 4 se puede ver el menú principal del software "Fuzzy Variables Manager". A partir de esta pantalla se pueden definir las variables de entrada y salida y las reglas descritas en los apartados anteriores y, además, se obtiene el paso a la definición de las máscaras, la jerarquía de reglas, la compilación, el simulador y el emulador.

### 4.1 VARIABLES DE MÁSCARA

Las variables de máscaras son variables digitales, sobre las que actúa el supervisor del proceso controlado, que permiten seleccionar una tabla para cada momento de la operación. Por ejemplo, en el guiado de un vehículo el sistema borroso debe utilizar distintas estrategias según el conductor quiera avanzar hacia delante (MARCHA en la figura), o aparcar el vehículo (PARADA). La máscara crea una condición digital de orden superior a las estrategias definidas en las tablas de reglas.

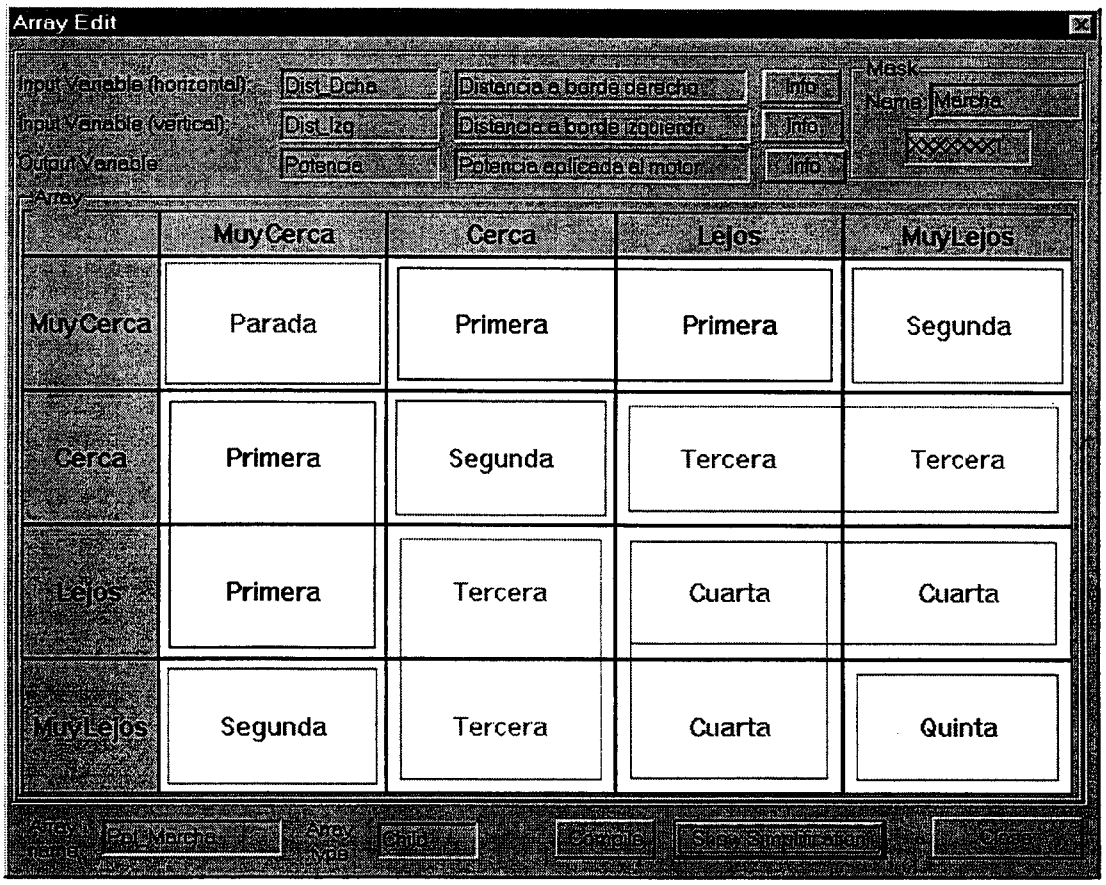



Figura 3. Definición y simplificación de reglas.

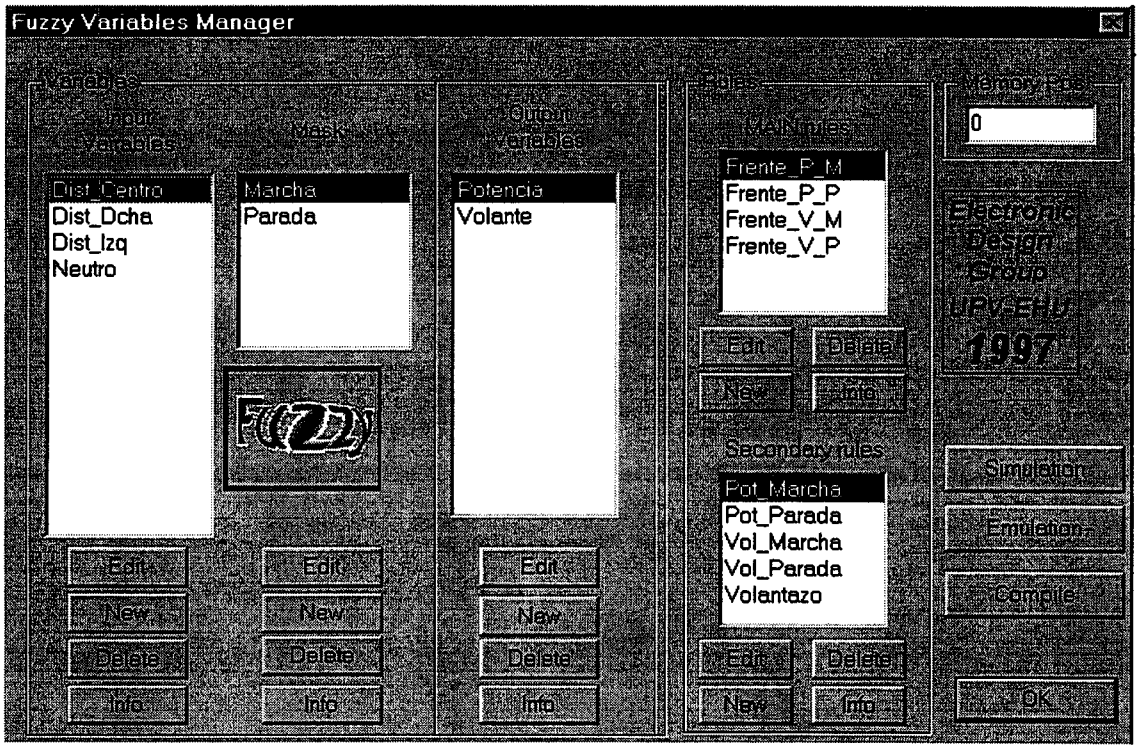



Figura 4. Pantalla principal de programa.

## 4.2 JERARQUÍA DE REGLAS

En la figura 4 vemos que existen dos tipos de tablas de reglas: principales y secundarias. En las principales, además de lo ya descrito, se puede sustituir el consecuente de una celda de la matriz por una llamada a una tabla secundaria, permitiendo de esta forma la definición de reglas de más de dos variables. Las tablas secundarias son muy semejantes a las principales pero están condicionadas a la tabla principal a que se asocian en cuanto a la máscara y el consecuente.

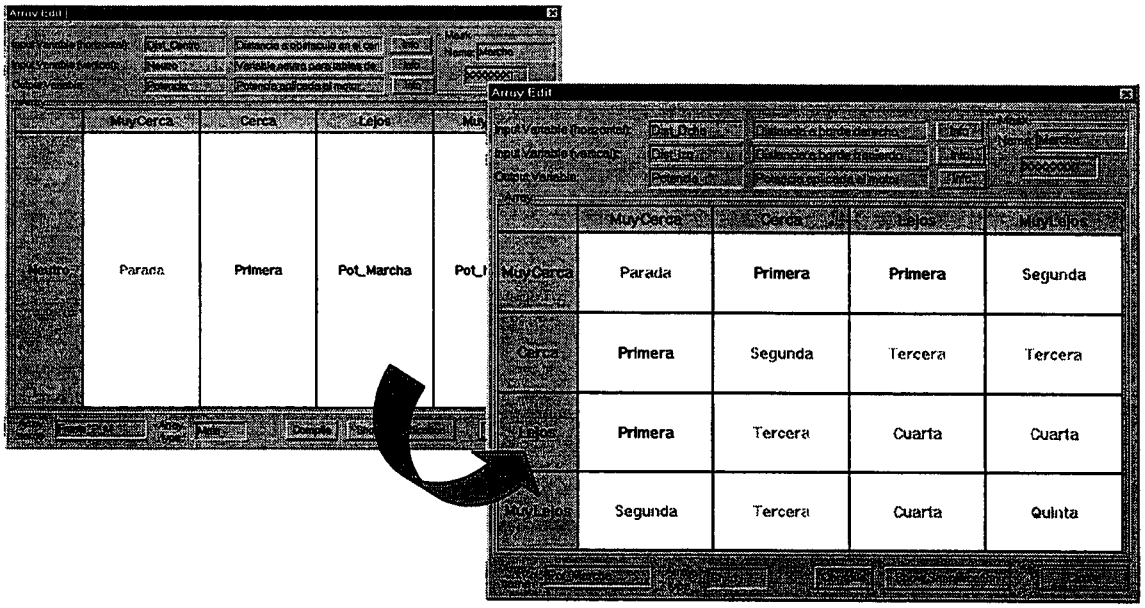



Figura 5. Enlace de una tabla principal con una tabla secundaria.

En la figura 5 se muestra un ejemplo práctico del enlace jerárquico de tablas. El sistema de control del ejemplo consta de un árbol de decisión con cuatro tablas principales y cinco tablas secundarias. En la figura 5 desde dos reglas de la tabla principal se llama a una tabla secundaria permitiendo el empleo de más de dos variables en un sistema de control.

### 4.3 COMPILADOR

El botón Compilar, de la pantalla principal (figura 4), genera un informe con el programa final usado en el circuito integrado fuzzy diseñado. Este programa compilado también es utilizado por el simulador. En la figura 6 se muestra una parte del listado del programa generado. Se puede apreciar el salto producido por las máscaras para seleccionar las tablas adecuadas (líneas 2 y 3). En las líneas 4 a 6 se ve el enlace creado entre dos tablas. En muchas de las líneas se aprecian las simplificaciones de subpremisas, por ejemplo en la línea 10 se examina que distancia a derecha es Lejos-, lo que significa que puede ser MuyCerca, Cerca o Lejos.

## 5. SIMULADOR Y EMULADOR

Volviendo a la pantalla principal mostrada en la figura 4 se pueden ver las opciones de simulación y emulación. El simulador y emulador sólo son válidos para el ejemplo del péndulo invertido, ya que sólo se han introducido las ecuaciones físicas de funcionamiento del péndulo.

Este simulador tiene dos pantallas fundamentales: un diagrama tiempo-variables (figura 7) y una pantalla que muestra el grado de cumplimiento de cada consecuente y el cálculo del centro de gravedad (figura 8). En estas pantallas se permite el funcionamiento paso a paso, continuo o manual.

La función principal del simulador es comparar los resultados con los obtenidos con otros simuladores y valorar el sistema de control a implementar en el controlador hardware.

El emulador tiene un carácter más genérico. Se encarga de transferir los datos de variables de entrada y salida entre el computador y un controlador hardware, presentando los datos en la pantalla de tiempos. Evidentemente, es necesario disponer del controlador y el interface adecuado.

## 6. CONCLUSIONES

Se ha realizado un software de definición de variables y reglas borrosas que incorpora una serie de novedades en cuanto a simplificación de antecedentes y reglas, adición de máscaras y jerarquía de reglas que permite definir múltiples estrategias para un proceso y hacerlas depender de más de dos variables.

También se ha incorporado un simulador y un emulador que permiten comparar los resultados con los obtenidos por otros simuladores y controladores hardware.

```
[    0] 0000XXXXXXXXXXX (0,0)  NOP
[    1] 0000XXXXXXXXXXX (0,0)  NOP
[    2] 00010111XXX01000 (17,8) ***  Salto condicional: bit 0
[    3] 0000000010011101 (0,9d)   *  salto a (157)
[    4] 1010000110100011 (a1,a3) Dist_Centro is Lejos+
[    5] 1100011011100010 (c6,e2) Dist_Dcha is MuyLejos+
[    6] 1100011011100001 (c6,e1) Dist_Izq is MuyLejos+
[    7] 0100111111XXXXXX (4f,80) then Potencia is Quinta
[    8] 1010000110100011 (a1,a3) Dist_Centro is Lejos+
[    9] 1010000110100010 (a1,a2) Dist_Dcha is Lejos+
[   10] 1101111000100010 (de,22) Dist_Dcha is Lejos-
[   11] 1010000110100001 (a1,a1) Dist_Izq is Lejos+
[   12] 0100110010XXXXXX (4c,80) then Potencia is Cuarta
[   13] 1010000110100011 (a1,a3) Dist_Centro is Lejos+
[   14] 1010000110100010 (a1,a2) Dist_Dcha is Lejos+
[   15] 1010000110100001 (a1,a1) Dist_Izq is Lejos+
[   16] 1101111000100001 (de,21) Dist_Izq is Lejos-
[   17] 0100110011XXXXXX (4c,80) then Potencia is Cuarta
[   18] 1010000110100011 (a1,a3) Dist_Centro is Lejos+
```

Figura 6. Ejemplo de un programa objeto generado por el software de definición de sistemas borrosos.

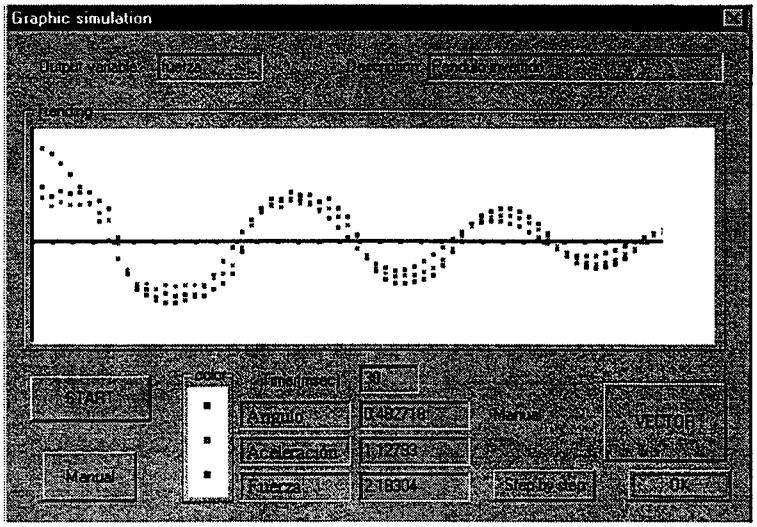



Figura 7. Simulador: pantalla tiempo-variables

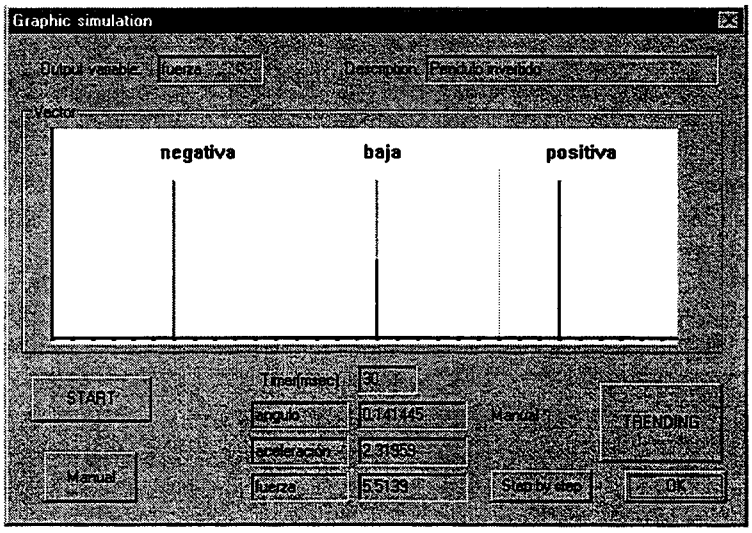



Figura 8. Simulador: pantalla de cálculo de consecuente

**Agradecimientos**

La realización de este software está encuadrada dentro del proyecto "Circuito Integrado de Control por Inferencia Borrosa (CICIB)" subvencionado por la Universidad del País Vasco/Euskal Herriko Unibertsitatea, UPV 147.345-TB058/96.

**Referencias**

[1] L. A. Zadeh, "Fuzzy Sets", Informat. Control, vol. 8, pp. 338-353, 1965.

[2] C. C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller", IEEE Trans. Syst. Man Cybern., vol. SMC-20, nº 2, pp. 404-435, 1990.

[3] W. Pedrycz, "Fuzzy Control and Fuzzy Systems", Ed. John Wiley & Sons, 1993.

[4] R. Reyero, C. F. Nicolás, "Sistemas de control basados en lógica borrosa: "Fuzzy Control", Omron Electronics, S. A. y IKERLAN, Madrid, 1995.

[5] G. Aranguren, M. Rodríguez y M. Barrón, "Controlador Secuencial Segmentado para Lógica Borrosa Implementado en FPGA", ESTYLF'96, Oviedo, pp. 255-259, 1996.

[6] V. Samoladas, L. Petrou, "Special-purpose Architectures for Fuzzy Logic Controllers", Microprocessing and Microprogramming, vol. 40, nº 4, pp. 275-289, 1994.

# INFLUENCIA DE LA PARTICIÓN DE ENTRADA EN LA LEY DE CONTROL FUZZY: UN EJEMPLO ANALÍTICO

**M. Santos**
Dpto. Informática y Automática.
Facultad de CC. Físicas. UCM
email: msantos@eucmax.sim.ucm.es

**J.M. de la Cruz**
Dpto. Informática y Automática.
Facultad de CC. Físicas. UCM
email: jmcruz@eucmax.sim.ucm.es

**S. Dormido**
Dpto. Informática y Automática.
Facultad de Ciencias. UNED
email: sdormido@uned.es

## Resumen

No es fácil delimitar la repercusión de los parámetros de configuración de un regulador fuzzy en su acción de control. Sin embargo, un conocimiento general de su influencia facilita la correcta elección, esencial para el éxito de la aplicación. En este trabajo se ha partido de la ley de control analítica de un regulador borroso para establecer de forma rigurosa la influencia de la partición de entrada.

**Palabras Clave**: Controladores Fuzzy, Parámetros de Configuración, Partición de Entrada.

## 1 INTRODUCCIÓN

El problema del diseño de los sistemas de control se reduce a la **elección** de un determinado controlador y al **ajuste de sus parámetros**, de tal forma que se alcancen ciertas especificaciones.

La elección de un controlador basado en la lógica borrosa viene motivada tanto por la naturaleza de la información con la que se trabaja (información lingüística, imprecisa, ...) como por el tipo de aplicación para la que se implementa: sistemas complejos, no lineales, etc., y la naturaleza de los objetivos.

Un **Controlador basado en Lógica Fuzzy** (FLC) modela de forma lingüística la estrategia de control utilizada por un operario o experto en el control manual del proceso.

Una vez elegida la lógica borrosa para implementar el regulador, éste admite una amplia variedad de posibilidades, tanto en su configuración inicial como para el ajuste de sus parámetros, que van a influir notablemente en su funcionamiento.

La configuración inicial de un FLC se lleva a cabo dando forma o valores adecuados a los diversos parámetros que intervienen en él. En cada uno de sus componentes (Figura 1), intervienen una serie de elementos que admiten a su vez cierta variedad de especificaciones, lo que hace que estos sistemas estén sobreparametrizados, y cuenten con un elevado número de factores que repercuten significativamente tanto en el funcionamiento como en la robustez y estabilidad del controlador [2].

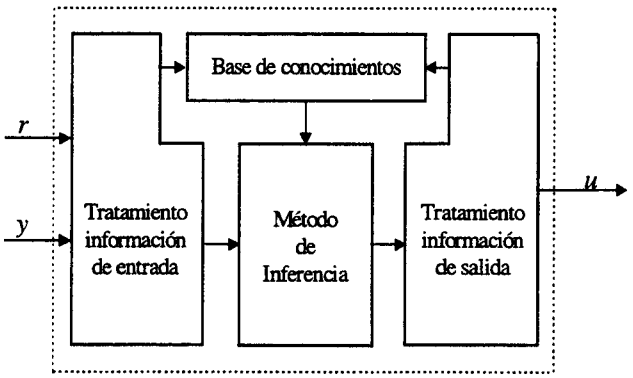



Figura 1: Componentes de un Controlador Fuzzy

Es importante considerar que en un sistema de control borroso se puede actuar sobre cualquiera de sus elementos, pero no siempre se conocen las consecuencias directas de estas variaciones porque no está clara la relación implícita que existe entre esos parámetros y la respuesta del sistema [7].

El estudio de todas las combinaciones posibles sería una tarea inabordable, ya que no existe una única determinación que optimice el sistema de control. Además, debido a las fuertes relaciones que existen entre

los componentes del controlador fuzzy, no se puede tratar cada elemento por separado, ni computar de forma independiente su influencia sobre la salida. Sin embargo, algunas especificaciones se van imponiendo por la generalización de su uso o simplicidad [10].

En este trabajo hemos abordado el estudio de uno de los parámetros de configuración del FLC, la partición de entrada, ya que un mejor conocimiento de su influencia permite su correcta elección y la mejora del funcionamiento del sistema de control.

Para ello, en el punto 2, tras encuadrar los parámetros de configuración de un controlador fuzzy, se expone de forma cualitativa la influencia de la partición de entrada en la acción de control, resultados que se justifican con el estudio analítico de la acción de un FLC que se lleva a cabo en la sección 3. En el apartado 4 se presentan los resultados mediante ejemplos de simulación. Por último, se resumen las conclusiones.

# 2 PARÁMETROS DE CONFIGURACIÓN DE UN FLC

Se pueden definir como parámetros de configuración de un FLC aquellos que, una vez fijados en una fase previa de selección, *no conviene* variar durante la aplicación. Sin embargo, como su especificación afecta al comportamiento del controlador, es necesario hacer un estudio de su influencia en el dominio de trabajo para hacer una elección adecuada. A esta categoría pertenecen, por ejemplo:

- Número y determinación de las variables de entrada y de salida. Incorporar múltiples entradas y salidas (MIMO) no varía su estructura general, pero sí aumenta la complejidad.
- Número y especificación de los términos lingüísticos asociados a cada variable (partición de los espacios de entrada y salida).
- Discretización y normalización de los universos de discurso.
- Método de inferencia aplicado para obtener la conclusión.
- Estrategias de borrosificación y de desborrosificación empleadas para convertir la información concreta ↔ borrosa

Frente a ellos, los parámetros de ajuste son aquellos que pueden tomar distintos valores iniciales para calibrar una misma estructura, o que se pueden variar en línea con el fin de obtener unas especificaciones deseadas. Se pueden considerar, entre otros:

- Factores de escala o ganancias de las variables de entrada y de salida.
- Forma, media o centro, amplitud, etc., de las funciones de pertenencia asociadas a los términos lingüísticos, tanto para variables de entrada como de salida.
- Base de reglas.

Sobre estos últimos se realiza la sintonía, obteniendo su valor inicial mediante algún procedimiento que tenga en cuenta la máxima información posible [9].

## 2.1 INFLUENCIA CUALITATIVA DE LA PARTICIÓN DE ENTRADA

La *Partición borrosa de los espacios de entrada y salida* aborda el problema de la decisión del número de conjuntos fuzzy primarios o básicos asociados a cada variable lingüística, y determina la granularidad o finura del control borroso que se puede obtener.

La partición del espacio entrada/salida no es determinística y no tiene una única solución. Como ya se ha dicho, es difícil particularizar su influencia sin determinar la estructura de los demás componentes del sistema borroso. Usualmente se requiere un procedimiento de prueba para encontrar el tamaño óptimo de la partición.

Para su estudio, es importante considerar en primer lugar la determinación y número de las variables de trabajo, que se realiza según el dominio de la aplicación y la estructura del controlador a implementar; por ejemplo, la acción de control puede ser incremental o absoluta, y la configuración del controlador tipo PI, PID, etc.

También van a influir los distintos algoritmos o estrategias de borrosificación que se emplean para convertir la información concreta en valores borrosos, y que dependerán de la naturaleza de la información de entrada que reciba el FLC.

Dentro de la base de conocimiento, **la partición de entrada** hace referencia a los términos lingüísticos utilizados para caracterizar los antecedentes de las reglas borrosas de control, que son definidos subjetivamente, basándose en la experiencia y el juicio del ingeniero de control.

Su influencia va indisolublemente unida a la especificación de las funciones de pertenencia. Partiendo de que debe cubrir totalmente los universos de discurso correspondientes con un cierto grado de completitud, se le asocia la mayor o menor resolución del control que se

obtiene, según se empleen muchos o pocos términos respectivamente. En la práctica, es común encontrar desde 2 hasta 9 términos, pero lo más frecuente es utilizar 3, 5 ó 7. Para la variable de salida se utilizan generalmente mayor número de etiquetas que para las variables de entrada.

Un conjunto de términos primarios podría ser, por ejemplo, simétrico. Si no es el caso, o se usa un universo no normalizado, los términos pueden ser asimétricos o estar desigualmente distribuidos en el universo.

También es determinante en la dimensionalidad de la tabla de reglas, dado que la cardinalidad del conjunto de términos en un espacio de entrada determina el número de reglas de control que pueden ser construidas, con lo que esto conlleva de consumo o ahorro de memoria y tiempo computacional. Por ejemplo, para un sistema MISO con dos entradas (*e*, *ce*) y una salida, con *Ne* y *Nce* el número de conjuntos primarios para *e* y para *ce* respectivamente, el máximo número de reglas borrosas es el producto de las cardinalidades de ambas variables: *Ne* x *Nce*.

En la siguiente tabla (Tabla 1) se muestra la variación del número de reglas con la partición de entrada, tomando para cada una de las variables el mismo número *N* de términos lingüísticos.

Tabla 1: Número de reglas.

| | | *Número de variables de entrada* | | | |
|---|---|---|---|---|---|
| | | *2* | *3* | *4* | *5* |
| *N* | *2* | 4 | 8 | 16 | 32 |
| | *3* | 9 | 27 | 108 | 324 |
| | *4* | 16 | 64 | 256 | 1024 |

Por otro lado, la normalización de un universo continuo requiere un conocimiento *a priori* del espacio de entrada/salida ya que, para normalizarlo a un intervalo de valores -generalmente [-1,1]-, se requiere su previa discretización.

La cuantificación discretiza un universo de discurso continuo en cierto número de segmentos (*niveles de cuantificación*). La elección de los niveles de cuantificación refleja también cierto conocimiento del proceso y de las especificaciones, porque tiene una influencia directa en la resolución de la estrategia de control. Por ejemplo, podría buscarse una solución fina para errores pequeños, consiguiendo una precisión mayor con más niveles de cuantificación, y sería suficiente una resolución menor para errores grandes utilizando pocos niveles en torno a esa banda del error. Sin embargo, existen ciertas restricciones como que la discretización máxima que se admite es de 32 elementos.

En general, debido a la discretización, el control borroso es menos sensible a desviaciones en los valores de las variables de estado del proceso, pero produce otros efectos no deseables como oscilaciones, error estacionario, ciclos límites, etc. Cuando se utiliza una discretización trapezoidal [6] el tiempo de ejecución aumenta, pero el efecto de la discretización disminuye, mejorando la respuesta; particularmente cuando se consideran el error estacionario y los ciclos límites que se producen con la realización rectangular clásica.

Algunos de estos efectos aquí resumidos vienen comentados en [1], [3], [5], [8], y [11].

## 3 INFLUENCIA ANALÍTICA DE LA PARTICIÓN DE ENTRADA EN LA LEY DE CONTROL FUZZY

Para el estudio y formulación analítica de la influencia de la partición se va a analizar el comportamiento de los FLC tipo PI cuando se incrementa su partición borrosa del espacio de entrada, con lo que se introduce una nueva fuente de no linealidad en el controlador.

La influencia de la partición de los universos de discurso de las variables en la acción de control se ha comentado brevemente en la sección 2. Sin embargo, el desarrollo analítico de un controlador borroso en el que se aumenta su número de etiquetas lingüísticas para las entradas permite compararlo y formalizar esta influencia matemáticamente.

Para un FLC-PI (Figura 2) con variables de entrada error *e(t)* y cambio en el error *ce(t)*, y salida el incremento de control $\Delta u(t)$, se asignan dos etiquetas: n (negativo) y p (positivo) para cada variable de entrada, y tres para la salida añadiendo el término lingüístico z (cero), definidas de forma trapezoidal (Figura 3), con representación numérica o centro *Lx* ($x = e, ce$).

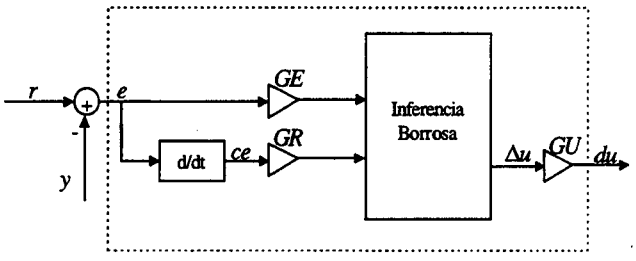



Figura 2: Controlador FLC-PI incremental.

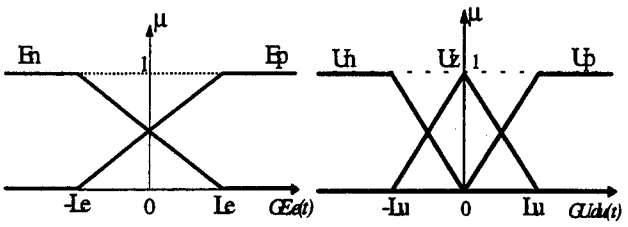


Figura 3: Funciones de pertenencia del error y del incremento de control.

La evaluación mediante el algoritmo de borrosificación del grado de pertenencia de las variables a los distintos conjuntos fuzzy se denota: Ei($e(t)$), CEj($ce(t)$), ..., donde el subíndice $i$ indica cualquiera de las etiquetas $i$ = n, p, z: Ep, CEn, Uz, etc. Las entradas son previamente ponderadas por sus correspondientes factores de escala $(GE, GR)$.

Un posible conjunto de reglas que describen el comportamiento del FLC-PI es el dado en la Tabla 2, donde cada una de ellas es de la forma:

$R_i$: ***si*** *GE.e(t) es Ei y GR.ce(t) es CEj* ***entonces*** *U es Uk*

Tabla 2: Reglas de un FLC-PI incremental con 2 etiquetas.

| | | *GE.e(t)* | |
|---|---|---|---|
| | U | **Ep** | **En** |
| *GR.ce(t)* | **CEp** | Up | Uz |
| | **CEn** | Uz | Un |

Al evaluar las reglas interpretando la conjuntiva como el mínimo, se obtienen ocho posibles conclusiones. Se tendrán por lo tanto ocho zonas de actuación del FLC-PI.

La salida del FLC se calcula mediante el método no lineal de la Media Borrosa (FM: Fuzzy Mean):

$$\Delta u_{FM} = \frac{\sum_{k=1}^{Nu} Uk.pk}{\sum_{k=1}^{Nu} Uk} = \frac{(Up - Un).Lu}{(Up + Uz + Un)} \qquad (1)$$

La contribución a la salida viene dada por las expresiones simbólicas:

Up = min (Ep($e(t)$)), CEp($ce(t)$))
Un = min (En($e(t)$), CEn($ce(t)$))
Uz =max(min(Ep($e$),CEn($ce$)),min(En($e$),CEp($ce$)))

Sustituyendo las expresiones matemáticas de las funciones de pertenencia de las variables, se obtiene una ley de control para cada una de las ocho zonas. Dado que las condiciones de los antecedentes están relacionadas entre sí, debido a la definición de las funciones de pertenencia, estas zonas se reducen a dos, que se pueden expresar en función de las variables de entrada de la forma [4]:

***Zona I***: $GR.|ce(t)|/Lce \leq GE.|e(t)|/Le$

$$\Delta u(t) = \frac{0.5Lu}{2Le - GE.|e(t)|} GE.e(t) + \frac{0.5Lu.Le}{Lce(2Le - GE.|e(t)|)} GR.ce(t)$$

***Zona II***: $GE.|e(t)|/Le \leq GR.|ce(t)|/Lce$

$$\Delta u(t) = \frac{0.5Lu.Lce}{Le(2Lce - GR.|ce(t)|} GE.e(t) + \frac{0.5Lu}{2Lce - GR.|ce(t)|} GR.ce(t) \qquad (2)$$

Con objeto de estudiar los efectos del incremento de la partición, para la misma estructura del FLC-PI incremental consideramos ahora tres etiquetas (n, p, z), tanto para las variables de entrada como para la salida. Se definen funciones de pertenencia triangulares para los conjuntos borrosos del error, cambio en el error y la salida del FLC como las descritas en la Figura 3 para el control, que mantienen un grado de solapamiento de 0'5.

La primera variación respecto al caso anterior se da en la dimensionalidad de la tabla de reglas (Tabla 1). Para las mismas variables de entrada, al aumentar su espacio es necesario reformular la base de reglas del controlador, que en este caso son 9, descritas en la Tabla 3.

Tabla 3. Reglas del controlador borroso PI incremental con tres etiquetas.

| | | *GE.e(t)* | | |
|---|---|---|---|---|
| | **U** | **P** | **Z** | **N** |
| *GR.ce(t)* | **P** | P | P | Z |
| | **Z** | P | Z | N |
| | **N** | Z | N | N |

La ecuación que describe simbólicamente el control es:

Δu = (Up - Un).Lu / Up + Uz + Un
Up = **or** (mín (Ep, CEp), mín (Ep, CEz), mín (Ez, CEp))
Uz = **or** (mín (Ep, CEn), mín (Ez, CEz), mín (En, CEp))
Un = **or** (mín (Ez, CEn), mín (En, CEz), mín (En, CEn))

Se ha aplicado la conectiva **or** propuesta por Lukasiewicz para la agregación de las reglas. Teniendo en cuenta estas ecuaciones, y sustituyéndolas en el algoritmo FM

para calcular la salida, se pueden establecer ocho zonas de control lineal en función de las relaciones entre las variables de entrada, resumidas a continuación (3). La función de control que se obtiene es no lineal.

***Zona I:*** $GR.|ce(t)| \le GE.|e(t)|$

$sg(e) = sg(ce)$

| | |
|---|---|
| $GR.\|ce(t)\| \le L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[GE.e(t) + 2GR.ce(t)]}{L + 2GR.\|ce(t)\|}$ |
| $GR.\|ce(t)\| \ge L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[2L - GE.e(t)]}{3L - 2GE.\|e(t)\|}$ |

$sg(e) \ne sg(ce)$

| | |
|---|---|
| $GR.\|ce(t)\| \le L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[GE.e(t) + 2GR.ce(t)]}{L + 2GR.\|ce(t)\|}$ |
| $GR.\|ce(t)\| \ge L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[2L - GE.e(t)]}{3L - 2GE.\|e(t)\|}$ |

***Zona II***: $GE.|e(t)| \le GR.|ce(t)|$

$sg(e) = sg(ce)$

| | |
|---|---|
| $GR.\|ce(t)\| \le L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[GE.e(t) + GR.ce(t)]}{L + 2GR.\|ce(t)\|}$ |
| $GR.\|ce(t)\| \ge L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[GE.e(t) + GR.ce(t)]}{3L - 2GE.\|e(t)\|}$ |

$sg(e) \ne sg(ce)$

| | |
|---|---|
| $GR.\|ce(t)\| \le L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[GE.e(t) - GR.ce(t)]}{L + 2GE.\|e(t)\|}$ |
| $GR.\|ce(t)\| \ge L - GE.\|e(t)\|$ | $\Delta u = \dfrac{Lu.[GE.e(t) + GR.ce(t)]}{3L - 2GR.\|ce(t)\|}$ |

(3)

# 4 RESULTADOS

Para comparar el efecto del incremento de la partición de entrada, establecemos unos parámetros de ajuste calculados por un método analítico de sintonía que se ha demostrado muy eficaz para este tipo de FLC simples. Este método asigna unos parámetros obtenidos a partir del comportamiento del controlador en el estado estacionario, haciendo tender a cero las variables en (2), que son función de los parámetros de sintonía de un regulador PI convencional [8], [9].

Se puede observar en las ecuaciones del control (3) que las ganancias del controlador fuzzy tienen distintos valores, pero en la mayoría de las zonas son mayores que las establecidas para el caso anterior de una partición menor. Equiparándolos a los coeficientes de un regulador PI convencional, los parámetros del controlador borroso varían entre los valores absolutos de:

$$0.5Kp \le GR.GU \le 3Kp \qquad 0.5Ki \le GE.GU \le 3Ki \qquad (6)$$

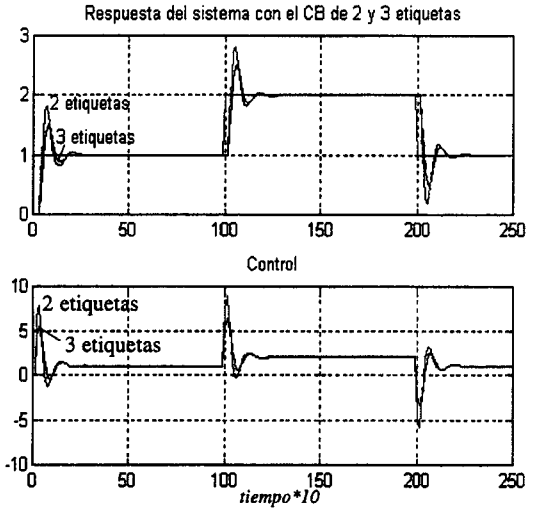
Por lo tanto, observando los coeficientes de las leyes de control (Tabla 4), una buena propuesta de parámetros iniciales será tomar como valores estacionarios:

$$GR = Kp.L/GU.\mathrm{Lu} \qquad GE = Ki.L/GU.\mathrm{Lu} \qquad (7)$$

Así se consigue que en los tramos donde las variables de entrada tienen igual signo se comporte como un PI lineal con ganancias *Kp* y *Ki*, y en las zonas donde tienen distinto signo incorpora la no linealidad haciendo más fuerte la acción de control, ya que las ganancias alcanzan un valor absoluto mayor.

La respuesta es ahora igual o menor que la que correspondería a un PI con esos parámetros, ya que en la mayoría de las zonas las ganancias del FLC son inferiores a las del PI. Pero si toman los valores extremos superiores, existen zonas en las que el controlador puede hacer al sistema oscilatorio o inestable.

En la Figura 4 se muestran los efectos de aumentar la partición de entrada, utilizando dos y tres etiquetas respectivamente para las variables de entrada.




| | |
|---|---|
| *Proceso* | $K = 1$; $Tp = 1$; $To = 0.2$ |
| *Controlador* | **PI** : $Kp = 0.3$; $Ki = 1.0$<br>**FLC**: $GU = 1$; Le = Lce = Lu = 15 |
| *Respuesta* | sobrelongación: 0.81 (2 etiquetas);<br>0.51 (3 etiquetas) |

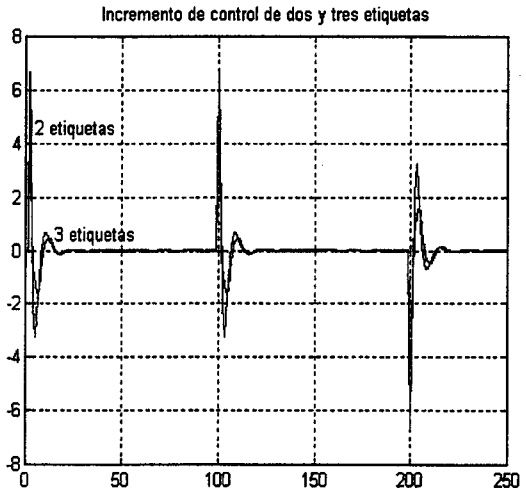



Figura 4: Respuesta del sistema con un FLC-PI con particiones de dos y tres etiquetas para las entradas.

Como se comprueba en la respuesta, mejora el transitorio, ya que la sobrelongación es menor manteniendo el tiempo de subida. Los incrementos de la acción de control también manifiestan esta diferencia.

Por eso, para los mismos parámetros, aumentar la partición supone mejorar la resolución del control ya que la salida del controlador es más precisa cuando los valores de entrada cambian, siendo más sensible a éstos cambios, y sin embargo mantiene un comportamiento lineal en el estacionario.

# 5 CONCLUSIONES

Es importante considerar que en un sistema de control fuzzy se puede actuar sobre cualquiera de sus elementos, pero no está clara la relación implícita que existe entre esos parámetros y la respuesta del sistema.

La partición de entrada tiene una gran influencia sobre la granularidad del control, lo que se puede observar empíricamente mediante un estudio cualitativo de la respuesta de un sistema al variarla.

Sin embargo, es interesante realizar también de forma analítica esta comparación, para controladores simples, donde su ley de control se puede expresar como una función no lineal de las entradas.

En este trabajo se ha enunciado matemáticamente la ley de control de un FLC con dos y tres etiquetas en función de las variables de entrada, lo que permite compararlas de forma rigurosa. El resultado confirma la heurística, ya que para los mismos parámetros de sintonía, aumentar la partición supone incrementar la resolución del control puesto que la salida del controlador es más precisa y sensible a cambios en las variables de entrada.

**Agradecimientos**

Este trabajo ha sido parcialmente financiado por el proyecto TAP94-0832-C02-01 de la CICYT.

**Referencias**

[1] J. Aguilar-Martín, (1993). Grado de recubrimiento de particiones borrosas. *Proc. III Congreso Español sobre Tecnologías y Lógica Fuzzy*, 137-144.

[2] M. Braae y D. A. Rutherford (1978). Fuzzy relations in a control setting. *Kybernetes*, vol. 7, n° 3, 185-188.

[3] A. Barreiro y R. Sanz (1993). Control y supervisión de procesos industriales con lógica borrosa. *Estudios sobre lógica borrosa y sus aplicaciones*, 69-94.

[4] J. J. Buckley y H. Ying (1989). Fuzzy controller theory: limit theorems for linear fuzzy control rules. *Automática*, vol. 25, 469-472.

[5] C. C. Lee (1990). Fuzzy logic in control systems: Fuzzy logic controller - Part I. *IEEE Trans. Syst. Man Cybern.*, vol. 20, n° 2, Mar./Apr., 404-418.

[6] A. Ollero y A. J. García-Cerezo (1989). Direct digital control, auto-tuning and supervision using fuzzy logic. *Fuzzy Sets Syst.*, vol. 30, 135-153.

[7] K.L. Tang, R.J. Mulholland (1987). Comparing fuzzy logic with classical controller design. *IEEE Trans. Syst. Man Cybern.*, vol. SMC-17, n. 6, 1085-1087.

[8] M. Santos (1994). Contribución a los Métodos de Sintonía de los Controladores Basados en la Lógica Borrosa. Tesis Doctoral, DIA, UCM.

[9] M. Santos, S. Dormido, A. P. de Madrid, F. Morilla, J.M. Cruz (1996). Tuning fuzzy logic controllers by classical techniques. *Lectures Notes in Computer Science*, Springer-Verlag, vol. 1105, 214-224.

[10] M. Santos, S. Dormido, J.M. de la Cruz (1996). Influence of the information processing in fuzzy logic controllers. *Proc. of the IPMU'96*, Vol. I, 81-86.

[11] P. Viljamaa y H. N. Koivo (1993). Tuning of multivariable fuzzy logic controller. *Proc. 2nd IEEE Int. Conf. on Fuzzy Systems*, San Francisco, 697-701.

# CONTROLADOR BORROSO MULTIVARIABLE COOPERATIVO-CORRECTOR

M. C.Rivero Nuñez P. Lasso Ariño F. Monasterio-Huelín

Email: crivero@robot.tat.upm.es fmh@robot.tat.upm.es

Universidad Politécnica de Madrid. E.T.S.I.Telecomunicación de Madrid. Cdad. Universitaria s/n, Madrid 28040

## Resumen

En este trabajo se aborda el problema de la descomposición de un sistema de Control borroso multivarible. para ello se propone la utilización de nociones en el marco de la Teoría de Juegos. Utilizando la noción de acoplo entre variables se realiza una partición del espacio de reglas en subsistemas que serán tratados como Jugadores, corrigiendo las acciones de los demás.

**Palabras Clave:** Control borroso multivariable, control borroso de procesos de polimerización.

## 1. Introducción.

Es casi obligado citar a Zadeh [1], Mamdani y Assilian [2] como los pioneros en el planteamiento de una nueva metodología de diseño de controladores. A diferencia de los métodos algebraicos (que llamaremos convencionales) esta nueva metodología (que data de principios de los 70), se fundamenta en un razonamiento lógico sobre variables lingüísticas, en el marco matemático iniciado por Zadeh en 1965. Se pretende con este método resolver problemas de control de procesos complejos, o mal definidos, utilizando para ello la experiencia humana. Desde entonces la teoría de control borroso se ha desarrollado profundamente, abordando todas (o casi todas) las cuestiones de la Teoría de Control: estabilidad, robustez, sintonización de los parámetros del controlador etc. [3]. Como era de esperar aparecieron también nuevos problemas. Abordaremos uno de ellos en este trabajo: la dimensionalidad del conjunto de reglas (o base de reglas) en los sistemas de control multivariable.

La mayor parte de la investigación de Controladores Borrosos se ha centrado en sistemas de una entrada y una salida, con controladores de tipo PID (o similares); en estos momentos podría decirse que esta clase de sistema se "comprende" bastante bien, en relación con las técnicas de diseño de controladores convencionales [4]. No ocurre lo mismo con los sistemas multivariables.

Takagi y Sugeno [5] proponen un modelo de razonamiento (modelo TSK), que consiste básicamente en descomponer el espacio de entrada de regiones borrosas y aproximar el sistema en cada región por un modelo simple. El modelo borroso total es una combinación de subsistemas interconectados de modelos más simples. Yager [6] unifica la idea de identificación de parámetros y estructuras. Mediante técnicas de programación no lineal resuelve el problema de la identificación de la estructura, es decir de la distribución del espacio de entrada entre los diferentes subsistemas de reglas.

Pero estos estudios se aplican fundamentalmente en la elaboración de modelos borrosos de sistemas, y no en el control. No obstante estas ideas pueden utilizarse en el diseño de controladores.

En [7] se utiliza una estructura jerárquica de sistemas de reglas, distribuyendo las variables entre los diferentes niveles, obteniendo así un conjunto de reglas completo con un número de reglas lineal en vez de exponencial. Cada nivel da una salida que es modificada por el siguiente. El razonamiento se realiza utilizando variables de diseño de controladores como por ejemplo la relación de pseudo-amortiguamiento, grado de oscilación etc., y las salidas son parámetros del controlador como por ejemplo la ganancia, expresada en términos porcentuales.

No pretendemos hacer una revisión de publicaciones sobre el tema, tan solo indicar que hay dos formas fundamentales de plantear el problema de los sistemas multivariables:

- dividiendo en regiones todo el espacio de entrada (más útil en modelado que en control)
- repartiendo las variables de entrada en una estructura jerárquica.

En el siguiente apartado proponemos una forma de razonamiento para la descomposición en una estructura jerárquica (no necesariamente vertical), diferente de la que se estudia en [7]. La idea fundamental consiste en hacer un reparto de las variables de entrada en un juego cooperativo que tenga en cuenta las relaciones causales entre las variables, de forma perecida a como se hace en [7], definiendo variables intermedias utilizadas porcentualmente, lo que hemos denominado modo corrector.

En el apartado 2 se describe un controlador borroso aplicado a un proceso de polimerización. En [8] se hace un estudio interesante sobre las principales diferencias entre la construcción de controladores borrosos y convencionales aplicados al control de procesos.

## 2. Controlador cooperativo-corrector.

En el diseño de controladores borrosos de sistemas multivariables se intenta que los conjuntos de reglas satisfagan ciertas propiedades: completitud, consistencia, continuidad e interacción [4]. Nos interesa especialmente el concepto de completitud, es decir que para cualquier vector de entrada al controlador borroso se dispare alguna regla con grado de verdad de la premisa distinto de cero.

Si un sistema multivariable está muy acoplado, es decir si cualquier salida $y_i \in Y=\{y_1,....,y_r\}$ depende de todas las entradas $U=\{u_1,...,u_s\}$, el controlador deberá tenerlo en cuenta de alguna manera. A simple vista parece razonable pensar que es suficiente con que la señal de control contemple este acoplo. Es decir, si lo que desea es diseñar un controlador realimentado, cada señal de control deberá ser función de todas las salidas del sistema y por lo tanto, la primera solución que a uno se le ocurre es diseñar un conjunto de reglas del controlador borroso bajo la forma general:

$$\left.\begin{array}{l}\textbf{IF } v_1 \text{ es } A_{1j1} \textbf{ AND } .... \textbf{ AND } v_n \text{ es } A_{njn} \\ \textbf{THEN } (u_1 \text{ es } B_{1j1},....,u_s \text{ es } B_{sjs})\end{array}\right] \quad (1)$$

donde :
$V=\{v_1,....,v_n\}$: conjunto de variables de entrada (error, incremento del error)

$L_{vi}=\{A_{i1},....,A_{ipi}\}$: conjunto de términos lingüísticos asociados a la variable de entrada $v_i$ **(ALTO-POSITIVO, NULO)**
$U=\{u_1,.....,u_s\}$: conjunto de variables de salida
$L_{ui}=\{B_{i1},....,B_{iqi}\}$: conjunto de términos lingüísticos asociados a la variable de salida $u_i$. Para que este conjunto de reglas del controlador borroso sea completo se necesitan $p=\prod_{i=1}^{n} p_i$ reglas.

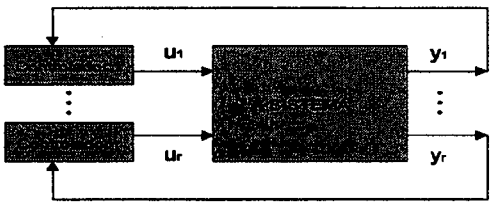


**Figura 1**

Cuando el sistema tiene muchas entradas y muchas salidas, el número total de reglas con la representación (1) puede ser excesivamente elevado, por lo que conviene descomponer el sistema de control en subsistemas (pero esto no puede hacerse de cualquier manera) que sigan satisfaciendo al menos la propiedad de completitud.
Cuando los acoplos internos del sistema son muy débiles, puede diseñarse un controlador completamente desacoplado (ver fig. 1), es decir, se realiza una partición del conjunto de entradas $V=\{v_1,.....,v_n\}$ y de salidas

$Y=\{y_1,.....,y_r\}$ en s clases, siendo s el número de entradas del sistema (o variables de control), y se diseña un controlador independiente para cada clase de la partición.

Cuando se conocen los acoplos fuertes y débiles este método de diseño puede mejorarse despreciando los débiles y teniendo sólo en cuenta los fuertes, realizando una partición del conjunto de entrada del sistema $U=\{u_1,.....u_s\}$, y diseñando controladores independientes para cada nueva clase.

En ambos casos es necesario conocer los acoplos internos del sistema, y esto no siempre es posible.

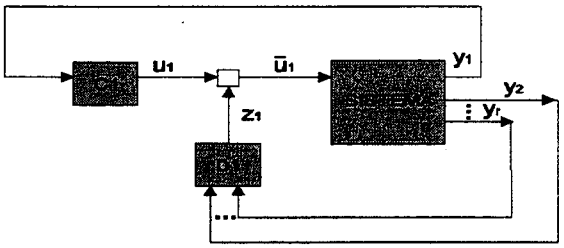


**Figura 2**

Esta técnica todavía puede ser mejorada compensando los acoplos débiles con señales de control aditivas (ver fig 2). Lo que se está haciendo en este caso es definir un conjunto de variables de control intermedias (que no necesitan ser las variables de estado), que llamaremos $Z=\{z_1,....,z_m\}$.

Lo que hemos hecho hasta aquí es una descripción de diferentes estructuras de control razonando alrededor del concepto de acoplo en términos de fuerte y débil. Pero esto no es más que un recurso discursivo, ya que por ejemplo la estructura de la figura 2 puede interpretarse en términos de acoplo lineal (C1) y acoplo no lineal (D1), como por ejemplo en el método del par calculado, en robótica [9]. En resumen, podríamos haber escogido cualquier concepto de carácter funcional para razonar

sobre él y descomponer el sistema de control en subsistemas de menor dimensión (las clases de la partición mencionadas anteriormente), puesto que lo que nos interesa en primer lugar es destacar el carácter estructural de la descomposición sin olvidar el aspecto funcional.

En los controladores borrosos es casi imprescindible definir esas variables intermedias (Z) con cierto sentido físico, ya que la metodología de diseño de las reglas exige un razonamiento lingüístico que relacione salidas (U) con entradas (V). Es decir, cuanto más abstractas sean estas variables intermedias más difícil será diseñar cada controlador parcial.

Las correcciones tampoco necesitan ser de carácter aditivo ($\bar{u}_1 = u_1+z_1$), sino que podrían ser (como hemos hecho en la aplicación del CSTR del apartado 3) de carácter porcentual ($\bar{u}_1 = u_1(1+z_1)$) para facilitar el razonamiento sobre el concepto "corrección". Así, por ejemplo, si un controlador (C1) decide "aumentar mucho la temperatura ($u_1$)", otro (D1) podría corregir esta acción con reglas del tipo "disminuye en un 20% la acción del otro controlador $z_1$".

Por otro lado, como puede observarse en la fig. 2, todavía no se ha hecho una descomposición satisfactoria en términos del nº de reglas ya que el controlador corrector que genera ($z_1$) sigue recibiendo muchas entradas. El número de entradas puede seguir siendo excesivo.

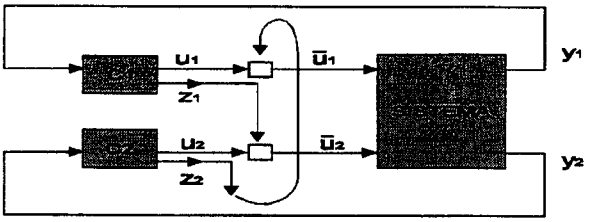



Figura 3

Para resolver esta cuestión, es decir, reducir el numero de reglas y que el conjunto de reglas siga siendo completo, proponemos en este trabajo la idea de un controlador cooperativo, que no será otra cosa que una manera de interpretar la interacción de los controladores entre sí para generar la acción final. La cooperación puede ser ciega o no informada cuando ninguno de los controladores haya sido informado de la acción de los demás (ver fg 3) o informada (ver fg 4) en caso contrario.

¿Bajo qué condiciones es posible descomponer el sistema de control en una estructura cooperativa-correctora que tenga en cuenta los acoplos intrínsecos del sistema?. La respuesta no es obvia.

Lo que queremos decir es que la noción de "acoplo" puede interpretarse en el marco de la lógica borrosa en términos de un juego cooperativo (o competitivo) entre diferentes controladores (o jugadores) que se reparten las variables de entrada. En la medida en que estén informados o no de las acciones de los demás, las correciones podrian ser más inteligentes.

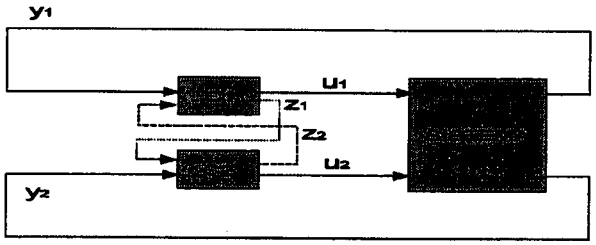



Figura 4

En el siguiente apartado analizamos una aplicación al problema de reactores de polimerización.

## 3. Aplicación a un reactor de polimerización

La idea de todo este trabajo parte de la aplicación que describimos más adelante. Se intentó diseñar un controlador que corrigiera la acción de otro que tiene un funcionamiento aceptablemente correcto, y que fue diseñado teniendo en cuenta un modelo simple de la planta.

La idea original fue no alterar substancialmente el sistema de reglas de este "buen" controlador, cuando se dispone de un nuevo modelo de la planta más complejo y preciso. Este nuevo modelo puede tener mayor dimensión, lo que obliga a pensar en la descomposición del sistema de control.

Vamos a aplicar algunas de las ideas anteriormente expuestas al problema de polimerización en un reactor continuo de tanque agitado (CSTR, Continuous Stirred Tank Reactor), mediante simulaciones en ordenador, con datos reales obtenidos de una importante empresa española del sector.

En la literatura de diseño de esta clase de sistemas se considera un CSTR con una camisa refrigerante por la que circula agua fría, ya que la reacción de polimerización es muy exotérmica, y se utiliza para el diseño de controladores, un modelo matemático [10] que considera una sola variable de control (el flujo de agua en la camisa) y una sola variable de salida (la temperatura dentro del reactor). Se diseña por tanto un Controlador de Temperatura. Este modelo supersimplificado ha sido utilizado para el diseño de controladores no lineales de todo tipo (linealización por realimentación de estados [11], etc.). Sin embargo los estudios teóricos y prácticos de modelado del proceso de polimerización muestran una gran variedad de comportamientos dinámicos [12] no contemplados en el modelo anterior. Por esta razón se ha tendido al diseño de controladores no basados en el modelo exacto de la planta (control adaptativo [13], borroso [14,15]. No obstante los modelos resultan útiles para realizar simulaciones y para analizar

cualitativamente algunas relaciones entre las variables de entrada, salida y de estado del sistema.

Para el desarrollo del modelo es necesario realizar un balance de masas y de energía a nivel macroscópico. Hemos desarrollado un modelo matemático del CSTR, que a diferencia del modelo comentado (que llamaremos modelo simple) no desprecia factores como la contracción de volumen en la conversión de monómero a polímero debido al cambio de estado (de líquido a sólido); se introduce la posibilidad de controlar independientemente el flujo de monómero y de disolvente, en concordancia con los estudios sobre la cinética de este tipo de procesos, consideramos una densidad variable dentro del reactor además de un control variable del flujo de salida (del producto).

Con estas pequeñas modificaciones el sistema de control tendría cuatro variables de control (flujo de monómero(FA), flujo de disolvente(Fe), flujo de agua de la camisa(Fj), y flujo de salida del reactor(Fo)) y tres variables de salida (volumen dentro del reactor(V), temperatura dentro del reactor(T) y concentración del polímero(B)). Diseñamos así un Controlador de Temperatura y de Volumen. En la Figura 5 se muestra el esquema de control estudiado en [16], siguiendo las directrices expuestas al principio de este apartado. En la Figura 6 se indica el conjunto de términos lingüísticos para las variables de entrada y para las de salida, y en la Tabla 2 el conjunto de reglas utilizado. El controlador secundario es un controlador corrector del Controlador de Volumen (Cooperación no informada),. mientras que el Controlador de Volumen del Primario coopera informativamente con el controlador de Temperatura del Primario.

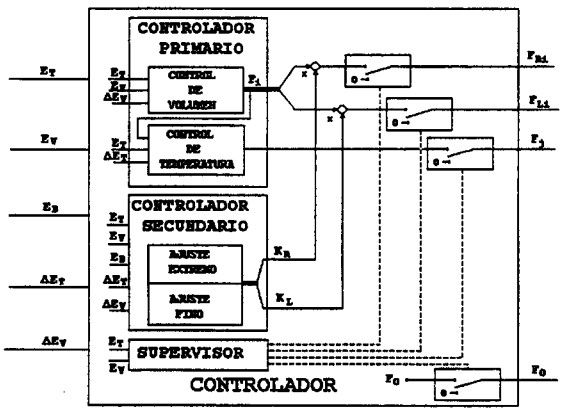



**Figura 5**

El bloque borroso secundario se ha dividido en dos conjuntos de reglas con vistas a cumplir dos objetivos: reducir los transitorios y aumentar la precisión (reducir la desviación con respecto al punto de trabajo definido en régimen estacionario). En la Figura 5 se denomina a estas dos partes "**ajuste extremo**" y "**ajuste fino**". (También se ha incluido en la figura un tercer bloque general denominado "**supervisor**", que en realidad viene a ser una serie de "alarmas" que cortan determinados flujos de control del reactor cuando las variables de entrada al controlador están fuera del rango normal).

En las Gráficas 1 a 3 se presentan resultados de simulación comparativos del funcionamiento de los diversos subsistemas de control del controlador completo de la Figura 5.

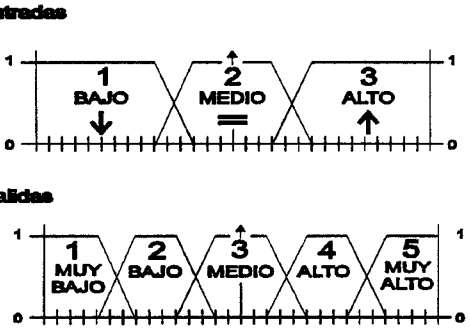
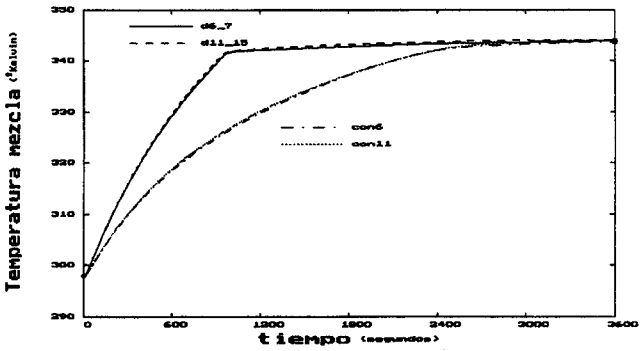
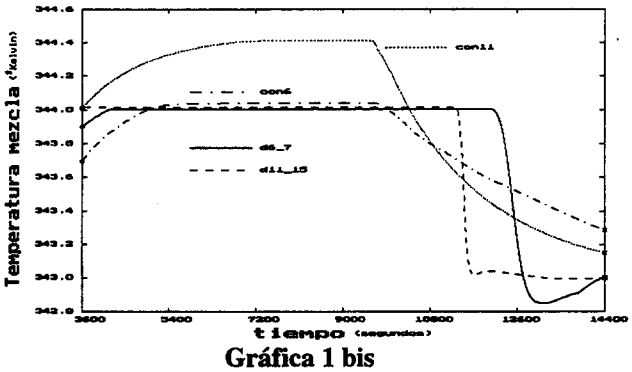



**Figura 6**




**Gráfica 1**

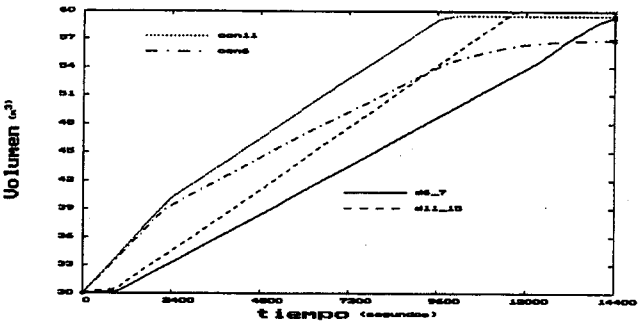



**Gráfica 1 bis**

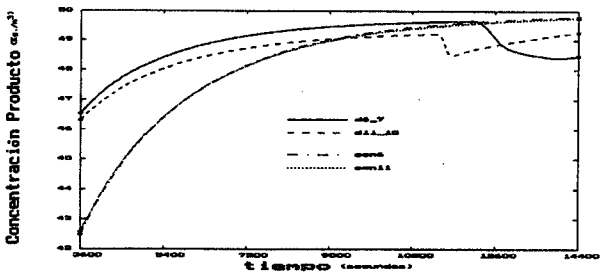
Los autores en [17] están estudiando con mayor profundidad que en [16] el acance teórico comentado en el apartado anterior, aplicándolo a esta clase de reactores.

Tabla 1: Condiciones de las simulaciones

| | PUNTO DE PARTIDA | OBJETIVO |
|---|---|---|
| T | 25°C (298 °K) | 70°C (343°K) |
| V | 30 $m^3$ | 60$m^3$ |
| $C_B$ | 0 $Kg/m^3$ | 50 $Kg/m^3$ |




**Gráfica 2**




**Gráfica 3**

## CONCLUSIONES

El diseño de controladores borrosos multivariables ha sido poco estudiado en la literatura especializada. Hemos propuesto una forma de razonamiento para la descomposición del espacio de entrada, que permite el diseño de las reglas de control basándose en la experiencia y en el conocimiento cualitativo de las relaciones causales entre las variables de entrada. El diseño pretende ser gradual, es decir, se diseñan controladores de menor dimensión y se sintonizan convenientemente; la ampliación de la dimensionalidad se hace diseñando nuevos controladores que cooperan con los que han sido ya probados, corrigiendo porcentualmente sus acciones. Hemos denominado a esta forma de descomposición Control Cooperativo-Corrector, aplicandolo a un reactor de polimerización (CSTR). Todavía hay muchas cuestiones abiertas. La sistematización de diseño con este enfoque está siendo estudiada en la actualidad [17].

## 5. REFERENCIAS

[1] L. A. Zadeh, "Outline of a New Approach to de Analysis of Complex Systems and Decision Processes", IEEE Tr. Syst, Man an Cybern, SMC-3, n°1, pp. 28-44, 1973.

[2] E. H. Mamdani y S. Assilian, "An experiment in Lingüistic Synthesis with a Fuzzy Logic Controller", Int. J. Man-Machine Studies, vol.7, pp. 1-3, 1975.

[3] R.R. Yager y D.P. Filev, **Essentials of Fuzzy Modeling and Control**, John Wiley and Sons, Inc., 1994

[4] D. Driankov, H.Hellendoorn y M. Reinfrunk, **An Introduction to Fuzzy Control**, Springer- Verlag, 1993

[5] T. Takagi y M. Sugeno, "Fuzzy Identification of systems and its applications to modeling and control", IEEE Tr. on Syst., Man and Cybern., Vol 15, pp. 116-132, 1985

[6] R.R. Yager y D.P. Filev, "Unified Structure and Parameter Identification of Fuzzy Models", IEEE Tr. on Sys., Man and Cybern, Vol. 23, n° 4, Jul/Aug 1993, pp. 1198- 1205

[7] G.V.S.Raju y J.Zhou, "Adaptive Hierarchical Fuzzy Controller ", IEEE Tr. on Syst, Man and Cybern, Vol. 23, n° 4, Jul/Aug 1993, pp. 973-980

[8] J. A. Bernard, "Use of a rule-Based System for Process Control", IEEE Control Systems Magazine, Oct. 1988, pp. 3-13

[9] M. W. Spong y M. Vidyasagar, **Robot Dynamics and Control**, John Wiley and Sons, 1989.

[10] R. Aris, **Análisis de Reactores**, Ed. Alhambra 1973

[11] J. Alvarez, J. Alvarez y E. Gonzalez, " Global nonlinear Control of a Continuous Stirred Tank Reactor", Chemical Engineerig Science, Vol. 44, n° 5, pp. 1147-1160, 1989

[12] A. D. Schmidt, A. B. Clinch y W. H. Ray, "The Dynamic Behavior of Continuous Polymerization Reactors-III. An Experimental Study of Multiple Steady States in Solution Polymerization", Chemical Engineering Science, Vol. 39, n° 3, pp. 419-432, 1984

[13] C Kiparissides, E. Sidiropoulou, S. Voutetakis y N.B.Satineidis, "A comparative Study of LQC, DMC and Extended STR Control Strategies"

[14] B. Roffel y P.A. Chin, "Application of Fuzzy Logic in the Control of Polymerization Reactors", Control Eng. Practice, Vol. 1, n° 2, pp. 267-272, 1993

[15] J Song y S. Park, " A Fuzzy Dynamic Learning Controller for Chemical Process Control", Fuzzy Sets and Systems, Vol. 54, 1993, pp.121-133

[16] P. Lasso, "Diseño de un controlador para un reactor químico CSTR de polimerización ", PFC curso 1996-1997.

[17] C. Rivero, "Modelado y control borroso cooperativo-corrector de un reactor de polimerización CSTR (Aplicación polipropileno)", Tesis doctoral en preparación.

# TABLA 2

```
PeriodoC=2
TempObj = 343
VolObj= 60
CBObj =50

ControlV:BORROSO
Entradas:
(-50,20)(-1,-1,-0.04,-0.02)(-0.04,-0.02,0.05,0.1)(0.05,0.1,1,1)
(-60,5)(-1,-1,-0.1,-0.05)(-0.1,-0.05,0.05,0.1)(0.05,0.1,1,1)
(-0.4,0.4)(-1,-1,-0.4,-0.2)(-0.4,-0.2,0.2,0.4)(0.2,0.4,1,1)
Salidas:
(-1,0.3)(-1,-1,-0.8,-0.6)(-0.8,-0.6,-0.3,-0.1)(-0.3,-
0.1,0.1,0.3)(0.1,0.3,0.6,0.8)(0.6,0.8,1,1)
Reglas:
1,1,0 - 5
1,2,1 - 5
1,2,2 - 4
1,2,3 - 3
1,3,1 - 2
2,1,1 - 5
2,1,2 - 4
2,1,3 - 4
2,2,1 - 4
2,2,2 - 3
2,2,3 - 2
2,3,1 - 2
2,3,2 - 2
3,1,1 - 3
3,1,2 - 2
3,1,3 - 2
3,2,1 - 2
ELSE - 1

ControlT : BORROSO
Entradas:
(-50,20)(-1,-1,-0.04,-0.02)(-0.04,-0.02,0.05,0.1)(0.05,0.1,1,1)
(-1.6,4)(-1,-1,-0.5,-0.25)(-0.5,-0.25,0.1,0.2)(0.1,0.2,1,1)
(-1,0.3)(-1,-1,-0.3,-0.1)(-0.3,-0.1,0.1,0.3)(0.1,0.3,1,1)
Salidas:
(-1,0.3)(-1,-1,-0.8,-0.6)(-0.8,-0.6,-0.3,-0.1)(-0.3,-
0.1,0.1,0.3)(0.1,0.3,0.6,0.8)(0.6,0.8,1,1)
Reglas:
1,0,0 - 1
1,3,3 - 2
2,1,0 - 1
2,2,1 - 3
2,2,2 - 3
2,2,3 - 4
2,3,1 - 4
2,3,2 - 4
3,1,1 - 4
ELSE - 5

Control2:BORROSO
Entradas:
(-20,20)(-1,-1,-0.1,-0.05)(-0.1,-0.05,-0.025,0.025)(-
0.025,0.025,0.05,0.1)(0.05,0.1,1,1)
(-1.6,4)(-1,-1,-0.5,-0.25)(-0.5,-0.25,0.1,0.2)(0.1,0.2,1,1)
(-60,5)(-1,-1,-0.1,-0.05)(-0.1,-0.05,-0.025,0.025)(-
0.025,0.025,0.05,0.1)(0.05,0.1,1,1)
(-0.4,0.4)(-1,-1,-0.4,-0.2)(-0.4,-0.2,0.2,0.4)(0.2,0.4,1,1)
(-4,4)(-1,-1,-0.5,-0.25)(-0.5,-0.25,0.25,0.5)(0.25,0.5,1,1)
Salidas:
(-0.5,0.5)(-1,-1,-0.8,-0.6)(-0.8,-0.6,-0.3,-0.1)(-0.3,-
0.1,0.1,0.3)(0.1,0.3,0.6,0.8)(0.6,0.8,1,1)
(-1,0.5)(-1,-1,-0.8,-0.6)(-0.8,-0.6,-0.3,-0.1)(-0.3,-
0.1,0.1,0.3)(0.1,0.3,0.6,0.8)(0.6,0.8,1,1)
Reglas:
1,1,3,0,0 - 5,1
1,1,2,3,0 - 5,1
1,1,2,2,0 - 5,1
1,1,4,2,0 - 5,1
1,1,4,3,0 - 3,1
1,2,4,2,0 - 5,1
1,2,4,3,0 - 3,1
1,1,1,1,0 - 5,2
1,1,1,2,0 - 5,1
1,2,1,2,0 - 5,1
1,2,1,1,0 - 5,3
1,3,1,1,0 - 5,3
1,3,1,2,0 - 5,3
1,3,1,3,0 - 5,2
4,2,1,1,0 - 1,5
4,2,1,2,0 - 1,5
4,3,1,1,0 - 1,5
4,3,1,2,0 - 1,5
4,2,4,2,0 - 1,1
4,2,4,3,0 - 1,1
4,3,4,2,0 - 1,3
4,3,4,3,0 - 1,2
2,2,2,2,1 - 4,3
2,2,3,2,1 - 4,2
3,2,2,2,1 - 3,4
3,2,3,2,1 - 2,2
2,2,2,2,2 - 4,4
2,2,3,2,2 - 4,2
3,2,2,2,2 - 2,4
3,2,3,2,2 - 2,2
2,2,2,2,3 - 4,4
2,2,3,2,3 - 3,2
3,2,2,2,3 - 2,4
3,2,3,2,3 - 2,3
ELSE - 3,3


INICIALIZA
 KdM = FlujoLM/FlujoAM
 ErrV1=cuant(Vol-VolObj,0.04)
 ErrT1=cuant(Temp-TempObj,0.2)
 VEVolcuant = 0
 VETcuant = 0
 ECBcuant = cuant(CB-CBObj,0.2)
 FlujoSal = 0
 FlujoCam = 0
 TiempoV = Tiempo
 TiempoT = Tiempo
 CerradoFj = 0
 CerradoFli = 0
 CerradoFai = 0
END.
BEGIN
 ETcuant = cuant(ET,0.2)
 EVolcuant = cuant(EVol,0.04)
 IF (Tiempo - TiempoT) >= (2*PeriodoC) THEN
  TiempoT = Tiempo
  VETcuant = ETcuant - ErrT1
  ErrT1 = ETcuant
 END
 IF (Tiempo - TiempoV) >= (10*PeriodoC) THEN
  TiempoV = Tiempo
  VEVolcuant = EVolcuant - ErrV1
  ErrV1 = EVolcuant
  ECBcuant = cuant(ECB,0.2)
 END
 (FlujoNormA) = ControlV(ETcuant,EVolcuant,VEVolcuant)
 (KAi,KLi) = Control2 (ETcuant,VETcuant,EVolcuant,VEVolcuant,ECBcuant)
 FA1 = FlujoAM*(1 + FlujoNormA)
 FlujoA = FA1*(1 + KAi)
 FlujoL = FA1*KdM*(1 + KLi)
 (FlujoNormJ) = ControlT(ETcuant,VETcuant,FlujoNormA)
 FlujoCam = FlujoCamM*(1 + FlujoNormJ)
 IF (Temp > 348) OR ((Temp >= 346) AND (CerradoFai = 1)) THEN
  FlujoA = 0
  CerradoFai = 1
 ELSIF (Temp < 335) OR ((Temp <= 338) AND (CerradoFj = 1)) THEN
  FlujoCam = 0
  FlujoL = 0
  CerradoFj = 1
  CerradoFai = 0
 ELSE
  CerradoFj = 0
  CerradoFai = 0
 END
 IF (Vol > 65) OR ((Vol >= 62) AND (CerradoFli = 1)) THEN
  FlujoA = 0
  FlujoL = 0
  FlujoSal = FlujoSalM
  CerradoFli = 1
 ELSIF (Vol < 45) OR ((Vol <= 50) AND (FlujoSal = 0)) THEN
  FlujoSal = 0
  CerradoFli = 0
 ELSE
  CerradoFli = 0
  FlujoSal = FlujoSalM
 END
END.
```

Reglas añadidas

Reglas modificadas

Etapa de entrada: cuantificación y muestreo con periodo variable

SUPERVISOR

# CONTROLADOR FUZZY PARA LA EVITACIÓN DE OBSTÁCULOS EN UN ROBOT MÓVIL NO HOLÓNOMO

**Juan Pedro Uribe**
Dpto. de Inteligencia Artificial
IKERLAN
Centro de Investigaciones Tecnológicas
e-mail: jpuribe@ikerlan.es

**Joseba Urzelai**
Dpto. de Inteligencia Artificial
IKERLAN
Centro de Investigaciones Tecnológicas
e-mail: robot_movil@ikerlan.es

## Resumen

En la presente comunicación se describe el diseño y desarrollo de un sistema de navegación, basado en información sensorial, que permite a un robot móvil no holónomo realizar evitación de obstáculos, utilizando la información de su entorno obtenida por un cinturón de sensores de ultrasonidos. El control del robot no precisa de ninguna información previa de su entorno, sino que se adapta a éste utilizando la información inmediata de los sensores de ultrasonidos y la disponible a partir de un sistema de memoria espacial. El controlador que permite generar las consignas a los motores del robot ha sido desarrollado basándose en el concepto de percepción general y se han utilizado principios de lógica difusa para su realización. La plataforma sobre la que se han realizado los tests del sistema de navegación es un prototipo de robot móvil desarrollado en IKERLAN VEAII provisto de una alta maniobrabilidad capaz de evolucionar de forma holónoma y no holónoma.

**Palabras Clave**: Robot móvil, Navegación Local, Memoria espacial, Lógica Borrosa.

## 1 DESCRIPCIÓN DEL ROBOT MÓVIL

El objetivo básico del proyecto en el que se enmarca el sistema de navegación que nos ocupa es construir y gobernar un vehículo provisto de un brazo manipulador capaz de operar de forma autónoma y teleoperada, para su posible utilización en aplicaciones de telemanipulación en entornos interiores y exteriores.

La realización de este proyecto permite la búsqueda de soluciones técnicas y científicas novedosas en el campo de la robótica móvil, en conjunción con la telemanipulación, ofreciendo una plataforma que posibilita testear los diferentes subsistemas que intervienen en él.

El prototipo desarrollado está dividido en tres partes claramente diferenciadas: una estación de control (responsable de los aspectos de control que no necesitan respuesta en tiempo real), un brazo manipulador embarcado en el vehículo (un robot PUMA 560 instalado en la parte frontal) y el vehículo. El conjunto vehículo-brazo está provisto de una serie de sensores (ultrasonidos, cámaras de visión, telémetro láser) que le proveen de información de su entorno de evolución y le permiten actuar ante eventos imprevistos (obstáculos en el entorno, cambios de pendiente del terreno, etc.).

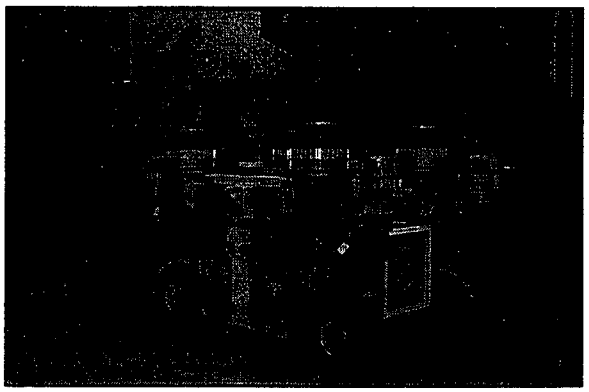


Figura 1

El vehículo móvil, tal como se aprecia en la figura 1, es de base pseudo-rectangular y tiene unas dimensiones de 1400 x 800 mm. Está compuesto por tres niveles diferentes:

El sistema de locomoción está basado en cuatro ruedas accionadas independientemente en tracción y dirección. A la hora de desarrollar este subsistema se ha primado el hecho de que el robot deba evolucionar en entornos no sólo planos, sino que también debe ser capaz de subir y bajar pendientes, evolucionar en terrenos irregulares, etc.

El control de trayectorias del vehículo permite a éste realizar desplazamientos en cualquier dirección, así como

llevar a cabo las maniobras necesarias para posicionarse en cualquier configuración accesible de su entorno de evolución.

Los sensores de que dispone el vehículo son, por una parte, un cinturón de 24 sensores de ultrasonidos que proveen información que es utilizada por el controlador para detectar y evitar obstáculos imprevistos y realizar seguimientos de contornos. Por otro lado, dispone de un telémetro láser que es utilizado para localizar al vehículo y para modelizar espacios libres por los que puede evolucionar el vehículo.

Como ya se ha mencionado, se ha planteado el estudio y desarrollo de técnicas de navegación basada en información sensorial. Dichas técnicas permiten al vehículo evolucionar en su entorno realizando trayectorias sin colisión, adaptando la trayectoria preprogramada a los posibles obstáculos imprevistos, realizando una evitación de éstos o efectuando planificaciones de movimientos teniendo en cuenta las sucesivas modelizaciones del entorno que llevan a cabo los sistemas de sensorización incorporados al vehículo.

## 2 PERCEPCIÓN GENERALIZADA Y PERCEPCIÓN DISTRIBUIDA

Es bien conocida la limitación de resolución direccional que tienen los sensores de ultrasonidos. A pesar de que pueden determinar con gran exactitud la distancia a la que se encuentra un objeto, éste puede localizarse en cualquier punto dentro del cono del sensor. Además, el ángulo del cono depende de la superficie del objeto, de la distancia y del ángulo de incidencia, por lo tanto, resulta difícil modelizar un entorno a fin de guiar un robot en él. Tanto el concepto de percepción general como el de percepción distribuida evitan modelizar el entorno y tratan de construir una serie de vectores que suministren información acerca del mismo.

En el caso de la percepción distribuida, dichos vectores se denominan *vectores distribuidos de percepción*, puesto que existe un vector por cada lado del vehículo. En la percepción general, en cambio, existe un único vector que se denomina *vector de percepción general* [2][3][8].

Para componer cada vector distribuido de percepción $p_d$, se aproxima la nube de puntos sensorizados del lado en cuestión mediante una recta. El vector distribuido de percepción $p_d$ tendrá la dirección perpendicular a la recta calculada anteriormente y su módulo será igual a la percepción máxima del lado.

$$p_d = \frac{d_{max} - d_{d,min}}{d_{max} - d_{min}} \tag{1}$$

donde $d_{min}$ y $d_{max}$ son la distancia mínima y máxima, respectivamente, que puede medir el sensor. $d_{d,min}$ es la distancia medida por el sensor que ha detectado el obstáculo más cercano. $p_d$ está limitado entre 0 y 1, por lo tanto:

$$p_d = \begin{cases} 0 & si \quad d_{d,min} > d_{max} \\ 1 & si \quad d_{d,min} < d_{min} \end{cases} \tag{2}$$

De esta manera se componen los cuatro vectores distribuidos de percepción del robot rectangular.

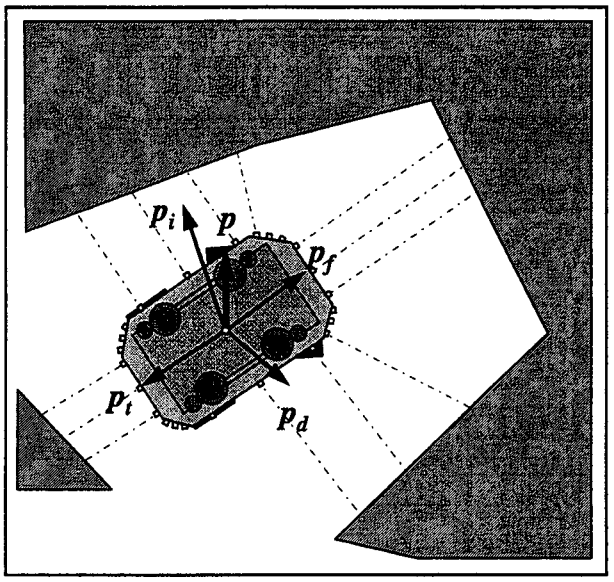



Figura 2. Representación de los vectores distribuidos de percepción y el vector de percepción general.

El vector de percepción general $p_g$ se compone mediante los vectores distribuidos de percepción. La dirección corresponde a la suma vectorial ponderada de los vectores distribuidos y el módulo es igual al valor máximo de los módulos de percepción distribuida:

$$p_g = p_{d,max} \frac{\sum k_d p_d}{\left|\sum p_d\right|} \tag{3}$$

El término $k_d$ de la suma vectorial realiza una ponderación de los vectores distribuidos en función de las dimensiones del vehículo, puesto que éstas influyen en gran medida en su comportamiento.

# 3 MEMORIA ESPACIAL

En algunas situaciones, el vehículo pierde la percepción respecto al entorno debido a la mala orientación (excesivo ángulo de los sensores respecto al obstáculo). Sin embargo, suponiendo obstáculos estáticos, el robot puede conocer la posición de los mismos valiéndose de la información obtenida en movimientos anteriores, es decir, el robot podría memorizar las percepciones anteriores (desplazadas odométricamente) para suplir las pérdidas de sensorización.

La nube de puntos que percibe el vehículo a través de los sensores es aproximada por una recta; esta recta se caracteriza por una pendiente $m = tg\ \alpha$ y el punto más próximo p. Tal y como se puede apreciar en la figura 3, un movimiento del vehículo respecto a la pared equivale a una traslación con giro de la recta y respecto del vehículo.

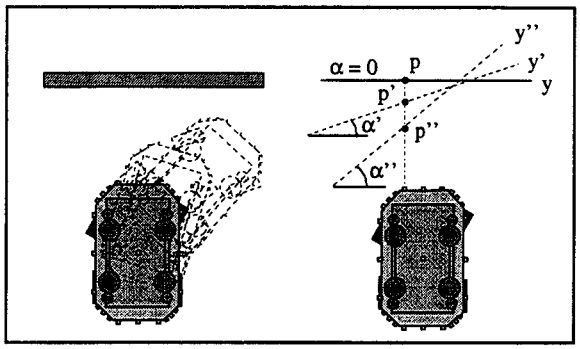



Figura 3: Representación de la información de los sensores mediante una recta caracterizada por **m** y **p**.

Por lo tanto, el robot memorizará los parámetros $(m_i,p_i)$ de cada iteración para calcular los nuevos puntos en la siguiente iteración. La percepción real $(m_i,p_i)$ tiene siempre prioridad sobre la percepción memorizada $(m_{im},p_{im})$. El robot obtiene los puntos memorizados calculando la intersección de los haces de los sensores con la recta definida por $(m_{im},p_{im})$.

Existe una serie de restricciones referentes a los parámetros $(m_{im},p_{im})$ memorizados. En el caso de la percepción frontal, cuando la pendiente memorizada $m_{im}$ supera una pendiente máxima $m_{max}$, la información puede ser trasladada al lateral y se eliminan los puntos de la memoria, ya que en caso contrario el robot seguiría indefinidamente una recta ficticia, siendo imposible ejecutar curvas convexas. Otra de las restricciones corresponde al caso en el que la pendiente $m_{im}$ es menor que el ángulo del cono del sensor (este ángulo depende de la superficie del objeto). Siendo $\{p_{1l}, p_{2l}, \ldots p_{nl}\}$ los puntos correspondientes a los n sensores del lado l, se realiza el siguiente análisis:

Si para todo $p_{il}$ se tiene que $p_{il} < p_{i+1,l}$ ó $p_{il} > p_{i+1,l}$ en ese caso se trata de una pared recta que deberían detectar los sensores, por lo tanto, se eliminan los puntos de la memoria.

Si por el contrario se tiene que $p_{il} < p_{jl} > p_{kl}$ ó $p_{il} > p_{jl} < p_{kl}$, siendo $i < j < k$, en ese caso se trata de un obstáculo indetectable para el robot (debido al excesivo ángulo de incidencia) a pesar de que la pendiente resultante se encuentre dentro del cono de sensorización; por lo tanto, se mantienen los puntos de la memoria.

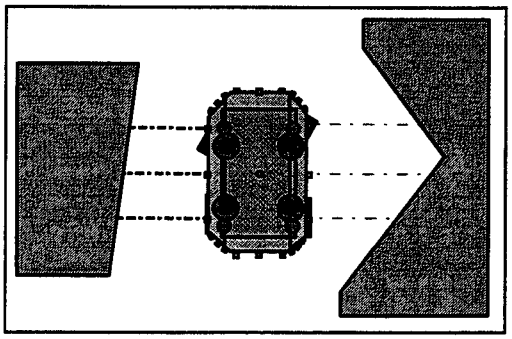


Figura 4: En la izquierda se muestra una situación en la que los puntos de la memoria deben ser eliminados, ya que la pared debería ser detectada por los sensores. En la situación de la derecha, en cambio, el robot no puede detectar la pared, ya que el ángulo que forman los dos muros es excesivamente cerrado; en este caso, el robot debe guiarse por los puntos de la memoria.

Otro de los aspectos importantes de la memoria espacial es el desplazamiento de información de la memoria frontal a las memorias laterales, de adelante a atrás en el sentido del movimiento, a medida que cambia la orientación del obstáculo respecto al vehículo.

Sean $\{p'_{1l}, p'_{2l}, \ldots p'_{nl}\}$ los puntos de los *n* sensores laterales obtenidos a partir de los puntos de los *m* sensores frontales $\{p_{1f}, p_{2f}, \ldots p_{mf}\}$, y sean $\{p_{1l,m}, p_{2l,m}, \ldots p_{nl,m}\}$ los puntos de los sensores laterales memorizados por el robot. Se utiliza el operador *mínimo* para establecer los puntos laterales definitivos:

Para todo *i*, $p_{il} = \mathbf{min}\ (p'_{il}, p_{il,m})$ (4)

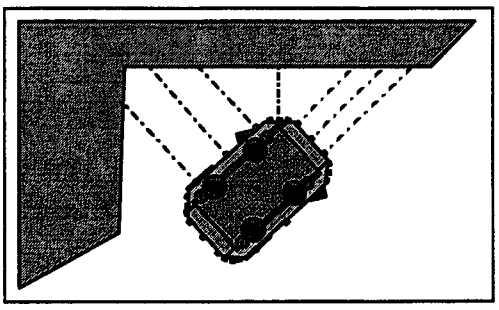


Figura 5: Gracias al desplazamiento de memoria el robot es capaz de detectar la pared frontal con sus sensores laterales antes de que se encuentre dentro de su cono de sensorización

El desplazamiento de memoria resulta muy útil en situaciones tales como muros que forman un ángulo recto (figura 5). Cuando la información de todos los sensores laterales corresponda a la memoria desplazada desde el frontal, se eliminará la información de la memoria de los sensores frontales, ya que dicha información se encuentra íntegramente en los sensores laterales.

Finalmente, cabe resaltar que gracias a la memoria espacial se consigue minimizar el número de situaciones en el que el robot pierde completamente la percepción y, aun más, el desplazamiento de percepción del frontal a los laterales permite suministrar información incluso antes de que los sensores laterales sean capaces de detectar cualquier obstáculo.

# 4 CONTROLADOR FUZZY

Cuando el robot móvil ejecuta una misión calculada por un planificador de trayectorias, a menudo se encuentra con obstáculos en su camino. En estos casos, el robot móvil debe evitar dichos obstáculos y volver a un punto de la trayectoria calculada por el planificador.

El objetivo de este controlador es posicionarse sobre un punto objetivo final, evitando los posibles obstáculos que pudieran surgir en el camino. El robot móvil se guía por la información obtenida mediante los sensores de ultrasonidos y conoce en todo momento la posición y orientación del punto final.

En este artículo sólo se va a desarrollar el controlador implementado con percepción general.

## 4.1 ENTRADAS

La representación de las entradas se basa en el vector de percepción generalizada y el ángulo del robot móvil respecto al punto final. Las variables de entrada del controlador son el módulo de la percepción general $p$, el ángulo de la percepción general $\alpha$ y el ángulo del robot móvil respecto al punto final $\beta$. Las variables borrosas se denominan *modP*, *angP* y *angGoal* respectivamente.

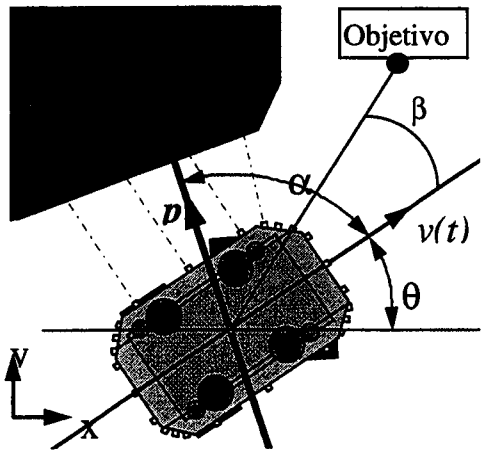



Figura 6: Representación de los antecedentes y consecuentes del controlador de percepción general.

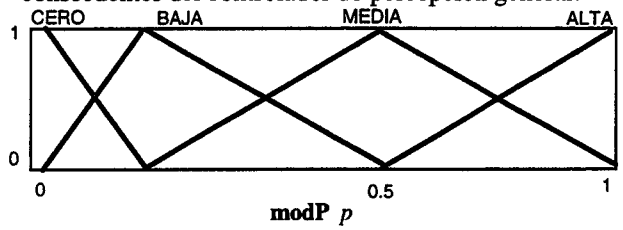



Figura 7: Adjetivos del módulo de la percepción general.

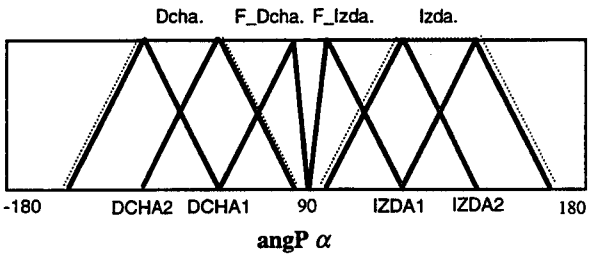



Figura 8: Adjetivos del ángulo de la percepción general.

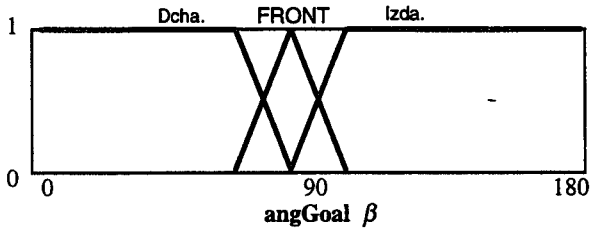



Figura 9: Adjetivos del ángulo del ángulo del robot respecto al punto final.

## 4.2 SALIDAS

Las variables de salida son la variación de la orientación del vehículo $\theta$ y la aceleración del mismo $a$. Las correspondientes variables borrosas se denominan *incTeta* y *Acel* respectivamente.

Las funciones de pertenencia de los adjetivos de las variables de salida borrosas son del tipo *singleton*.

La variable *incTeta* representa la variación de la orientación del vehículo (no de las ruedas). Las funciones de pertenencia de los adjetivos se muestran en la figura 10.

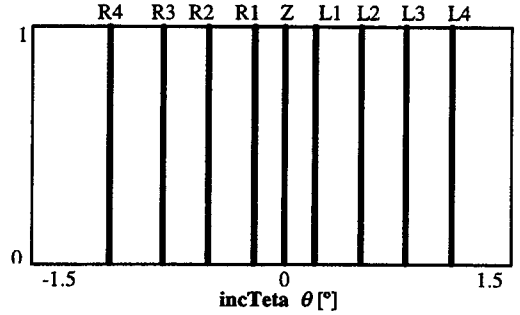



Figura 10: Adjetivos singleton para el cambio de orientación del vehículo.

La variable *Acel* se define mediante cinco adjetivos que permiten acelerar y frenar el vehículo y un adjetivo EM_BREAK que frena bruscamente el robot en situaciones de peligro. La figura 11 muestra las funciones de pertenencia de los adjetivos.

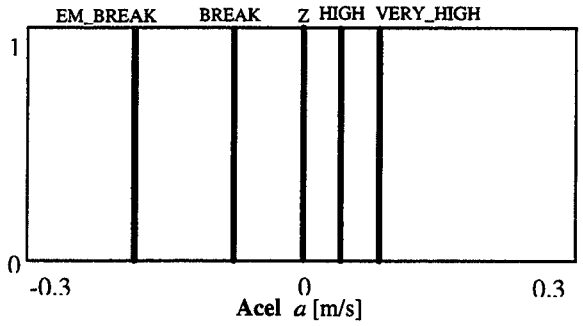



Figura 11: Adjetivos singleton para la aceleración del robot móvil autónomo.

## 4.3 BASE DE REGLAS

El objetivo de este módulo es desplazar el robot móvil desde su posición inicial hasta el punto final de destino. El punto final ejerce una fuerza de atracción sobre el robot móvil y los posibles obstáculos que pudieran aparecer en el camino ejercen fuerzas de repulsión. Se trata de conseguir un equilibrio entre estas fuerzas para que el robot móvil pueda desplazarse por los espacios libres hasta el punto final.

Partiendo de esta idea, se han definido dos grupos de reglas para el control de la orientación. El primero representa la fuerza de atracción que ejerce el punto final sobre el robot móvil. El otro grupo de reglas representa la fuerza de repulsión ejercida por los obstáculos. Estas reglas obligan al robot móvil a alejarse de los obstáculos que percibe.

El control de velocidad se hace en el grupo de las reglas de repulsión, puesto que si el robot móvil no detecta ningún obstáculo debe tender a acelerar hasta alcanzar la velocidad máxima. El robot móvil debe frenar cuando tiene que realizar maniobras de evitación pasando cerca de los obstáculos.

Cabe destacar que las reglas de repulsión tienen prioridad sobre las de atracción, ya que una colisión del robot móvil contra un obstáculo sería intolerable.

La evaluación de reglas se realiza por el método de producto y combinación de suma y la defuzzificación mediante el método de los centroides, en ambos casos.

### a. Control de la Dirección

Esta base de reglas consta de veintidós reglas y tiene como entradas el módulo de la percepción general $p$, el ángulo de la percepción general $\alpha$ y el ángulo del robot respecto al punto final $\beta$. Genera como salida variaciones de la dirección del vehículo.

Consta de dos grupos de reglas, el primero hace tender el robot móvil hacia el objetivo cuando el camino está libre para lo cual garantiza la lejanía de los obstáculos (modP) y gira hacia el objetivo siempre que en esa dirección no haya obstáculos (angP y angGoal).

En la tabla 1, las reglas van precedidas por:

IF modP is NOT ALTA AND modP is NOT MEDIA AND.

o también se puede poner:

IF modP is CERO OR modP is BAJA AND.

Tabla 1: Base de reglas que controla la dirección del vehículo.

| $\alpha/\beta$ | DCHA | FRONT | IZDA |
|---|---|---|---|
| NOT DCHA | R4 | -- | -- |
| NOT F_DCHA | -- | Z | -- |
| NOT F_IZDA | -- | Z | -- |
| NOT IZDA | -- | -- | L4 |

El segundo grupo de reglas resuelve el conflicto entre la atracción del objetivo y la repulsión de un obstáculo en ese camino. La estrategia es la siguiente: Si hay un obstáculo enfrente (FRONT), aunque el objetivo también lo esté, prima evitar el obstáculo. Si está a un lado y algo al frente (DCHA1, IZDA1) también hay que evitarlo aunque no haya que girar tanto. Por último, si está a un lado y el robot lo está sobrepasando (DCHA2) se tiende hacia el objetivo. Dentro de estos tres grupos se establecen grados de maniobra en función de la distancia al obstáculo.

### b. Control de la Velocidad

Esta base consta de doce reglas y tiene como entradas el módulo de la percepción general $p$ y el ángulo de la percepción general $\alpha$. Tiene como salida el valor de la aceleración del vehículo.

La velocidad está limitada entre $v_{max}$ y $v_{min}$, donde $v_{min}$ se calcula como:

$$v_{min} = \text{Min}\ (0.05,\ 1\text{-}p) \qquad (5)$$

La idea es que, si no hay obstáculo o éste está alejado, el robot acelera, y si está más cerca: se para si el obstáculo está situado frontalmente, frena si está delante pero algo a la derecha y acelera si está a un lado (cambia un poco dependiendo si la percepción es MEDIA o ALTA).

Tabla 2: Base de reglas que controla la velocidad del vehículo (VH: Very High H: High, Z: Zero, B:Break ,E: Emergency).

| α/p | CERO | BAJA | MEDIA | ALTA |
|---|---|---|---|---|
| DCHA2 | VH | VH | VH | VH |
| DCHA1 | VH | VH | H | B |
| F_DCHA | VH | VH | Z | E |
| F_IZDA | VH | VH | Z | E |
| IZDA1 | VH | VH | H | B |
| IZDA2 | VH | VH | VH | VH |

# 5 RESULTADOS Y CONCLUSIONES

Después de sintonizar los adjetivos mediante algoritmos genéticos [2], los resultados en el simulador son muy satisfactorios y los de la plataforma real muy prometedores (se acaba de empezar a realizar pruebas).

Al igual que en el seguidor de paredes [1], el siguiente paso es el de implementar un controlador con la percepción distribuida con el objeto de conseguir mayor suavidad en el movimiento y resolver con mayor elegancia los problemas en entornos con alta densidad de obstáculos.

Resaltar la robustez aportada por el sistema de memoria espacial.

En la figura 12 se muestra un ejemplo típico de evitación de obstáculos:

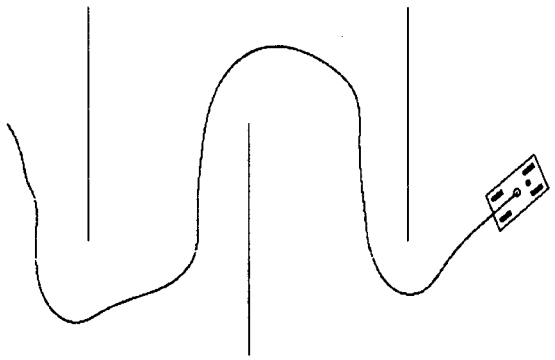


Figura 12: Evitación de obstáculos.

A continuación en las figuras adyacentes se muestra, por un lado, el módulo de EVITACIÓN cayendo en un mínimo local y, a continuación, una estrategia mixta, de tal forma que cuando el objetivo queda ATRÁS se habilita el modulo de SEGUIMIENTO de paredes hasta que se encara el objetivo.

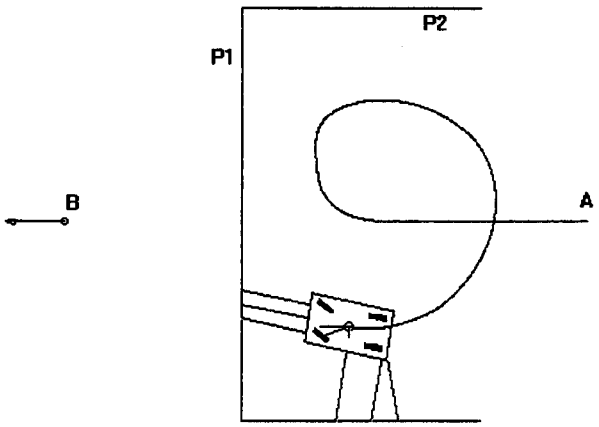



Figura 13: Mínimo Local en evitación de obstáculos.

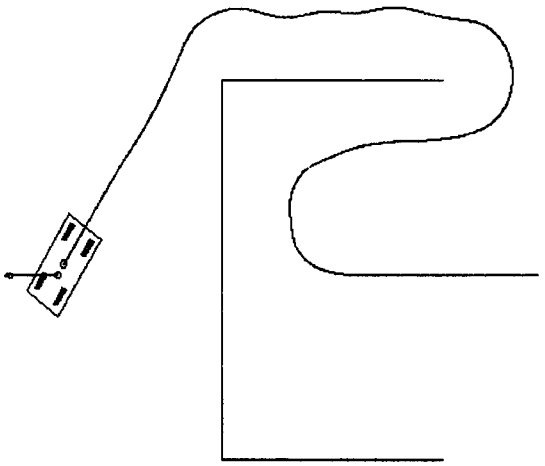


Figura 14: Estrategia mixta de evitación de obstáculos y seguimiento de paredes.

**Referencias**

[1] Borenstein J., Koren Y., The Vector Field Histogram - Fast Obstacle Avoidance for Mobile Robots. IEEE Transactions on Robotics and Automation, vol. 7, n° 3, June 1991, pp. 278-288.

[2] Braunstingl R., Mujika J., Uribe J.P., *A Wall Following Robot With A Fuzzy Logic Controller Optimized By a Genetic Algorithm.* FUZZ-IEEE/IFES'95 FUZZY ROBOT COMPETITION, YOKOHAMA, MARCH, 1995.

[3] Braunstingl R., Sanz P. and Ezkerra J.M., *Fuzzy Logic Wall Following of a Mobile Robot Based on the Concept of General Perception.* ICAR'95, Seventh

International Conference on Advanced Robotics, Catalonia, September 1995.

[4] Crowley J.L., *Navigation for an Intelligent Mobile Robot.* IEEE Journal of Robotics and Automation, vol. RA-1, n° 1, march 1985, pp. 31-41.

[5] Kathib O., *Real-Time Obstacles Avoidance for Manipulators and Mobile Robots.* The International Journal of Robotics Research, vol.5, n° 1, Spring 1986, pp. 90-98

[6] Li W., *Fuzzy Logic-based reactive behaviour of an autonomous mobile system in unknown environments.* Engng applic. Artif. Intell., 1994, 11(8), pp. 743-760.

[7] Reignier P. *Pilotage Reactif d'un Robot Mobile. Etude du Lien entre la Perception et l'Action.* Ph. D. Thesis, Institut National Polytechnique de Grenoble, December 1994.

[8] Urzelai J., Uribe J.P., Ezkerra J.M., *Fuzzy Controller for wall-following with a non holonomous mobile robot.* FUZZ-IEEE/IFES'97, BARCELONA, July, 1997.

# DESCOMPOSICION JERARQUICA DE CONTROLADORES BORROSOS MULTIOBJETIVO BASADA EN METACONOCIMIENTO

**L. Magdalena, F. Monasterio y C. Rivero**
Univ. Politécnica de Madrid
llayos@mat.upm.es

## Resumen

La definición de controladores multivariable en los que la actuación a realizar está condicionada por la consecución de múltiples objetivos acoplados supone un problema de difícil solución. En el caso de los controladores borroso, la obtención del conocimiento necesario para el diseño de un controlador tal puede resultar compleja debido a la interacción existente entre los distintos objetivos.

El presente trabajo aborda la posibilidad de construir controladores borrosos capaces de resolver problemas de cierta complejidad, utilizando técnicas jerárquicas y aplicando controladores de gran sencillez (como son los controladores PD borrosos). La vía para conseguir la definición de la estructura jerárquica es el aprovechamiento de los diferentes niveles de conocimiento disponible sobre el problema.

## 1 Introducción

El diseño de controladores borrosos multivariable representa en la actualidad un tema abierto dentro de la teoría de control borroso. La definición de controladores en los que la actuación a realizar está condicionada por la consecución de múltiples objetivos acoplados supone un problema de difícil solución. En el caso de los controladores borroso, la obtención del conocimiento necesario para el diseño de un controlador tal puede resultar compleja debido a la interacción existente entre los distintos objetivos. En muchos casos, existe el conocimiento experto relativo a la consecución de los objetivos, pero este conocimiento no es fácilmente trasladable a reglas.

Una posible vía para conseguir resolver este tipo de problemas sería la descomposición horizontal (no jerárquica) del problema en base a los diferentes objetivos, y la posterior agregación de las soluciones individuales. No parece sin embargo la solución más adecuada ya que no aborda el desacoplo de los objetivos, manteniendo probablemente el mismo nivel de dificultad a la hora de realizar la extracción de conocimiento. Por otro lado, las interacciones que aparecen entre las soluciones individualizadas pueden suponer la inviabilidad de este tipo de enfoque ya que la agregación de soluciones individualizadas no supone la creación de una solución global.

El presente trabajo aborda la posibilidad de redefinir este tipo de problemas mediante la jerarquización del controlador y el desacoplo de los objetivos. El procedimiento a seguir será la redefinición de los objetivos de control basada en metaconocimiento, similar al que podría emplear el controlador humano en una situación tal, para posteriormente alcanzar dichos objetivos empleando sencillos algoritmos de control (tipo PID). La idea es pues la extracción, en forma de metaconocimiento, de las relaciones existentes entre los diferentes objetivos, para posteriormente aplicar este metaconocimiento al desacoplo de los objetivos, obteniendo de esta forma controladores independientes dirigidos a obtener objetivos desacoplados. La introducción del metaconocimiento en el sistema se realizará en forma de información contextual, es decir, a través de la definición de las funciones de normalización a aplicar a las variables de entrada y salida del controlador borroso. El ajuste de las funciones de normalización como mecanismo de aprendizaje ya ha sido analizado anteriormente [5], en este caso nos centraremos en su aplicación como vía de introducción de metaconocimiento en el controlador.

El planteamiento se realizará en base a un problema clásico como es el control del carro con el péndulo invertido. Se trata de un sistema inestable y de fase no mínima ([3]), que presenta lo que se podrían de-

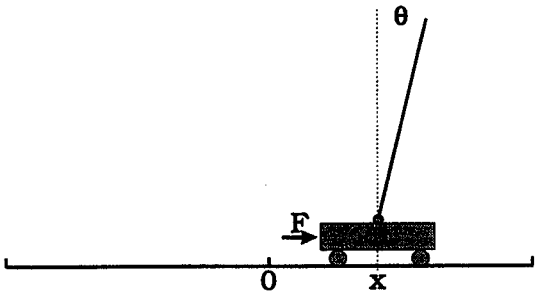



Figura 1: Carro y péndulo invertido

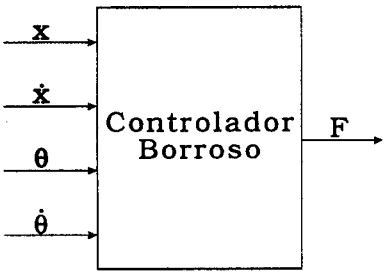



Figura 2: Controlador borroso

nominar comportamientos no intuitivos, en los cuales la acción de control parece oponerse a la consecución directa del objetivo. Se iniciará el desarrollo con una breve descripción de dicho problema.

Debe quedar claro que dicho ejemplo no se aborda con el objetivo de conseguir un controlador óptimo para el sistema, sino con el de mostrar que es posible obtener un controlador aceptable basándonos sólo en el conocimiento disponible. Este mismo problema ha sido abordado con otros criterios como el de control adaptativo estable ([6]), o la utilización de mecanismos de aprendizaje ([2, 4, 5]), no pretendiéndose mejorar los controladores obtenidos en estos trabajos sino simplemente analizar las posibilidades del enfoque.

## 2 El problema del carro y el péndulo

El problema del carro y el péndulo (Figura 1) es un clásico problema de control que ha sido empleado en numerosas ocasiones para analizar las capacidades de aprendizaje y las propiedades de diferentes sistemas de control [1]. Consiste en un carro con ruedas situado sobre un rail de dimensión predefinida. Sobre el carro se coloca un péndulo invertido conectado al carro mediante un rodamiento. El objetivo es doble: mantener el péndulo en equilibrio y centrar el carro en el rail. La consecución de este doble objetivo se debe realizar mediante la aplicación al carro de fuerzas longitudinales. La Figura 2 ilustra la estructura del controlador borroso a aplicar, mostrando sus variables de entrada $(x, \dot{x}, \theta, \dot{\theta})$ y la variable de actuación $(F)$.

Las ecuaciones que gobiernan el comportamiento dinámico del carro son las siguientes:

$$\ddot{x} = H_1(\dot{x}, \theta, \dot{\theta}, F) = \frac{F - \mu_c sign(\dot{x}) + \tilde{F}}{M + \tilde{m}} \quad (1)$$

y

$$\ddot{\theta} = H_2(\dot{x}, \theta, \dot{\theta}, F) = -\frac{3}{4l}(\ddot{x}\cos\theta - g\sin\theta + \frac{\mu_p \dot{\theta}}{ml}), \quad (2)$$

donde

$$\tilde{F} = ml\dot{\theta}^2 \sin\theta + \frac{3}{4}m\cos\theta(\frac{\mu_p\dot{\theta}}{ml} - g\sin\theta) \quad (3)$$

y

$$\tilde{m} = m(1 - \frac{3}{4}\cos^2\theta). \quad (4)$$

Las dimensiones habitualmente empleadas en el problema se describen en la Tabla 1

## 3 Sistemas multiobjetivo y metaconocimiento

El caso del carro con el péndulo es un claro ejemplo de sistema multivariable en el que conviven diferentes objetivos acoplados. Nos encontramos por un lado con la necesidad de equilibrar el péndulo, y por otro con la de desplazar el carro a una determinada posición. Ambos objetivos son atacados mediante una única variable de control, la Fuerza aplicada al carro, y por tanto es imposible abordarlos de forma independiente. No es pues posible la descomposición directa del problema de control en dos subproblemas que aborden la consecución individual de cada uno de los objetivos a partir de la información correspondiente a las distintas variables de estado del sistema (posiciones de carro y péndulo). Sin embargo, trataremos de alcanzar esa descomposición mediante la adecuada aplicación del conocimiento disponible. La idea fundamental es intentar utilizar un mecanismo similar al que aplicaría un controlador humano en las mismas circunstancias. En este caso abordaremos el diseño de una estructura jerárquica, relacionando los diferentes objetivos con las variables de estado correspondientes, y redefiniendo dichos objetivos de acuerdo con el conocimiento disponible.

Iniciaremos el análisis por la determinación de los distintos objetivos, y de las variables de estado del sistema que determinan el grado de consecución de dichos objetivos. Cada uno de los controladores borrosos en que descompongamos el sistema estará encargado de la consecución de un objetivo, y trabajará en base a

Tabla 1: DIMENSIONES Y PARÁMETROS DEL SISTEMA

| Símbolo | Descripción | Valor |
|---|---|---|
| $x$ | Posición del carro | [-1.0,1.0] m |
| $\theta$ | Angulo del péndulo | [-0.3,0.3] rad |
| $F$ | Fuerza aplicada al carro | [-10,10] N |
| $g$ | Gravedad | 9.8 m/s$^2$ |
| $l$ | Longitud del péndulo | 1 m |
| $M$ | Masa del carro | 1.0 kg |
| $m$ | Masa del péndulo | 0.1 kg |
| $\mu_c$ | Fricción carro-rail | 0.0005 N |
| $\mu_p$ | Fricción del rodamiento del péndulo | 0.000002 kg m$^2$ |

las variables de estado que determinan su grado de obtención. Posteriormente trataremos de obtener las relaciones que determinan la interacción entre los diferentes objetivos. La definición de las salidas de cada uno de los controladores se realizará en base a las interrelaciones y acoplamientos existentes entre los diferentes objetivos del sistema, o lo que es los mismo, de acuerdo con los efectos que produzcan las acciones aplicadas para alcanzar un cierto objetivo sobre el grado de cumplimiento de los restantes objetivos. Esta información constituirá una especie de metaconocimiento, clave para la adecuada descomposición del problema.

Para ilustrar las ideas planteadas volveremos al ejemplo del carro con el péndulo, analizando los distintos objetivos presentes y las relaciones que aparecen entre ellos. En este caso el objetivo primero sería situar el carro en cierta posición, y por tanto las variables que describen el cumplimiento del objetivo son las correspondientes al carro $(x, \dot{x})$. El segundo objetivo sería equilibrar el péndulo, lo que se mediría en función de las variables relativas a él $(\theta, \dot{\theta})$.

Analicemos ahora más profundamente cada uno de los dos objetivos. El mantenimiento del péndulo en una determinada posición se basa en la existencia de un punto de equilibrio (la posición vertical definida por el ángulo 0) y en la posibilidad de aplicar una adecuada acción que devuelva al péndulo a dicho punto de equilibrio. Cuando la desviación es positiva, la fuerza aplicada debe tener ese mismo signo para producir un desplazamiento negativo del péndulo. Si aislamos este objetivo, podremos conseguirlo aplicando un simple controlador PD. El efecto de aplicar únicamente este controlador sería el de equilibrar el péndulo, perdiendo todo control sobre el carro que iría vagando sobre el rail hasta alcanzar uno de sus extremos.

De igual forma, el control del carro se podría llevar a cabo mediante un controlador PD, aunque en este caso sería el péndulo el que quedaría fuera de control.

Abordemos a continuación la obtención del metaconocimiento que nos permita la adecuada descomposición del problema. El planteamiento sería el de conseguir que esos objetivos, no independientes, fueran redefinidos de tal forma que el cumplimiento de un único objetivo conllevara la consecución indirecta de los restantes. El problema es la necesidad de tener un conocimiento adicional sobre estos objetivos, que nos permita establecer esa conexión que haga que cumplir un objetivo nos lleve a cumplir los restantes.

Estudiemos la mecánica que emplearía un humano para resolver este problema y tratemos de obtener el metaconocimiento requerido. El problema plantea dos situaciones bien diferenciadas: cuando el objetivo de carro y péndulo requieren acciones de igual signo, y cuando ambos objetivos requieren acciones contrapuestas.

Cuando el carro esta desplazado respecto de la posición objetivo y el péndulo se inclina hacia el exterior, el péndulo requiere la aplicación de una fuerza orientada hacia el exterior, mientras que el carro precisa la acción contraria. Un ser humano generalmente realizaría la primera de las acciones (fuerza hacia el exterior) para hacer cambiar la orientación del péndulo quedando este inclinado hacia el interior, y de esta forma hacer coincidir ambos objetivos, requiriendo tanto péndulo como carro una fuerza hacia el interior.

Cuando ambos elementos precisan una fuerza de igual sentido, la acción del ser humano sería la de aplicar una fuerza en el sentido preciso, tratando de conseguir que la intensidad de la misma no hiciera cambiar al péndulo de orientación, lo que nos devolvería a la situación anterior.

Si replanteamos el objetivo en base a la solución que se acaba de describir, tendríamos que el objetivo seria inclinar el péndulo hacia la dirección objetivo del carro, conseguido lo cual se aplicaría una fuerza que llevara al carro hacia su objetivo, sin cambiar la orientación del péndulo.

Podríamos expresar esta idea mediante reglas con la siguiente estructura:

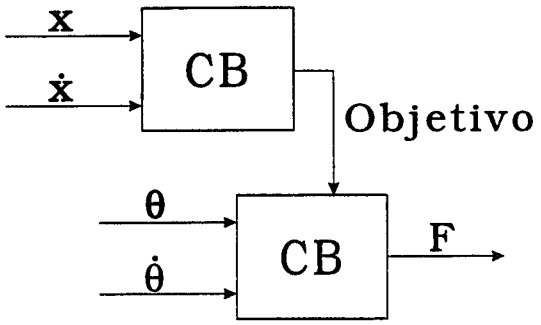



Figura 3: Controlador borroso jerarquizado

**Si** la *posición del carro* es *positiva*
**se requiere** una *fuerza negativa*
**lo que se corresponde con** una *inclinación del péndulo negativa.*

que viene a representar la relación existente entre los objetivos de carro y péndulo. Si pretendo centrar el carro, deberé aplicarle una fuerza en cierto sentido, el cual está relacionado con determinada orientación del péndulo.

La cuestión siguiente sería la de averiguar si el objetivo marcado para el péndulo es compatible con este objetivo redefinido del carro, y la respuesta es si. Es posible encontrar una posición de equilibrio en la que el péndulo se mantenga inclinado mientras el carro se desplaza bajo la acción de una fuerza de igual sentido a la orientación del péndulo. Podemos por tanto redefinir el objetivo del controlador que trabaja en función de $(x, \dot{x})$, expresando como su objetivo la definición de cual es la posición en que deseamos situar al péndulo. El controlador que trabaja sobre condiciones del péndulo se encargará de mantener al péndulo en dicho punto de equilibrio.

La Figura 3 ilustra como un primer Controlador Borroso que trabaja sobre las variables relativas al carro se encarga de definir el punto de trabajo del péndulo. A continuación, un segundo Controlador Borroso calcula la fuerza que debe aplicarse al carro para mantener al péndulo en el punto predeterminado.

## 4 Ajuste de contexto mediante metaconocimiento

En la sección anterior hemos analizado la redefinición de los objetivos a conseguir por los controladores, en la presente trataremos de analizar la incorporación de dichos objetivos en el sistema. Los objetivos manejados son básicamente situaciones definidas en el espacio de estados o en el espacio de las variables de actuación, hacia las cuales tratamos de llevar al sistema. Cada controlador situado en un determinado nivel de la estructura jerárquica podrá generar como salidas tanto objetivos para los niveles inferiores de la estructura jerárquica, como valores de las variables de actuación.

Desde el punto de vista de la teoría de control, podríamos considerar que estos objetivos representan puntos de consigna para los controladores situados en los niveles inferiores de la estructura jerárquica. Desde el punto de vista de los sistemas basados en conocimiento, podemos interpretar que se trata de información contextual que redefine el significado de los objetivos predefinidos. De esta forma, cada controlador borroso contiene un conjunto de reglas y de funciones de pertenencia normalizadas (un algoritmo de control) que le permiten alcanzar su objetivo predefinido. La información procedente de los niveles superiores es empleada para modificar las funciones de normalización aplicadas tanto a las variables de entrada como a las de salida, modificando de esta forma el significado real del algoritmo borroso.

En el caso del carro, el nivel superior de la estructura jerárquica estará compuesto por el controlador que trabaja en variables del carro y su salida será la definición de un cierto valor objetivo para la posición del péndulo (y posiblemente un valor objetivo para la fuerza). El nivel inferior recibirá las salidas del nivel superior y tratará de mantener al péndulo en la posición prefijada.

## 5 Resultados

Se presentan en esta sección los resultados obtenidos en la definición de un controlador borroso jerarquizado aplicado al control del carro con péndulo invertido. Partimos de los rangos de definición de las variables que han sido empleados normalmente en otros estudios de aplicación de control borroso al problema del carro con el péndulo ([2]).

Tabla 2: RANGOS DE VARIACIÓN

| Símbolo | Descripción | Valor |
|---|---|---|
| $x$ | Posición del carro | [-1.0,1.0] m |
| $\dot{x}$ | Derivada de la Posición | [-1.0,1.0] m |
| $\theta$ | Angulo del péndulo | [-0.3,0.3] rad |
| $\dot{\theta}$ | Derivada de la Posición | [-1.0,1.0] m |
| $F$ | Fuerza aplicada al carro | [-10,10] N |

Las salidas del nivel superior se obtienen en base a un controlador PD, haciendo que el valor del ángulo objetivo ($\theta_0$) sea la salida de dicho controlador. Su base de reglas (Tabla 3) se ha tomado directamente de la bibliografía ([7]), y es la correspondiente a un PD borroso. Las etiquetas empleadas son {-L,-M,-S,Z,S,M,L}

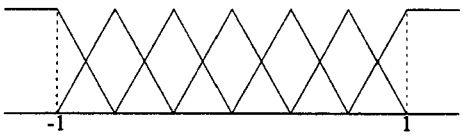



Figura 4: Funciones de pertenencia normalizadas.

que se corresponden con los valores {*Negativo grande, Negativo medio, Negativo pequeño, Cero, Positivo pequeño, Positivo medio, Positivo grande*} definidos como conjuntos borrosos triangulares y simétricos, distribuidos uniformemente en el intervalo normalizado de variación de la correspondiente variable (Figura 4).

Tabla 3: REGLAS DE CONTROL ($CB_1$)

| $\dot{x}$ \ $x$ / $\theta_0$ | -L | -M | -S | Z | S | M | L |
|---|---|---|---|---|---|---|---|
| -L | L | L | L | L | M | S | Z |
| -M | L | L | M | M | S | Z | -S |
| -S | L | M | S | S | Z | -S | -M |
| Z | L | M | S | Z | -S | -M | -L |
| S | M | S | Z | -S | -S | -M | -L |
| M | S | Z | -S | -M | -M | -L | -L |
| L | Z | -S | -M | -L | -L | -L | -L |

Tabla 4: REGLAS DE CONTROL ($CB_2$)

| $\dot{\theta}$ \ $\theta$ / $F$ | -L | -M | -S | Z | S | M | L |
|---|---|---|---|---|---|---|---|
| -L | -L | -L | -L | -L | -M | -S | Z |
| -M | -L | -L | -M | -M | -S | Z | S |
| -S | -L | -M | -S | -S | Z | S | M |
| Z | -L | -M | -S | Z | S | M | L |
| S | -M | -S | Z | S | S | M | L |
| M | -S | Z | S | M | M | L | L |
| L | Z | S | M | L | L | L | L |

Para la obtención de los parámetros de ajuste realizamos un reducido análisis basado en el estudio de comportamiento en base a un conjunto de condiciones iniciales. Diez posiciones iniciales distintas distribuidas uniformemente en torno a $\theta = 0$ son utilizadas. Cada base de conocimiento es probada durante 60 segundos arrancando desde cada una de las posiciones iniciales. Un proceso parcial de simulación (60 segundos) es detenido si las variables del sistema superan los rangos predefinidos ($x = [-1, 1]$ m, $\theta = [-0.3, 0.3]$ rad). Las condiciones aplicadas son: $\theta(0) = 0.01$ rad con $x(0) = 0.01, 0.1$ and 0.5 m, y $\theta(0) = 0.15$ rad con $x(0) = 0.1$ y 0.5 m; junto con sus valores simétricos $(-\theta, x)$. La evaluación se basa en una medida de calidad calculada a partir de la posición del carro

$$P = \frac{1}{10 \times 6000} \sum_{10init.} \sum_{6000ciclos} \exp(-|x|). \quad (5)$$

Aplicando esta medida de calidad y trabajando con diferentes valores de ángulo objetivo se han obtenido los resultados presentados en la Figura 5. En esta figura se representa la evaluación obtenida por diferentes controladores, en función del factor de escala aplicado para obtener el ángulo que se marcan como objetivo del controlador situado en el segundo nivel. Los factores están comprendidos en [0,0.3] con escalones de 0.025. Un factor de escala de 0.25 indica que la salida normalizada ([-1,1]) del PD borroso situado en el primer nivel, es convertida a un valor situado en el rango [-0.25,0.25]. Esto significa que el ángulo de consigna ($\theta_0$) tomará valores en dicho rango. El valor de la fuerza aplicada se obtiene mediante un segundo PD borroso con entradas $\theta$ y $\dot{\theta}$, y cuya salida es la fuerza (3). Es importante destacar el signo opuesto de ambos PD. Mientras que en el primero, una posición negativa del carro da lugar a una inclinación positiva del péndulo, en el segundo, una inclinación negativa del péndulo genera una fuerza de igual signo.

Empleando como factor de escala 0.225, lo que equivale a un rango de variación de $\theta_0$ comprendido en el intervalo [-0.225,0.225], se han obtenido los resultados presentados en las figuras 6 y 7. Estos resultados corresponden a la evolución de carro y péndulo, partiendo de las condiciones iniciales $x = 0.5$ y $\theta = 0.15$. Como se puede comprobar, el controlador no alcanza una estabilización total del sistema, entrando en un ciclo límite en el que las máximas desviaciones alcanzadas son un desplazamiento máximo del carro de 3 cm respecto de su objetivo, y un desplazamiento máximo del péndulo de 0.018 rad (aproximadamente 1 grado). Estos resultados son mejorables en base a otras técnicas de diseño, pero no debe perderse de vista que el proceso seguido ha sido el de emplear conocimiento intuitivo y no técnicas de mayor complejidad. En cualquier caso se obtiene un comportamineto similar al que logra un humano abordando este tipo de problema, ya que por lo general un humano no alcanza la estabilización total sino una cierta oscilación. Otra cuestión interesante de resaltar es el hecho de que una gran franja de valores máximos para $\theta_0$ hubiera generado resultados similares, como puede deducirse de la figura 5 a partir de la amplia meseta superior que presenta la evaluación. En consecuencia, el planteamiento intuitivo de orientar el péndulo en la dirección hacia la que queremos llevar el carro, es válido y fácilmente ajustable.

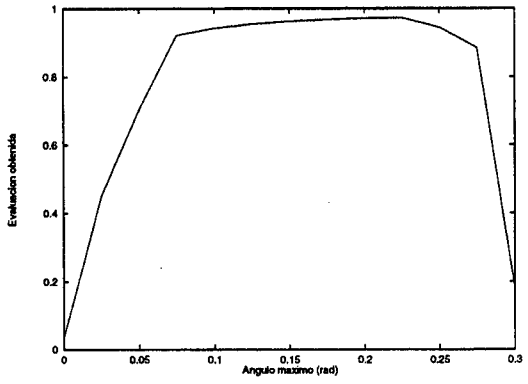



Figura 5: Evaluación con diferentes ángulos de ajuste.

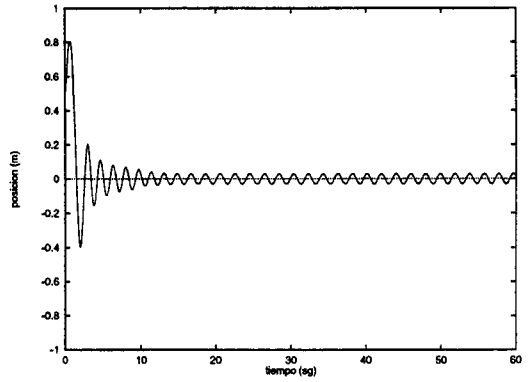



Figura 6: Trayectoria del carro.

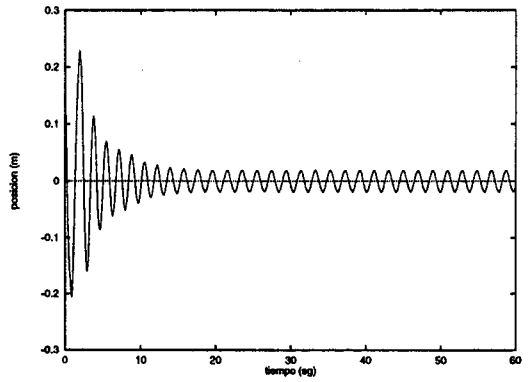



Figura 7: Trayectoria del péndulo.

## 6 Conclusiones y lineas futuras

El presente trabajo aborda la posibilidad de construir controladores borrosos capaces de resolver problemas de cierta complejidad, utilizando técnicas jerárquicas y aplicando controladores de gran sencillez (como son los controladores PD borrosos). La vía para conseguir la definición de la estructura jerárquica es el aprovechamiento de los diferentes niveles de conocimiento disponible sobre el problema.

A través del problema del carro con el péndulo invertido se ha desarrollado un ejemplo concreto que muestra la mecánica a seguir. En dicho ejemplo se ha llegado a obtener un controlador borroso global, en base a dos sencillos controladores PD, y aprovechando al máximo el conocimiento disponible sobre los objetivos del sistema.

Como linea futura de trabajo se plantea la sistematización del enfoque y su aplicación a otros problemas.

## Referencias

[1] A.G. Barto, R.S. Sutton, and C.W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man and Cybernetics*, 13(5):834–846, Sep/Oct 1983.

[2] M.G. Cooper and J.J. Vidal. Genetic design of fuzzy controllers. In *Proceedings 2nd International Conference on Fuzzy Theory and Technology*, October 1993.

[3] J. Van de Vetge. *Feedback control systems*. Prentice Hall International, third edition, 1994.

[4] M.A. Lee and H. Takagi. Integrating design stages of fuzzy systems using genetic algorithms. In *Proceedings 2nd IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'93*, volume 1, pages 612–617, March 1993.

[5] L. Magdalena. Adapting gain of FLCs with genetic algorithms. *International Journal of Approximate Reasoning*, 1997. To appear.

[6] L.X. Wang. *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. PTR Prentice-Hall, Englewood Cliffs, New Jersey, 1994.

[7] R. R. Yager and D. P. Filev. *Essentials of Fuzzy Modeling and Control*. John Wiley & Sons, Inc., 1994.

# NUEVOS METODOS DE RAZONAMIENTO EN SISTEMAS DE CLASIFICACION BASADOS EN REGLAS DIFUSAS *

**Oscar Cordón**
Dpto. de Ciencias de la Computación e I.A.
E.T.S. de Ingeniería Informática
Universidad de Granada
18071 - Granada
e-mail: ocordon@decsai.ugr.es

**Maria José del Jesus**
Dpto. de Informática
Escuela Politécnica Superior
Universidad de Jaén
23071 - Jaén
e-mail: mjj@apolo.ujaen.es

**Francisco Herrera**
Dpto. de Ciencias de la Computación e I.A.
E.T.S. de Ingeniería Informática
Universidad de Granada
18071 - Granada
e-mail: herrera@decsai.ugr.es

## Resumen

El método clásico de razonamiento o inferencia en sistemas de clasificación basados en reglas difusas clasifica un patrón con la clase que forma parte del consecuente de la regla que tiene mayor grado de emparejamiento con el mismo. Esta contribución analiza dos nuevos métodos de razonamiento, alternativos al método clásico, que combinan los diferentes grados de emparejamiento de las reglas del sistema con el patrón de entrada.

**Palabras Clave:** Clasificación, Reglas difusas, Métodos de Razonamiento.

## 1 INTRODUCCIÓN

Los sistemas de clasificación basados en reglas difusas son capaces de manejar el conocimiento impreciso en forma de vaguedad y ambiguedad que a veces envuelve a un problema de clasificación. Para construir un sistema de clasificación basado en reglas difusas debemos de determinar el método de razonamiento utilizado para clasificar un patrón de entrada a partir de la base de reglas difusas.

El método clásico de razonamiento o inferencia en sistemas de clasificación basados en reglas difusas clasifica un patrón con la clase que forma parte del consecuente de la regla que tiene mayor grado de emparejamiento con el mismo [6, 8, 12, 11, 13, 14].

El método de inferencia utilizado no aprovecha toda la potencia del razonamiento aproximado, elemento básico en los sistemas difusos, por cuanto se queda con la clase correspondiente a la regla de mayor grado de compatibilidad y no tiene en cuenta la información proporcionada por el resto de la base de reglas. Por tanto, se pierde la información que proporcionan otras reglas con diferentes etiquetas lingüísticas que tambien se emparejan con el valor del atributo, aunque a un grado menor.

En este trabajo proponemos un nuevo modelo general de razonamiento o inferencia para sistemas de clasificación basados en reglas difusas, y presentamos los resultados obtenidos con dos propuestas de métodos de razonamiento, particularizaciones de este modelo general, y que son una alternativa al método clásico. Los nuevos métodos propuestos no clasifican de acuerdo a la regla con mayor grado de emparejamiento, sino que utilizan toda la información proporcionada por aquellas reglas que tienen un grado de emparejamiento positivo con el patrón de entrada.

La organización de la contribución es la siguiente. En la sección 2 se introducen los sistemas de clasificación difusos y los diferentes tipos de reglas utilizados en éstos. En la sección 3 se presenta el modelo general de inferencia, y los tres métodos particulares estudiados en este trabajo, el método clásico y los dos nuevos métodos alternativos. En la sección 4 se exponen los experimentos realizados y los resultados obtenidos. Por último, se extraen algunas conclusiones y se destacan algunos aspectos de interés.

## 2 SISTEMAS DE CLASIFICACION BASADOS EN REGLAS DIFUSAS

Un sistema de clasificación basado en reglas difusas es un sistema difuso compuesto por una base de reglas difusas (BR) que pueden presentar una de las siguientes formas:

a) Reglas difusas con una clase asociada (modelo clásico)

$$\text{Si } X_1 \text{ es } A_{i1} \text{ y } \ldots \text{ y } X_n \text{ es } A_{in} \text{ entonces } Y \text{ es } C_k \qquad (1)$$

donde $X_1, \ldots, X_n$ son las variables asociadas a los diferentes atributos del sistema de clasificación, $A_{i1}, \ldots, A_{in}$ son las etiquetas lingüísticas utilizadas para discretizar los dominios continuos de las variables que representan las características seleccionadas, e $Y$ es la clase $k$ a la cual pertenece el patrón. Un ejemplo de un sistema que emplea reglas de este tipo puede encontrarse en [6].

b) Reglas difusas con grado de certeza

$$\text{Si } X_1 \text{ es } A_{i1} \text{ y } \ldots \text{ y } X_n \text{ es } A_{in}$$

*Trabajo realizado en el proyecto CICYT-TIC96-0778

entonces $Y$ es $C_k$ con grado $r_i$ (2)

donde se introduce un grado de certeza asociado a la selección de la clase $k$ para los patrones que se emparejan con esta regla. Son empleadas en [8, 12].

c) Reglas difusas con grados de certeza asociados a las diferentes clases

Si $X_1$ es $A_{i1}$ y ... y $X_n$ es $A_{in}$
entonces $Y$ es $(y_i^1, ...., y_i^M)$ (3)

donde $y_i^j$ representa el grado en que los patrones que se emparejan con la regla $i$ pertenecen a la clase $j$ y $M$ indica el numero de clases para el problema, las cuales estarían enumeradas desde 1 hasta $M$. Son utilizadas en los sistemas propuestos en [11, 13].

Este último modelo de regla extiende a los dos anteriores empleando distintos valores para los parámetros del vector $(y_i^1, ...., y_i^M)$. Así, con

$$y_i^k = 1, \quad y_i^j = 0, \quad j <> k, \quad j = 1, \ldots, M \tag{4}$$

tenemos el primer caso, y con

$$y_i^k = r_i, \quad y_i^j = 0, \quad j <> k, \quad j = 1, ..., M \tag{5}$$

el segundo.

Si notamos por $R_i(x)$ el grado de emparejamiento de la regla $R_i$ con el ejemplo $x$, que se obtiene aplicando un operador de conjunción, usualmente una t-norma, sobre el grado de satisfación de las cláusulas $(X_j \quad es \quad A_{ij})$, podemos notar la salida de un sistema de clasificación basado en reglas difusas como el vector

$$y(x) \in [0, 1]^M \tag{6}$$

cuyas componentes son

$$y_j(x) = \psi((y_i^j, R_i(x)), i = 1, ..., L), \quad j = 1, ..., M \tag{7}$$

siendo $L$ el número de reglas. La clase seleccionada, k, es aquella para la que se verifica que

$$y_k(x) = max_j \quad y_j(x). \tag{8}$$

Los métodos de razonamiento presentados en este trabajo están basados en el primero de los modelos de regla. La extensión de los mismos a los demás modelos es inmediata.

A continuación analizaremos el método clásico de razonamiento y los métodos propuestos.

# 3 METODOS DE RAZONAMIENTO

Una vez que el sistema de clasificación ha sido construido, es decir, una vez que se han obtenido las reglas difusas que identifican al problema de clasificación, dicho sistema aplicará el mecanismo de razonamiento sobre la BR y el patrón de test cada vez que le sea presentado un nuevo patrón a clasificar.

Un método de razonamiento difuso es un proceso por el cual, dado un ejemplo, se deduce una conclusión acerca de la clase a la que pertenece el ejemplo haciendo uso de la información contenida en la BR.

Como se ha indicado anteriormente, el modelo de razonamiento clásico para los sistemas de clasificación difusos considera la regla con mayor grado de emparejamiento con el ejemplo y selecciona la clase con mayor grado asociado a dicha regla. Si notamos la BR como $R = \{R_1, \ldots, R_L\}$, para inferir la clase a la cual pertenece el patrón $e_l = (ex_1, \ldots, ex_n)$, calculamos para cada regla $R_k$ el grado de emparejamiento del antecedente con el ejemplo, $R_k(e_l)$, y obtenemos la regla $R_j$ que verifica,

$$R_j(e_l) = max_{i=1,...,L} R_i(e_l). \tag{9}$$

Se dice que el ejemplo pertenece a la clase que se indica por el consecuente de la regla, $C_k$. Lo notaremos como

$$FRM_1(e_l) = clase(R_j) = C_k. \tag{10}$$

En nuestros experimentos hemos utilizado la t-norma del mínimo para determinar $R_j(e_l)$:

$$R_j(e_l) = Min\{A_{j1}(ex_1), ..., A_{jn}(ex_n)\}. \tag{11}$$

En [6] se utiliza la misma t-norma, mientras que en [8] se utiliza el producto, y en [11, 13] se utiliza el operador promedio media aritmética.

Como ya se ha indicado anteriormente, este método de razonamiento desperdicia la información asociada a todas aquellas reglas cuyo grado de emparejamiento con el patrón de entrada es menor que el grado de emparejamiento de la regla seleccionada.

En definitiva, si nosotros consideramos la regla con mayor grado de emparejamiento, estamos considerando un único conjunto difuso para cada valor de los atributos, y no tenemos en cuenta la información de otras reglas con otros subconjuntos difusos que contienen el valor del atributo en su soporte, y que presentan, por tanto, un grado de emparejamiento positivo con el patrón aunque de menor cuantía. Trabajando de esta forma *estaríamos considerando una partición intervalar disjunta en lugar de una partición difusa.*

Al trabajar con un sistema de control difuso sabemos que los mejores resultados se obtienen cuando se utilizan métodos de defuzzificación que integran los conjuntos difusos obtenidos por todas las reglas que se activan [5]. Por tanto, parece lógico utilizar métodos de razonamiento en sistemas de clasificación con reglas difusas que integren toda la información contenida en el modelo. Es decir, es lógico emplear toda la información que nos proporcionan las reglas, en este caso, los grados de emparejamiento del patrón con las mismas, y los consecuentes de éstas.

Por otro lado, existen autores que, empleando otros tipos de sistemas de clasificación, han examinado diferentes esquemas de combinación de los resul-

tados obtenidos por múltiples sistemas de clasificación [2, 3, 10].

La idea propuesta en este trabajo es la siguiente. Si consideramos la BR, $R = \{R_1, \ldots, R_L\}$, podemos dividir ésta en $M$ subconjuntos de acuerdo a la clase indicada por su consecuente,

$$R = R_{C_1} \cup R_{C_2} \cup \ldots \cup R_{C_M}, \quad (12)$$

y podemos definir el grado de asociación del patrón $e_l$ con la clase $C_k$ como:

$$G_{C_k} = \psi(R_i(e_l), i = 1...|R_{C_k}|), \quad k = 1, ..., M \quad (13)$$

con $G_{C_k} \in [0, 1], k = 1, ..., M$, y siendo $\psi()$ una función de agregación sobre los grados de emparejamiento.

En este trabajo vamos a considerar los dos siguientes métodos que están basados en estas ideas:

$$FRM_2(e_l) = C_k \quad (14)$$

tal que

$$G_{C_k} = max_{i=1,...,M}\{G_{C_i}\} \quad (15)$$

$$G_{C_i} = \frac{S_{C_i}}{S_{C_l}} \quad (16)$$

$$S_{C_l} = max_{i=1,...,M}\{S_{C_i}\} \quad (17)$$

$$S_{C_i} = \sum_{R_j \in R_{C_i}} R_j(e_l) \quad (18)$$

obteniendose la suma normalizada de los grados de activación de cada regla como grado final de asociación entre el patrón y la clase.

$$FRM_3(e_l) = C_k \quad (19)$$

tal que

$$G_{C_k} = max_{i=1,...,M}\{G_{C_i}\} \quad (20)$$

$$G_{C_i} = \frac{S_{C_i}}{|R^s_{C_i}|} \quad (21)$$

donde $R^s_{C_i} = \{R_i \in R_{C_i}, R_i(e_l) > 0\}$, obteniendose la media de los grados de activación de cada regla como grado final de asociación entre el patrón y la clase.

Con estos dos métodos de razonamiento combinamos los grados de activación de cada regla para obtener un grado de asociación global del ejemplo a cada una de las clases. En ambos casos, *el patrón $e_l$ se clasifica en la clase que presenta el máximo valor para $FRM_2$ o $FRM_3$*.

El método de razonamiento notado $FRM_2$ ya ha sido propuesto y estudiado en [1] y [9] (la propuesta realizada en el último trabajo es equivalente a este método).

El tercer método, notado $FRM_3$, es una nueva propuesta que considera no solo la información global acerca de la clase a la que pertenece el patrón examinado de acuerdo a las reglas activadas, sino que también tiene en cuenta la calidad o nivel de emparejamiento de dichas reglas.

# 4 EXPERIMENTACIÓN

Para comparar los métodos alternativos con el modelo clásico, hemos utilizado tres conjuntos de datos de aprendizaje bien conocidos, IRIS, WINE y PIMA.

La base de ejemplos IRIS es un conjunto de 150 ejemplos de la flor del iris con tres clases y cuatro atributos. WINE es una base de datos de vinos con trece características importantes, 178 ejemplos y tres clases. PIMA es un conjunto de 768 casos resueltos sobre el diagnóstico de la diabetes donde se tienen en cuenta 8 variables con dos posibles clases (tener la enfermedad o no tenerla).

En los experimentos realizados, hemos considerado una partición difusa constituida por tres etiquetas con funciones de pertenencia triangulares en el caso de WINE y PIMA, y cinco en el caso de IRIS.

Los conjuntos de ejemplos se han dividido de cinco formas distintas, en conjuntos de entrenamiento y test (70% y 30% respectivamente) y los resultados que aparecen en las tablas son medias del porcentaje de clasificación correcta en cada caso.

Las BR utilizadas se han generado utilizando las dos primeras fases del método de aprendizaje propuesto en [4]. La primera de ellas es independiente del método de razonamiento, generando un gran número de reglas que cubren el conjunto de ejemplos, y la segunda selecciona un conjunto de reglas que cooperan para obtener un buen sistema de clasificación considerando para ello un método de razonamiento concreto. Esta selección se hace mediante un algoritmo genético (AG), y el método de razonamiento se integra en el proceso para inferir sobre los ejemplos de entrenamiento y obtener el mejor porcentaje de clasificación posible.

A continuación se describen brevemente las dos primeras etapas del método de aprendizaje. Posteriormente se presentan los resultados y un breve análisis de los mismos.

## 4.1 Método de Aprendizaje

Las dos primeras etapas del método propuesto en [4] son:

1. Un proceso de *generación* de una BR capaz de incluir todo el conocimiento que proporciona el conjunto de ejemplos. Esta BR se genera de forma iterativa a partir de una semántica prefijada para las etiquetas de las variables.

2. Un proceso de *selección* de reglas basado en AGs, que trata de eliminar del conjunto obtenido en el paso anterior, las reglas innecesarias o redundantes. En esta etapa se utilizan los distintos métodos de inferencia que hemos presentado en la sección 3.

### 4.1.1 Generación de la BR

Esta primera etapa consiste en un proceso de generación de la BR basado en una exploración iterativa del espacio de búsqueda del problema.

Para ello, además del conjunto de ejemplos de entrenamiento, se considera una Base de Datos previamente definida, constituida por particiones difusas uniformes con funciones de pertenencia triangulares. Se utiliza el número de términos lingüísticos en cada variable que permita obtener el nivel de granularidad deseado.

Notaremos nuestro conjunto de ejemplos de entrenamiento como $E_p = \{e_1, e_2, \ldots, e_p\}$.

La BR se va construyendo seleccionando la regla más adecuada en cada paso del algoritmo de acuerdo al estado actual del conjunto de entrenamiento. Una vez obtenida la regla, se contabiliza su cubrimiento sobre dicho conjunto. Los ejemplos cubiertos en un grado mayor que un valor $\varepsilon$ especificado se eliminan del conjunto. De esta forma, es el cubrimiento iterativo de los ejemplos lo que guía la búsqueda hacia zonas prometedoras del espacio de búsqueda en cada paso. La eliminación del conjunto de entrenamiento de los ejemplos cubiertos, descarta la posibilidad de que algunos ejemplos dominen todo el proceso de búsqueda.

En este proceso de generación, la selección de la mejor regla se efectúa en función de un conjunto de criterios que permitan asegurar la completitud y consistencia de la BR finalmente generada. Los tres criterios considerados se enumeran a continuación. Su expresión puede ser consultada en [4].

1. *Un valor de frecuencia alto.* $\psi_{E_p}(R_i)$
2. *Alto grado de cubrimiento de los ejemplos positivos.* $G_\omega(R_i)$
3. *Un conjunto de ejemplos negativos pequeño.* $g_n(R_i^-)$

Estos tres criterios son combinados en una función de evaluación de la siguiente forma:

$$F(R_i) = \psi_{E_p}(R_i) \cdot G_\omega(R_i) \cdot g_n(R_i^-) \qquad (22)$$

Las reglas que tienen un mayor valor de esta función son las más adecuadas.

El **método de generación** consta de las siguientes etapas:

1. Inicialización:
   (a) Introducir los valores de los parámetros $k, \omega$, y $\varepsilon$.
   (b) Fijar el grado de cubrimiento de los ejemplos a cero ($CV[l] \leftarrow 0, l = 1, \ldots, p$).
   (c) Inicializar la base de reglas final, $B^g$, al conjunto vacío.
2. Inicializar el conjunto de reglas difusas candidatas, $B^c$, al conjunto vacío.
3. Para cada $e_l \in E_p$, generar la regla difusa $R_c$ que mejor grado de compatibilidad tenga con el ejemplo $e_l$. Si $R_c \notin B^c$ añadirla a $B^c$.
4. Evaluar todas las reglas contenidas en $B^c$ y seleccionar la regla $R_r$ que tenga un valor más alto de la función de evaluación.
5. Introducir $R_r$ en el conjunto de reglas $B^g$.
6. Para cada $e_l \in E_p$ hacer
   (a) $CV[l] \leftarrow CV[l] + R_r(e_l)$.
   (b) Si $CV[l] \geq \varepsilon$ entonces quitar el ejemplo $e_l$ de $E_p$.
7. Si $E_p = 0$ entonces parar, si no, volver al paso 2.

### 4.1.2 Selección de la BR

Como consecuencia de la naturaleza iterativa del proceso de generación de reglas, puede aparecer un fenómeno de sobreaprendizaje, en el sentido de que existan reglas innecesarias o redundantes lo que puede provocar que algunos ejemplos estén cubiertos en un grado mayor que el deseado, y esto haga que la BR obtenida clasifique peor. Para resolver este problema y mejorar el rendimiento, se propone un proceso de simplificación y combinación de las reglas obtenidas en el proceso anterior para derivar una BR que nos permita clasificar correctamente.

El proceso de selección o simplificación fue presentado en [7] y está basado en un AG con codificación binaria en el cual la selección de individuos se realiza con el procedimiento de muestreo estocástico universal, la recombinación mediante el operador de cruce clásico binario multipunto (realizado en dos puntos) y la mutación mediante el operador de mutación uniforme. Además, se aplica el esquema de selección elitista.

El esquema de codificación genera individuos de longitud fija. Si consideramos que en la fase anterior se ha obtenido un conjunto de $r$ reglas, una cadena de r-bits $C = (c_1, \ldots, c_r)$ representa un subconjunto de reglas candidato a formar la BR obtenida finalmente como salida de esta etapa, $B^s$, con el siguiente significado,

$$\text{Si } c_i = 1 \text{ entonces } R_i \in B^s \text{ si no } R_i \notin B^s \qquad (23)$$

La población inicial se genera introduciendo un cromosoma que representa el conjunto de reglas que se obtuvo en la fase anterior, $B^g$, al completo. El resto de los cromosomas se inicializan aleatoriamente.

En este trabajo, hemos utilizado una función de evaluación basada en el error de clasificación sobre el conjunto de ejemplos de entrenamiento. Esta función, $E_1(\cdot)$, para un individuo $C_i$ toma como valor el porcentaje de ejemplos que son mal clasificados con el conjunto de reglas que representa el cromosoma $C_i$.

Para mantener la propiedad de completitud que se consideró en la etapa anterior, requerimos que todo ejemplo contenido en la base de ejemplos de entrenamiento sea cubierto por la BR codificada en un grado mayor o igual a $\tau$, como puede observarse a continuación:

$$C_{R(C_j)}(e_l) = \bigcup_{j=1\ldots T} R_j(e_l) \geq \tau$$
$$\forall e_l \in E_p \text{ y } R_j \in R(C_j) \qquad (24)$$

donde $\tau$ representa el grado mínimo de completitud aceptado en el proceso de simplificación y $R(C_j)$ el conjunto de reglas codificado en el cromosoma $C_j$.

Se define el *mínimo grado de completitud sobre el conjunto de entrenamiento* para el conjunto de reglas codificadas en el cromosoma $C_j$ de la siguiente forma,

$$TSCD(R(C_j), E_p) = \bigcap_{e_l \in E_p} C_{R(C_j)}(e_l) \qquad (25)$$

La función de evaluación final que penaliza la pérdida de la propiedad de completitud queda así:

$$F_1(C_j) = \begin{cases} E_1(C_j) & \text{si } TSCD(R(C_j), E_p) \geq \tau \\ 100 & \text{en otro caso} \end{cases} \qquad (26)$$

### 4.2 Resultados

Table 1: Porcentaje de clasificación para IRIS

| | GENERACION | | | SELECCION | | |
|---|---|---|---|---|---|---|
| I | Ent. | Test | NR | Ent. | Test | NR |
| 1 | 96.2115 | 93.7687 | 61.4 | 97.6630 | 92.8550 | 45 |
| 2 | 98.0296 | 95.7343 | 61.4 | 99.6338 | 95.6339 | 38.2 |
| 3 | 97.8088 | 94.9455 | 61.4 | 99.6441 | 94.5427 | 40.6 |

Table 2: Porcentaje de clasificación para WINE

| | GENERACION | | | SELECCION | | |
|---|---|---|---|---|---|---|
| I | Ent. | Test | NR | Ent. | Test | NR |
| 1 | 98.3947 | 89.8250 | 101.8 | 99.8347 | 89.9723 | 56.4 |
| 2 | 98.3905 | 92.7008 | 101.8 | 100 | 92.0212 | 60.6 |
| 3 | 93.7040 | 88.7494 | 101.8 | 99.8450 | 93.0821 | 57 |

Table 3: Porcentaje de clasificación para PIMA

| | GENERACION | | | SELECCION | | |
|---|---|---|---|---|---|---|
| I | Ent. | Test | NR | Ent. | Test | NR |
| 1 | 75.2415 | 68.0851 | 142 | 79.5095 | 70.4288 | 78.5 |
| 2 | 76.0266 | 74.0019 | 142 | 82.0390 | 71.9648 | 101.5 |
| 3 | 75.8507 | 74.0019 | 142 | 82.5621 | 75.3040 | 103 |

Los resultados obtenidos en las etapas de generación y selección se muestran en las tablas 1, 2 y 3, donde la primera columna, encabezada por la letra $I$, hace referencia al metodo de razonamiento utilizado: 1 el clásico, y 2 y 3 las dos propuestas alternativas. La columna encabezada por $NR$ indica el número medio de reglas obtenido en cada caso. El resto de columnas indican el porcentaje de clasificación sobre los ejemplos de entrenamiento y test para cada una de las etapas.

Los dos métodos de razonamientos basados en la combinación de los grados de emparejamiento de varias reglas ($FRM_2, FRM_3$) muestran un mejor comportamiento que el método clásico en la clasificación de los nuevos ejemplos (ejemplos de test) en las tres bases de ejemplos y en las dos fases del proceso de aprendizaje.

Analizando los resultados presentados en las tres tablas, observamos que el método de inferencia 2 presenta el mejor comportamiento despues de la fase de generación, pero sin embargo una vez que se aplica el algoritmo de selección y se integra con este el método de razonamiento, los mejores resultados son obtenimos por el tercer método, $FRM_3$.

## 5 CONCLUSIONES

Los resultados obtenidos son alentadores y cabe pensar que otras alternativas a los métodos propuestos son posibles y pueden presentar resultados interesantes.

Una primera propuesta alternativa con la que pensamos experimentar es usar el operador de agregación OWA, propuesto por Yager en [15], empleando un cuantificador para calcular los pesos asociados al operador que represente el concepto de mayoría difusa. La idea subyacente es que agregamos una mayoría difusa de los mejores grados de emparejamiento asociados a las reglas.

Por otro lado, cabe destacar la inmediata generalización de las propuestas realizadas para su uso en el tercer modelo de sistema de clasificación difuso. Recordemos que la expresión de las reglas era de la forma:

$$\text{Si } X_1 \text{ es } A_{i1} \text{ y } \ldots \text{ y } X_n \text{ es } A_{in} \quad \text{entonces } Y \text{ es } (y_i^1, ...., y_i^M) \qquad (27)$$

donde $y_i^j$ representa el grado en que los patrones que se emparejan con la regla $i$ pertenecen a la clase $j$. Se puede obtener el grado de asociación de la clase k con el patrón $x$ como:

$$y_k(x) = \psi(g(y_i^k, R_i(x)), i = 1, ..., L) \qquad (28)$$

donde $g(y_i^j, R_i(x))$ integra el grado de asociación entre el patron y la clase de acuerdo a la citada regla (la función g se ha utilizado como el mínimo, el producto y la media en diferentes trabajos) y para cada clase la funcion $\psi()$ agrega los anteriores grados de asociación de acuerdo a los distintos modelos anteriormente estudiados (máximo, suma normalizada, media, ...).

Como trabajo futuro nos planteamos analizar el comportamiento de los distintos métodos de razonamiento en los diferentes modelos.

## References


[1] A. Bardossy, L. Duckstein. Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological and Engineering Systems. CRC Press. (1995).

[2] R. Battiti, A.M. Colla. Democracy in Neural Nets: Voting Schemes for Classification. Neural Networks 7 (1994) 691-703.

[3] J.A. Benediktsson, P.H. Swain. Consensus Theoretic Classification Methods. IEEE Trans. on Systems, Man and Cybernetics 22 (1992) 688-704.

[4] O. Cordón, M.J. del Jesus, F. Herrera. Sistema de Clasificación con Reglas Difusas Utilizando Algoritmos Genéticos. Actas del VI Congreso Español Sobre Tecnologías y Lógica Fuzzy (ESTYLF96), Oviedo, 1996, 95-100.

[5] O. Cordón, F. Herrera, A. Peregrín. Aplicability of the Fuzzy Operators in the Design of Fuzzy

Logic Controllers. Fuzzy Sets and Systems 86 (1997) 15-41.

[6] A. González, R. Pérez. Completeness and Consistency Conditions for Learning Fuzzy Rules. Fuzzy Sets and Systems, 1997. Por aparecer.

[7] F. Herrera, M. Lozano, J.L. Verdegay. A Learning Process for Fuzzy Control Rules Using Genetic Algorithms. Fuzzy Sets and Systems, 1997. Por aparecer.

[8] H. Ishibuchi, K. Nozaki, H. Tanaka. Distributed representation of fuzzy rules and its application to pattern classification. Fuzzy Sets and Systems 52 (1992) 21-32.

[9] H. Ishibuchi, T. Morisawa, T. Nakashima. Voting Schemes for Fuzzy-Rule-Based Classification Systems. Proc. Fith IEEE Int. Conference on Fuzzy Systems, 1996, pp. 614-620.

[10] Y. Lu. Knowledge Integration in a Multiple Classifer System. Applied Intelligence 6 (1996) 75-86.

[11] D.P. Mandal, C.A. Murthy, S.K. Pal. Formulation of a Multivalued Recognition System. IEEE Trans. on Systems Man, and Cibernetics 22 (1992) 607-620.

[12] K. Nozaki, H. Ishibuchi, H. Tanaka. Adaptive Fuzzy Rule Based Classification Systems. IEEE Trans. on Fuzzy Systems 4 (1996) 238-250.

[13] S. Pal, D.P. Mandal. Linguistic Recognition System Based on Approximate Reasoning. Information Sciences 61 (1992) 135-161.

[14] K.S. Ray, J. Ghoshal. Approximate reasoning approach to pattern recognition. Fuzzy Sets and Systems 77 (1996) 125-150.

[15] R.R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decisionmaking. IEEE Trans. on Systems Man, and Cibernetics 18 (1988) 183-190.

# Hibridación de Técnicas para la Generación y Ajuste de Reglas en el Modelado Difuso[1]

**M. Delgado**
Dept. Ciencias de la Computación e Inteligencia Artificial. Universidad de Granada
mdelgado@ugr.es

**A. F. Gómez-Skarmeta**
**J. Gómez Marín-Blázquez**
**H. Martínez Barberá**
Dept. Informática y Sistemas.
Universidad de Murcia
Skarmeta@dif.um.es

## Resumen

Describiremos diferentes métodos de aproximación al problema de la generación y el ajuste de reglas difusas. Para la generación de las reglas utilizaremos diferentes algoritmos de agrupamiento. Esta aproximación presupone que los datos carecen de estructura. Para el ajuste de reglas utilizaremos dos técnicas diferentes. Una de ellas basadas en gradientes descendientes. La otra basada en el intento de ajustar las salidas de las reglas para reducir el error.

**Palabras Clave:** Agrupamiento; Gradiente Descendente; Aprendizaje No Supervisado; Modelado Difuso.

## 1. Introducción

En los últimos tiempos el uso de diferentes técnicas que podríamos englobar dentro del marco del Soft-Computing ha sido abundante en la literatura relacionada con el modelado difuso. Así encontramos propuestas de uso de redes neuronales difusas [1] [12], teoría de subconjuntos difusos en combinación con técnicas de gradiente descendente [14], o con técnicas de agrupamiento [6] [15] [14] [2]. Sin embargo sólo desde hace relativamente poco tiempo han ido apareciendo combinaciones de estas técnicas a la hora de abordar los diferentes problemas asociados con el modelado de sistemas [13] [7] [11].

Dentro del contexto del modelado difuso, el uso de técnicas basadas en agrupamiento ha tenido notable éxito. El objetivo de dicho agrupamiento es la posibilidad de detectar líneas de comportamiento en los datos obtenidos de un sistema en observación y del cual no se dispone de ninguna información adicional, con objeto de realizar un proceso de modelado difuso mediante un conjunto de reglas difusas. En [4] [3] [5] se plantean diversos métodos bien aproximativos bien descriptivos. Al final, tras los resultados descritos en estos trabajos, lo que obtenemos es un conjunto de reglas que son sólo una primera aproximación al problema del modelado difuso del sistema en estudio. El objetivo siguiente es abordar el problema de como optimizar las reglas generadas mediante los métodos que se han propuesto. Para ello proponemos el uso de técnicas de ajuste basadas en la modificación de alguno de los parámetros de las reglas de difusas, que complementen los resultados obtenidos mediante las técnicas de agrupamiento y den así sentido a la creación de sistemas híbridos aplicables al modelado difuso de sistemas.

En este artículo presentamos diferentes aproximaciones al problema de generar reglas difusas, creando dicha división mediante técnicas de agrupamiento. Con estas técnicas podemos crear unas reglas difusas sin suponer que exista ningún tipo de estructura en los datos de entrada. Dichas reglas serán el punto de partida para un ajuste posterior aplicando dos algoritmos diferentes de ajuste de reglas difusas. La idea consiste en aplicar de forma sucesiva diferentes algoritmos de refinamiento. Este esquema de trabajo nos permite encontrar reglas con un alto grado de aproximación partiendo de datos heterogéneos.

En la siguiente sección presentamos la notación y la forma en que se usan las reglas para describir el comportamiento del sistema, desarrollando un método para la creación de reglas difusas a partir de un agrupamiento. Tras ello, describiremos dos métodos diferentes de ajuste de dichas reglas para mejorar el error. Por último mostraremos algunos ejemplos y daremos algunas conclusiones.

[1] Co-financiado por Proyecto PCOM-08/96 TEC de la CARM y por el Instituto de Fomento de la Región de Murcia (INFO)

## 2. Modelado Difuso

### 2.1 Agrupamiento para la detección de Comportamientos

Consideraremos, sin perder generalidad, un sistema MISO (múltiple entrada, salida única). Deseamos encontrar un sistema que nos aproxime la función $\varphi : X^p \rightarrow Y$ que modeliza el sistema. Disponemos de un conjunto de muestras (ejemplos) del comportamiento del sistema en el espacio $(X^p \times Y)$, $\Omega = ((x_{t1}, x_{t2}, \ldots, x_{tp}), y_t) \quad t = 1,2,\ldots,n$, donde $X_1, X_2, \ldots, X_p$ son los dominios de discurso de las entradas e $Y$ es el dominio de discurso de la salida.

También consideramos que conocemos, de algún modo, la existencia de $k$ agrupamientos en los datos. Lo que es importante destacar es que los agrupamientos sobre los que trabajaremos pueden ser obtenidos directamente al aplicarse el análisis de agrupamiento difuso sobre los datos, o bien tras haberse realizado un preproceso utilizando un agrupamiento jerárquico, tal y como ya se ha estudiado en otros trabajos [5]. No necesitamos conocer ninguna otra información de la estructura de los datos (por ejemplo, la linealidad o no de estos). Utilizaremos un algoritmo de agrupamiento para hallar $k$ centroides. Con objeto de realizar el agrupamiento de los datos y poder detectar de esta forma los centroides asociados a los diferentes agrupamientos detectables, vamos a utilizar el algoritmo clásico de los mapas auto-organizativos de Kohonen, de forma que no sólo se modifica en cada iteración el centroide "ganador" o más cercano al dato de aprendizaje, sino un conjunto de centroides en su vecindad, vecindad que va disminuyendo a medida que van aumentando el número de iteraciones. Un aspecto importante a destacar es que el proceso de agrupamiento se realiza sobre el espacio producto de las entradas y las salidas. El objeto de este proceso es, tal y como se ha mostrado en diversos trabajos [8] [9] [4], que se detecten mejor los agrupamientos existentes en los datos ya que se tiene en cuenta no sólo la relación de interactividad entre las variables del espacio producto de las entradas, sino que también las consecuencias de su interrelación con el espacio de las salidas. Es decir, el agrupamiento se realizará sobre el espacio $(X^p \times Y)$, es decir, tendremos también en cuenta los valores de salida de los ejemplos. Con ello obtendremos $k$ centroides $C_h$ con la forma $(w_1, w_2, \ldots, w_p, w_{p+1}), h = 1,2,\ldots,k$ donde $w_1, w_2, \ldots, w_p$ pertenecen, respectivamente, a los dominios $X_1, X_2, \ldots, X_p$ y $w_{p+1}$ pertenece al dominio de $Y$

Suponiendo que ese agrupamiento defina las tendencias del comportamiento presentes en los ejemplos, lo que queremos es conseguir una caracterización de dicho comportamiento utilizando los centroides para crear k reglas difusas que responderán a la siguiente forma (if-then):

$$R_h : \textbf{Si } x \textbf{ es } A_h \textbf{ entonces } y \textbf{ es } B_h \quad h = 1,\ldots,k$$

donde $A_h$ y $B_h$ son, respectivamente, los conjuntos difusos en $X^p$ e $Y$ obtenidos como representación del h-ésimo centroide $C_h$ de $(X^p \times Y)$. Una vez creadas todas esas reglas (en el apartado siguiente veremos cómo) se pueden utilizar un mecanismo de inferencia para cualquier nueva entrada, utilizando para ello razonamiento aproximado.

### 2.2 Generación de Reglas Difusas

Por sencillez hemos escogido la forma de regla difusa simplificada [10] en la cual el conjunto difuso del consecuente de cada una de las reglas se sustituye por un valor simple o consecuente puntual, siendo, en nuestro caso, un número real. Así pues tendremos reglas del tipo:

$$R_h : \textbf{Si } x \textbf{ es } A_h \textbf{ entonces } y \textbf{ es } v_h \quad h = 1,\ldots,k$$

Los conjuntos difusos de $A_h$ se generan con centro (o punto medio, dependiendo del tipo de número difuso que empleemos) igual al centroide. Para calcular su extensión asignamos a cada ejemplo un centroide (en particular, el centroide más cercano a cada ejemplo) y vemos para cada centroide, entre los ejemplos que están asignados a él, cual es el que se halla a mayor distancia. La extensión del conjunto difuso generado por dicho centroide será siempre superior a dicha distancia. La razón de esto es para asegurar que todos los ejemplos pertenezcan, al menos, a un conjunto difuso. Debe ser mayor que dicha distancia para que ese elemento no tenga un valor de pertenencia a dicho conjunto igual a cero (ver más abajo las funciones de pertenencia), que sería equivalente a la no-pertenencia. Esto nos asegura también que se disparará, al menos, alguna regla, para cada uno de los ejemplos.

El valor de pertenencia a la que nos referimos es una interpretación de la expresión del antecedente de las reglas ($x$ es $A_h$). En particular, las reglas pueden expresarse de la siguiente forma:

de aprendizaje $g_S(t)$ con $1 \geq g_S(t) \geq 0$ y por la fuerza de disparo de la propia regla $\mu_{A_k}(t)$, siguiendo el siguiente criterio $v_k(t+1) = v_k(t) + g_s(t) \cdot \mu_{A_k}(t) \cdot \left| y_j^* - y_j \right|$. Con esto conseguimos que la salida global del sistema para dicho ejemplo $y_j$ se acerque a $y_j^*$.

El punto fuerte de este sistema es su sencillez. Por el contrario no nos asegura que el ajuste respecto a un ejemplo no nos perjudique otros ejemplos.

**Método 2**

Este segundo método intenta mejorar el error utilizando gradiente descendente. A partir del valor del error se obtienen fórmulas de la dependencia de $y_j$ respecto a las salidas de las reglas (las $v_h$) y respecto a los propios números difusos que las disparan. Así pues, se pueden modificar tanto las $v_h$ como los conjuntos difusos de las reglas. En particular, un método descendiente busca el vector $Z$ que minimiza una función objetivo $E(Z)$ donde $Z$ es un vector p-dimensional $Z = (z_1, z_2, \ldots, z_p)$ de parámetros ajustables. En este método, el vector que disminuye la función objetivo $E(Z)$ se expresa mediante $\left( -\partial E/\partial z_1, -\partial E/\partial z_2, \ldots, -\partial E/\partial z_q \right)$, y la regla de aprendizaje se expresa de la siguiente forma

$$z_i(t+1) = z_i(t) - \mathrm{K} \frac{\partial E(Z)}{\partial z_i}, \qquad (i = 1,2,\ldots,q)$$

donde t es la iteración del proceso de ajuste y K es una constante. En nuestro caso, la función objetivo a minimizar es la del error [II] cometido por el sistema de reglas difusas:

Si utilizamos números difusos triangulares, tal como se ha descrito anteriormente, vemos que el sistema depende de tres grupos de variables diferentes, que son: los centros de los números difusos ($c_{ij}$), la base de los números difusos $(b_{ij})$ y los consecuentes de las reglas $(v_i)$ con $(i = 1,2,\ldots,n)$, y $(j = 1,2,\ldots,p)$. Así pues el vector $Z$ tiene la forma $(z_1, z_2, \ldots, z_q) = (c_{11}, \ldots, c_{np}, b_{11}, \ldots, b_{np}, v_1, \ldots, v_n)$.

Desarrollando $z_i(t+1)$ tenemos:

$$c_{ij}(t+1) = c_{ij}(t) - \mathrm{K}_c \cdot \frac{\partial E}{\partial a_{ij}}$$

$$b_{ij}(t+1) = b_{ij}(t) - \mathrm{K}_b \cdot \frac{\partial E}{\partial b_{ij}}$$

$$v_i(t+1) = v_i(t) - \mathrm{K}_v \cdot \frac{\partial E}{\partial v_i}$$

Ya solo queda realizar un proceso iterativo de aplicación de estas tres expresiones sobre las variables descritas. El proceso acabará cuando la reducción del error entre dos iteraciones sea menor que una cota determinada. Las constantes que aparecen en las expresiones nos permiten ajustar la agresividad del método.

## 3. Resultados de la simulación

Hemos verificado los dos métodos antes descritos con la función no lineal Sinc:

$$z = Sinc(x,y) = \frac{Sin(x)}{x} \cdot \frac{Sin(y)}{y}$$

Como grupo de ejemplos tomamos 225 muestras aleatorias del área comprendida dentro del intervalo $[-10,10] \times [-10,10]$. Dentro de dicho intervalo se obtienen resultados en z comprendidos en el intervalo $[-0.210401, 0.991247]$, como se puede apreciar en Fig. 2. Los pares $(x,y)$ se utilizan como ejemplos de entrada y las z como ejemplos de salida, con lo que tenemos $e_j = (x_j, y_j, z_j)$ con $j = 1,2,..,225$.

Se han realizado varios agrupamientos con diferente número de centroides, en particular con 10,15,20 y 25 centroides. Partiendo de dichos agrupamientos se han creado las reglas siguiendo los pasos descritos en 2.2. Se les ha aplicado a dichas reglas los dos métodos descritos en 2.3. Para el método 1 se ha utilizado un $g_S(t)$ que varia de 0.05 a 0 en 150 Epochs[2] (que no se alcanzaron nunca). Para el método 2 se utilizaron unas constantes $K_c = K_b = K_v = 0.2$. A continuación mostramos algunas gráficas sobre los resultados. Estos resultados han sido obtenidos sobre un conjunto de validación de 650 muestras diferentes de los ejemplos de entrenamiento sobre las reglas generadas cada Epoch.

[2] Epoch: Iteración del algoritmo de ajuste. En cada iteración se realiza un ajuste por cada ejemplo.

$$R_h: \textbf{Si } x_1 \textbf{ es } A_{1h} \textbf{ y } x_2 \textbf{ es } A_{2h} \textbf{ y ... y } x_p \textbf{ es } A_{ph} \textbf{ entonces } y \textbf{ es } v_h \quad h=1,...,k$$

Cada una de las expresiones ($x_j$ es $A_{jh}$) se interpreta como el *grado de pertenencia* $\mu_{A_{jh}}(x_j)$ del valor de entrada $x_j$ en el conjunto difuso $A_{jh}$. Dichos grados de pertenencia vienen definidos por la *función de pertenencia* que depende del tipo de número difuso que se utilice. Nosotros utilizaremos números difusos triangulares. Dichos números quedan definidos por dos valores, centro (c) y base (b), cuya función de pertenencia es:

$$\mu_{A_{jh}}(x_j)=\begin{cases}1-\dfrac{2\cdot\|x_j-c_{jh}\|}{b_{jh}} & c_{jh}-b_{jh}/2 \le x_j \le c_{jh}+b_{jh}/2 \\ 0 & \text{En otro caso}\end{cases}$$

Para terminar de construir la regla necesitaremos un valor inicial de salida. Dicho consecuente puntual asociado a cada una de las reglas lo calcularemos como la media de las salidas de los ejemplos que la disparan ponderada por el valor de pertenencia de dicho ejemplo. Si tenemos $n$ ejemplos de dimensión $p$ de la forma $e=[(x_1,y_1),(x_2,y_2),...,(x_n,y_n)]$ con $x_i \in X^p$ e $y_i \in Y$, el valor inicial de salida de la regla $v_h$ se expresa de la siguiente forma:

$$v_h=\frac{\sum_{j=1}^{n}\mu_{A_h}(x_j)\cdot y_j}{\sum_{j=1}^{n}\mu_{A_h}(x_j)}$$

donde $\mu_{A_h}(x_j)$ es la *fuerza de disparo* de la regla $R_h$ para el ejemplo $x_j$. Existen varias formas de expresar dicha fuerza de disparo. Se podría definir dicha fuerza de disparo como una medida de la coincidencia global de cada una de las componentes del ejemplo $x_{ij}$, $i=1,2,...,p$ con cada uno de los números difusos $A_{ih}$ que componen la regla (como hemos visto antes, dicha coincidencia se mide por el valor de pertenencia). En particular utilizaremos la multiplicación [I] de las fuerzas de disparo.

$$\mu_{A_h}=\prod_{i=1}^{p}\mu_{A_{ih}}(x_{ij}) \qquad [\mathbf{I}]$$

Una vez definidas todas las reglas del sistema, podemos usarlas para inferir resultados sobre nuevos ejemplos. La forma de inferir dicho resultado utiliza el método de razonamiento simplificado de Mizumoto [10] y se define como la media de las salidas de las reglas que son disparadas por el ejemplo ponderadas por la fuerza de disparo de dicha regla para dicho ejemplo. Es decir:

$$y=\frac{\sum_{h=1}^{k}\mu_{A_h}v_h}{\sum_{h=1}^{k}\mu_{A_h}}$$

### 2.3 Ajuste de Reglas Difusas

El grupo de reglas que hemos generado ya constituye una aproximación bastante buena al sistema que queremos modelar. Podemos calcular una medida del error que cometemos al usarlas. Como solo conocemos las salidas de los ejemplos, podemos estimar con ellos el error que cometemos al aproximar a la función. Utilizaremos el Error Cuadrático Medio (Mean Square Error):

$$E=\frac{\sum_{j=1}^{n}(y_j^*-y_j)^2}{n} \qquad [\mathbf{II}]$$

Siendo $y_j^*$ el valor real de salida del ejemplo *j-ésimo* e $y_j$ el valor inferido por el sistema de reglas difusas para el ejemplo *j-ésimo*.

Partiendo de estas medidas podemos intentar reducir el valor de ellas mediante ajustes sobre las reglas. En particular deberíamos intentar que los valores inferidos se acerquen más a los reales. Para ello hemos creado dos aproximaciones basadas en ajuste de salidas y en gradientes descendientes.

#### Método 1

El Método 1 es un algoritmo que intenta mejorar la salida global mediante una serie de mejoras individuales. Este método modifica las $v_h$ de las reglas para conseguir un $y_j$ más cercano a $y_j^*$.

Para cumplir este objetivo tomaremos un ejemplo $e_j=(x_j,y_j^*)$. Se comprueban cuantas reglas se disparan con dicho ejemplo y tendremos así un subconjunto de ellas $P_j=\{R_h/\mu_{A_h}(x_j)>0\}$. De entre todas las reglas que se disparan, se escoge la que tiene una fuerza de disparo mayor $R_k \in P_j / v_h=\text{Max}(v_i \in P_j)$. Se ajusta la salida de dicha regla para acercar su salida $v_k$ al valor de salida del ejemplo $y_j^*$ ponderado por un factor

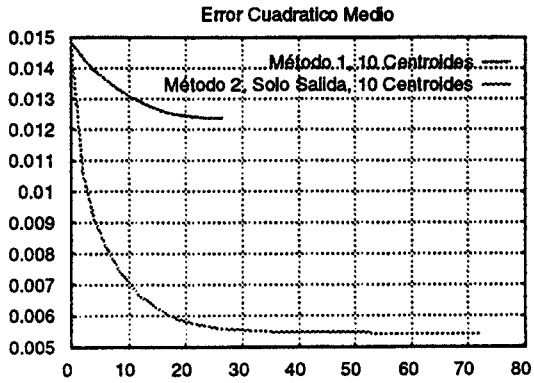



Fig. 1. Comparación del error obtenido por el método 1 y el método 2 (modificando sólo la salida) con 10 centroides.

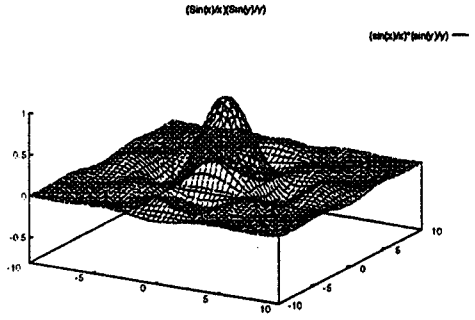


Fig. 2. Representación tridimensional de la función Sinc.

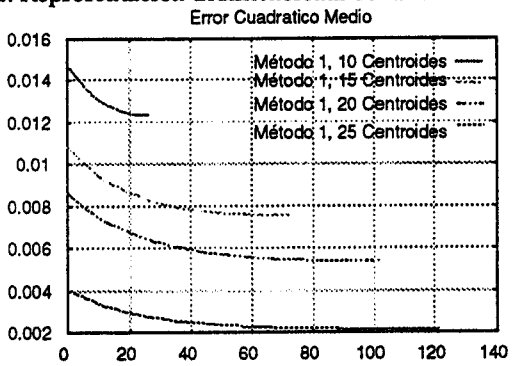



Fig. 3. Comparación del error obtenido por el método 1 en función del número de centroides.

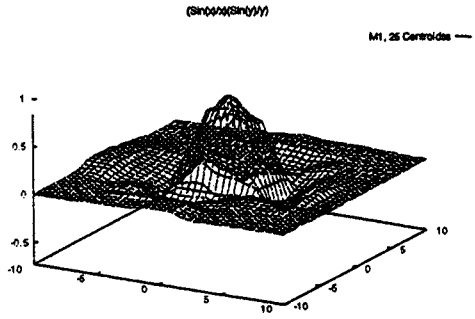


Fig. 4. Aproximación de la función Sinc usando el Método 1.

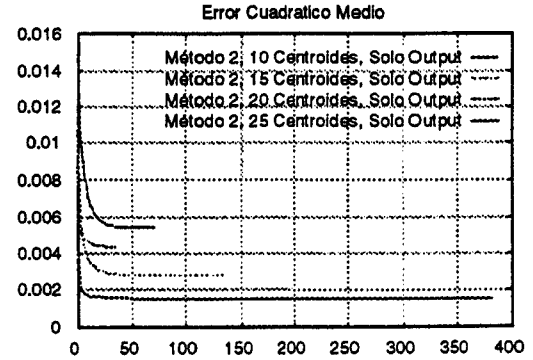



Fig. 5.Comparación del error obtenido por el método 2 en función del número de centroides.

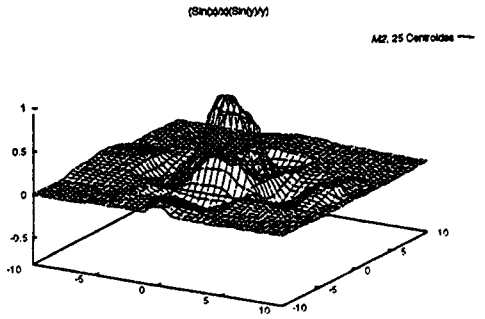


Fig. 6. Aproximación de la función Sinc usando el Método 2.

En la Fig. 1 podemos observar las diferencias entre ambos métodos. El Método 2 es mucho más efectivo que el 1 pero su coste computacional es muy elevado. De hecho, con una función de aprendizaje $g_s(t)$ que empiece en 0.5, el método 1 es capaz de alcanzar valores cercanos al valor final (0.0122) en tan sólo 3 epochs. El método 2 alcanza valores muy aceptables a partir del epoch 30.

En la Fig. 3 vemos como al añadir más centroides mejora mucho el error pero a costa de necesitar más epochs para alcanzar los valores finales y de añadir complejidad al cálculo de estos.

En la Fig. 5 vemos los resultados utilizando el método 2 con $K_c = K_b = 0$ (con lo que sólo modificaremos las salidas de las reglas), para poder comparar mejor con el método 1. Se consiguen mejores resultados en pocos epochs volviendo que dichos epochs más costosos.

En las Figuras 2, 4 y 6 se aprecia la forma que tiene realmente la función y la forma de la aproximación por ambos métodos.

## 4. Conclusiones y Vías Futuras

En este trabajo hemos proponemos un enfoque al modelado difuso consistente en una primera aproximación al modelo mediante el uso de técnicas agrupamiento que

nos permite la consideración de un conjunto mucho más reducido de reglas, inducidas a partir de los agrupamientos obtenidos sobre el espacio producto de las entradas y las salidas. Debido a que lo que se busca en esta etapa es una primera aproximación, los métodos propuestos son simples pero con gran capacidad aproximativa, al permitirnos trabajar con conjuntos difusos directamente en el espacio producto de las variables de entrada.

A continuación, una vez obtenido este primer modelo, presentamos dos métodos diferentes para realizar una ajuste del conjunto de reglas difusas obtenidas a partir de la información que suministra el agrupamiento inicial. Estos dos métodos se caracterizan en un caso por su sencillez y en el otro por la gran potencia de aproximación que alcanza, a costa de un mayor costo computacional.

Un aspecto a considerar una vez caracterizados los conjuntos difusos de las reglas, es como poder relacionar este enfoque con otros similares en la literatura pero que obtienen modelos difusos correspondientes a enfoques de tipo descriptivos, obtenidos a través de un conjunto de etiquetas predefinidas. En este sentido es necesario estudiar como relacionar los mecanismos de ajuste de las reglas con las matrices de pesos que indiquen la certeza que se tiene en cada posible combinacion de etiquetas.

## 5. Referencias


[1] Ahn, T.;Oh, S.; Woo, K., "Automatic Generation of Fuzzy Rules using the Fuzzy-Neural Networks", Fifth IFSA World Congress, pp. 1181-1185, 1993.

[2] Berenji, H.R.; Khedkar, P.S.; "Clustering in Product Space for Fuzzy Inference", Proceedings of the IEEE Int. Conf. on Fuzzy Systems, pp. 1402-1407, 1993

[3] Delgado, M., Gómez Skarmeta, A.F., Martín F., "Un Enfoque Aproximativo para la Generación de Reglas mediante Análisis Cluster", V Congreso Español sobre Tecnología y Lógica Fuzzy, Sep. 1995, pp. 43-48, Murcia, España

[4] Delgado, M., Gómez Skarmeta, A.F., Martín F.,"Generating Fuzzy Rules Using Clustering Based Approach", Third European Congress on Fuzzy and Intelligent Technologies and Soft Computing, August. 1995, pp. 810-814, Aachen, Germany

[5] Delgado, M., Gómez Skarmeta, A.F., Vila, A., "On the Use of a Hierarchical Clustering in Fuzzy Modeling", Int. Journal of Approximate Reasoning, vol 14, n°4, 237-259, 1996

[6] Gómez Skarmeta, A.F., Delgado, M., Martín, F., "Using Fuzzy Clustering in a Descriptive Fuzzy Modeling Approach", Sixth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU'96, July 1996, pp. 563-569, Granada, Spain

[7] Ishibuchi, H.; Nozaki, K.; Yamamoto, N.; Tanaka, H., "Construction of fuzzy classification systems with rectangular fuzzy rules using genetic algorithms", Fuzzy Sets and Systems, vol. 65, pp. 237-253, 1994

[8] Kroll, A., "Identification of functional fuzzy models using multidimensional reference fuzzy sets", Fuzzy Sets and Systems, vol. 80, pp. 149-158, 1996

[9] Langari. R., Wang, L., "Fuzzy models, modular networks, and hybrid learning", Fuzzy Sets and Systems, vol 79, pp. 141-150, 1996

[10] Mizumoto, M. "Method of Fuzzy Inference suitable for Fuzzy Control", *J.Soc. Instrument and Control Engrs*, 58, pp. 959-963, 1989

[11] Sun, C.T., "Rule-Base Structure Identification in an Adaptive-Network-Based Fuzzy Inference System", { IEEE Transactions on Fuzzy Systems, vol. 2, no. 1, February, pp. 64-73, 1994

[12] Vourimaa, P., "Fuzzy self-organizing map", Fuzzy Sets and Systems, vol. 66, pp. 223-231, 1994

[13] Wang, L., Langari. R., Complex Systems Modeling via Fuzzy Logic, IEEE Trans. on Systems, Man and Cybernetics, vol 26, no. 1, pp. 100-105, 1996

[14] Yager, R. R.; Filev, D. P., "Generation of fuzzy rules by mountain Clustering", Tech. Report MII-1318, Iona College, 1993.

[15] Yoshinari, Y.; Pedrycz W.; Hirota K., "Construction of Fuzzy Models through Clustering Techniques", Fuzzy Sets and Systems, Vol.54, pp.157-165, 1993

# Modelado a partir de ejemplos imprecisos mediante algoritmos GA-P

**Luciano Sánchez**
Dpto. Informática (U. Oviedo)
Campus de Viesques. Gijon.
luciano@lsi.uniovi.es

**Inés Couso**
Dpto. Estadística y D.M. (U. Oviedo)
c/ Calvo Sotelo s/n. Oviedo.
couso@pinon.ccu.uniovi.es

## Resumen

Los algoritmos GA-P son una variante de la programación genética que ha sido optimizada para resolver problemas de regresión simbólica. En este trabajo se muestra cómo un algoritmo GA-P que emplee aritmética de intervalos en su parte expresional puede resolver problemas de aprendizaje de modelos numéricos a partir de ejemplos imprecisos, con resultados comparables tanto a los métodos clásicos como a los algoritmos más conocidos de aprendizaje de modelos borrosos, y a la vez proporciona una expresión analítica sencilla del proceso que se esté modelando. El algoritmo propuesto ha sido empleado en un problema real de ingeniería eléctrica, la determinación de una ley empírica que relacione la longitud total de línea eléctrica en una población rural española con su número de habitantes y su extensión.

## 1 Introducción

Al modelar procesos físicos, es frecuente que necesitemos conocer el valor de una variable $Y$, basándonos en la información proporcionada por una tupla de variables $X = (X_1, \ldots, X_d)$. Por ejemplo, $Y$ podría ser la salida de un sistema y $X$ su entrada, o bien $Y$ podría ser una variable de estado no observable directamente y $g$ su observador a partir de las variables $X$. En cualquier caso, buscaremos una función $g$ tal que la diferencia $Y - g(X)$ sea pequeña para cualquier valor de $X$. Desde un punto de vista estocástico, podemos suponer que existe un experimento aleatorio gobernado por una distribución de probabilidad $P$, con resultados en un conjunto $\Omega$, tal que $Y : \Omega \longrightarrow \mathbb{R}$ es una variable aleatoria y $X : \Omega \longrightarrow \mathbb{R}^d$ es un vector aleatorio. Dado entonces un valor $X = x$, podemos medir lo adecuadamente que $g$ predice a $Y$ mediante el promedio de los cuadrados de los errores que cometemos en cada caso. Es bien conocido que la función $g$ que minimiza este valor es $g(x) = E[Y|X = x]$ y que, si definimos un estadístico $V_g = \frac{1}{N}\sum_{i=1}^{N}[Y_i - g(X_i)]^2$, bajo ciertas condiciones la función $\widehat{g} = \arg\min V_g$ será una buena estimación de $E[Y|X]$ (la estimación por mínimos cuadrados) y podemos, sin demasiado error, emplear $\widehat{g}$ en vez de $g$. Obsérvese que, al hacer este cambio, transformamos el problema de regresión en un problema de optimización, la búsqueda del mínimo de $V_g$.

En el caso más habitual, el mínimo de $V_g$ se calcula mediante un algoritmo numérico, suponiendo que $g$ es conocida a excepción de los valores de un conjunto de parámetros. Pero la deducción de la expresión de $g$ no suele ser sencilla. Por esta razón, se han desarrollado algunos procedimientos automáticos de determinación de la expresión analítica de la función que mejor describe una muestra dada. Este último problema se denomina *regresión simbólica* en la literatura de aprendizaje de máquinas, y se estudia dentro de las técnicas de inducción de programas a partir de ejemplos de funcionamiento, como la *programación genética* [6].

## 2 Predicciones multivaluadas

En ciertos problemas prácticos es interesante obtener los márgenes entre los que espera que esté la variable $Y$, conocidos los regresores $X_i$. Si en la introducción indicamos que deseábamos encontrar una función $g$ tal que la variable aleatoria $g(X)$ fuese una estimación aceptable de $E[Y \mid X]$, ahora buscaremos una aplicación multivaluada $\Gamma_\alpha : \mathrm{Im}(X) \longrightarrow I(\mathbb{R})$, donde $I(\mathbb{R})$ es el conjunto formado por los intervalos cerrados de $\mathbb{R}$, tal que el intervalo aleatorio $\Gamma_\alpha \circ X : \Omega \longrightarrow I(\mathbb{R})$ verifique

$$P\{\omega \in \Omega \mid Y(\omega) \in \Gamma_\alpha \circ X(\omega)\} \geq 1 - \alpha$$

para un valor $\alpha$ dado. El que este intervalo sea un buen predictor podría medirse de varias formas distintas.

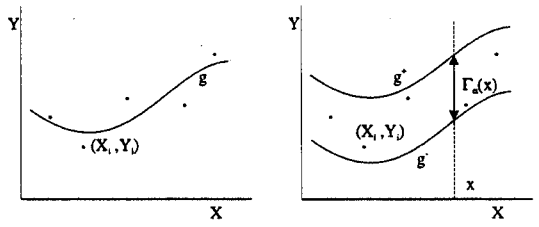



Figura 1: Estimación lineal y por intervalos

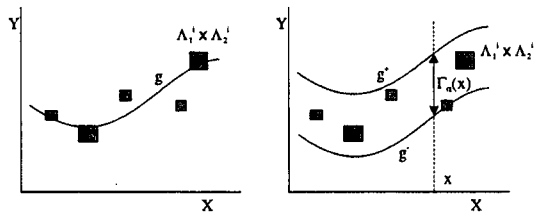



Figura 2: Estimación lineal y por intervalos con información imprecisa

Nos parece razonable afirmar que, dado el valor de $\alpha$, $\Gamma_\alpha$ será tanto mejor cuanto menor sea su longitud media.

Supongamos que los valores de la variable aleatoria $Y$ y del vector aleatorio $X$ no pueden observarse de forma precisa, sino que a lo sumo puede afirmarse que, para un resultado $\omega$ del experimento aleatorio, conocemos que $(X,Y)(\omega) \in \Lambda(\omega)$, donde $\Lambda = \Lambda_1 \times \Lambda_2$ y $\Lambda_1 : \Omega \longrightarrow I(\mathbb{R}^d)$ y $\Lambda_2 : \Omega \longrightarrow I(\mathbb{R})$ son conjuntos aleatorios, y $I(\mathbb{R}^d)$ es el conjunto de los rectángulos en $\mathbb{R}^d$ (ver figura 2).

Definamos ahora dos funciones $g^+$, $g^-$ de forma que $P\{\omega \in \Omega \mid g^- \circ X(\omega) < Y(\omega) < g^+ \circ X(\omega) \forall (X,Y) \in C(\Lambda)\} \geq 1-\alpha$, donde, para cualquier conjunto aleatorio $F$ $C(F) = \{U$ variable aleatoria $\mid U(\omega) \in F(\omega) \quad \forall \omega \in \Omega\}$ es el conjunto de las variables aleatorias *contenidas* en $F$. Deseamos que el margen medio entre $g^+$ y $g^-$ sea lo más estrecho posible para un $\alpha$ dado, pero ahora tenemos la dificultad añadida de que no conocemos $(X,Y)$, sino un conjunto aleatorio

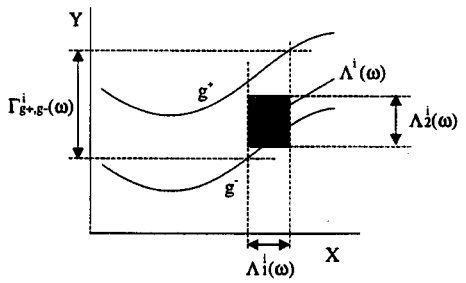



Figura 3: Cálculo de valores de $\Gamma \circ \Lambda_X$

$\Lambda$ que la contiene. Tomando un criterio pesimista, si

$$R = \{(x,y) \in \mathbb{R}^{d+1} \mid g^-(x) < y < g^+(x)\}$$

debemos garantizar que todos los puntos del conjunto $P_\Lambda(R) = \{P_{(X,Y)}(R) \mid (X,Y) \in C(\Lambda)\} = \{t \in [0,1] \mid t = P\{\omega \in \Omega \mid g^- \circ X(\omega) < Y(\omega) < g^+ \circ X(\omega)\}, (X,Y) \in C(\Lambda)\}$ sean mayores que $1-\alpha$. El conjunto $p_\Lambda(R)$ está limitado por los valores $1-\alpha^+$ y $1-\alpha^-$, definidos como

$$1-\alpha^+ = P\{\omega \in \Omega \mid \Lambda(\omega) \subset R\}$$

$$1-\alpha^- = P\{\omega \in \Omega \mid (\Lambda(\omega) \cap R) \neq \emptyset\}$$

por lo que debe cumplirse que $\alpha^+ \leq \alpha$ para que $\min_{(X,Y)\in C(\Lambda)} P\{\omega \in \Omega \mid g^- \circ X(\omega) < Y(\omega) < g^+ \circ X(\omega)\} \geq 1-\alpha$. Es decir, cuando $\alpha^+ \leq \alpha$, las condiciones impuestas al modelo se cumplirán para cualquier variable aleatoria compatible con la observación imprecisa de que disponemos. Las condiciones necesarias para que la cota $\alpha^+$ sea la menor (es decir, para que sea alcanzada por alguna de esas variables) están desarrolladas en [1].

Ahora se desea minimizar la longitud esperada del intervalo aleatorio $\Gamma_{(g^+,g^-)}$ (ver figura 3) $\Gamma_{(g^+,g^-)}(\omega) = \{y \in \mathbb{R} \mid y \in [g^-(x), g^+(x)] \wedge x \in \Lambda_1(\omega)\} = [\min_{x\in\Lambda_1(\omega)} g^-(x), \max_{x\in\Lambda_1(\omega)} g^+(x)]$ (donde la última igualdad se da porque $g^-$ y $g^+$ son continuas y sabemos que $\Gamma_{(g^+,g^-)}$ es medible por la misma razón) sujeto a $1-\alpha \leq \min_{(X,Y)\in C(\Lambda)} P\{\omega \in \Omega \mid g^- \circ X(\omega) < Y(\omega) < g^+ \circ X(\omega)\}$.

Para hacerlo, proponemos la siguiente estimación: sea $(\Lambda^1, \ldots, \Lambda^N) = (\Lambda_1^1 \times \Lambda_2^1, \ldots, \Lambda_1^N \times \Lambda_2^N)$ una muestra aleatoria de tamaño $N$ a partir del conjunto aleatorio $\Lambda = \Lambda_1 \times \Lambda_2$. Dada una realización muestral, un valor $\epsilon > 0$ y una función $h_\theta$ conocida a falta de $m$ parámetros, buscaremos $2m$ constantes $\theta_i^-$, $\theta_i^+$ para las que $[g^-(x), g^+(x)] = \{t \in \mathbb{R} \mid t = h_\theta(x), \theta \in [\theta_1^-, \theta_1^+] \times \ldots \times [\theta_m^-, \theta_m^+]\}$ de forma que la longitud del intervalo

$$\widehat{L}_h = \frac{1}{N}\left[\bigoplus_{i=1}^{N} \Gamma^i_{(g^+,g^-)}\right]$$

sea mínima, donde

$$\Gamma^i_{(g^+,g^-)} = \{y \in \mathbb{R} \mid y \in [g^-(x), g^+(x)] \wedge x \in \Lambda_1^i\}$$

y restringiendo la búsqueda al conjunto de funciones $g^+$, $g^-$ para las que

$$1-\epsilon \leq \frac{1}{N} \#\{i \in \{1 \ldots N\} \mid \Lambda^i \subset R\}.$$

La estimación de $\alpha$ será, para una segunda muestra independiente $(\Lambda_1^{1'} \times \Lambda_2^{1'}, \ldots, \Lambda_1^{M'} \times \Lambda_2^{M'})$, un intervalo de valores $[\hat{\alpha}_M^-, \hat{\alpha}_M^+]$, donde

$$\hat{\alpha}_M^+ = 1 - \frac{1}{M} \#\{i \in \{1 \ldots M\} \mid \Lambda_2^i \subset R\}.$$

$$\hat{\alpha}_M^- = 1 - \frac{1}{M}\#\left\{i \in \{1 \ldots M\} \mid \Lambda_2^i \cap R \neq \emptyset\right\}.$$

Obsérvese que, en realidad, $1 - \hat{\alpha}_M^+$ es una estimación de la creencia (probabilidad inferior) de que un par $(X, Y)$ observado de forma imprecisa pertenezca a la región $R$, y $1 - \hat{\alpha}_M^-$ es una estimación de la plausibilidad (probabilidad superior) del mismo suceso. Es decir, $1 - \hat{\alpha}_M^+ \xrightarrow[M\to\infty]{c.s.} P[\omega \in \Omega \mid \Lambda(\omega) \subseteq R] = \min_{(X,Y)\in C(\Lambda)} P_{(X,Y)}(R)$ y $1 - \hat{\alpha}_M^- \xrightarrow[M\to\infty]{c.s.} P[\omega \in \Omega \mid (\Lambda(\omega) \cap R) \neq \emptyset] = \sup_{(X,Y)\in C(\Lambda)} P_{(X,Y)}(R)$.

# 3 Método numérico de optimización. Algoritmos GA-P

La función multivaluada $\Gamma_{g^+,g^-}$ propuesta en la sección anterior está definida en términos de una función $h$ (cuya expresión analítica aún nos es desconocida), de $2m$ parámetros $\theta_i^-$, $\theta_i^+$; de un estadístico $\widehat{L}_h$ que depende de estos parámetros y de una muestra aleatoria simple construida a partir de un conjunto aleatorio. Dada una realización de esta muestra, para cada expresión de $h$ podríamos minimizar ese estadístico con respecto a los parámetros y de modo que se cumplan además las restricciones mencionadas. De este modo, relacionaríamos cada expresión de $h$ con un mínimo de $\widehat{L}_h$. Nuestro objetivo es encontrar la función $h$ con la que se alcanza el menor de todos estos mínimos, y los valores de los parámetros asociados a ese punto. Así expuesto, es fácil ver que el algoritmo numérico que nos ocupa es similar a un algoritmo de regresión simbólica, por lo que adaptaremos métodos ideados para resolver este último problema.

Los algoritmos GA-P son un método de programación evolutiva, híbrido entre los algoritmos genéticos y la programación genética, optimizado para realizar regresiones simbólicas. Como en cualquier otro método de programación evolutiva, en este algoritmo se mantiene un conjunto de posibles soluciones del problema de regresión (conjunto conocido como *población*) y, mediante determinadas operaciones, se crean nuevas soluciones y se desechan otras hasta que una de ellas es suficientemente buena. La descripción completa del método se encuentra en [3] y no consideramos necesario reproducirla aquí; a modo de resumen, diremos solamente que en un algoritmo GA-P, cada miembro de la población consiste en una cadena de parámetros (parte GA) y en una función (parte GP). En la parte GP se codifica la expresión analítica de la función, definida mediante la combinación de algunos operadores y de varios parámetros numéricos, y que estos últimos están almacenados en la parte GA.

## 3.1 Modificaciones al algoritmo básico

A diferencia de los problemas habituales de regresión simbólica, la función que se desea minimizar no depende directamente de las evaluaciones de la parte expresional en el conjunto de ejemplos. La adecuación de la función $h$ a un conjunto de puntos se mide mediante el promedio de las diferencias entre las funciones $g^+$ y $g^-$, luego para obtener el valor de $g^+(x)$ debería buscarse el máximo de $h$ dentro del rango permitido para sus parámetros,

$$g^+(x) = \max_{\mathbb{R}}\{h_\theta(x), \theta \in [\theta_1^-, \theta_1^+] \times \ldots \times [\theta_m^-, \theta_m^+]\}$$

y lo propio podríamos decir de $g^-$. Afortunadamente, los cálculos numéricos de este mínimo y de este máximo pueden evitarse con una codificación adecuada de la parte expresional del algoritmo GA-P. Esta codificación consiste en emplear aritmética de intervalos para realizar todas las operaciones envueltas en la parte expresional. Es decir, si $h$ se codifica mediante un árbol en el que los nodos terminales correspondan a intervalos $[\theta_i^-, \theta_i^+]$ y a las proyecciones de $\Lambda_1$ (que son rangos de valores $[X_j^-, X_j^+]$, $j = 1, \ldots, d$) y los nodos intermedios a operadores unarios $O_u(A) = \{x \in \mathbb{R} \mid x = o_u(t) \wedge t \in A\}$ o binarios $O_b(A, B) = \{x \in \mathbb{R} \mid x = o_b(t, u) \wedge t \in A, u \in B\}$ donde $A, B \in I(\mathbb{R})$, $o_a : \mathbb{R} \longrightarrow \mathbb{R}$ y $o_b : \mathbb{R} \times \mathbb{R} \longrightarrow \mathbb{R}$, la evaluación de la parte expresional de un individuo siempre será un intervalo. Más aún, eligiendo los operadores $O_u$ y $O_b$ de forma que su evaluación dependa solamente de los extremos de sus argumentos, el número de operaciones necesarias para evaluar el valor medio de la longitud de $\Gamma_{(g^+,g^-)}$ será proporcional al número de operaciones necesario para evaluar $h(x)$, con lo que el tiempo de convergencia será del mismo orden que en los algoritmos convencionales.

Los únicos operadores unarios que hemos empleado son el cambio de signo, la inversión y el logaritmo, y entre los binarios, hemos elegido a la suma, la resta, el producto, la división y la potencia. Las definiciones de las operaciones se diferencian ligeramente de las habituales en aritmética de intervalos, ya que se emplean aproximaciones continuas de los operadores.

# 4 Aplicación práctica

Este trabajo fue encargado por la compañía *Hidroeléctrica del Cantábrico, S.A.* con una finalidad concreta, que determinó la estructura del algoritmo de aprendizaje que se ha expuesto. A continuación se resume el problema planteado y después se recogen los resultados numéricos obtenidos.

### 4.1 Antecedentes

Las industrias eléctricas españolas no cobran directamente del usuario final, sino que son copropietarias de la empresa Red Eléctrica Española, que recibe todos los pagos y los redistribuye de acuerdo con varios criterios. Existe cierta disparidad de criterios acerca del sistema de reparto, y por ello actualmente una comisión está revisando algunas de las reglas de funcionamiento de R.E.E.

Concretamente, la empresa que encargó este trabajo cuestiona el sistema de asignación de costes de mantenimiento de la linea eléctrica de baja tensión. Está admitido que el coste del mantenimiento debe depender de la longitud de línea que posea cada compañía. Pero la medición de la longitud de esta línea es cara y difícil, ya que está tendida dentro de núcleos de población y su geometría es muy compleja. Si se tiene en cuenta que una compañía puede servir a más de 10000 núcleos, es comprensible que se empleen fórmulas legales para relacionar la longitud de línea en una población con su número de habitantes y con su radio. Estas fórmulas deberán publicarse en el B.O.E., lo cual obliga a que el tipo de modelo empleado no sea una red neuronal ni un banco de reglas borrosas, por razones obvias. Se necesita entonces un método capaz de buscar una expresión algebraica sencilla a partir de unos datos que además son imprecisos, ya que las mediciones de longitud se realizan sobre planos y sin tener en cuenta las diferencias de cota, que en algunos pueblos de montaña son muy acusadas.

Por otra parte, y dado que el modelo desarrollado será discutido en una ponencia en la que los demás copropietarios de R.E.E. plantearán modelos alternativos, la empresa exigió un intervalo de confianza para las predicciones. Estos requisitos (modelo algebraico sencillo, datos imprecisos, intervalo de confianza del resultado) justificaron el desarrollo de un nuevo modelo que extienda los GA-P a datos intervalares.

### 4.2 Metodología

En primer lugar se estudió el problema físico y se extrajeron ciertas conclusiones acerca de las dependencias entre las variables [9]. Teniendo en cuenta esta información, y aplicando técnicas clásicas de regresión lineal, lineal por tramos y no lineal, el mejor compromiso entre simplicidad y precisión se alcanzó con un modelo exponencial

$$\frac{\tilde{l}_i}{R_i} = 1.23 A_i^{0.2883} \qquad (1)$$

donde $\tilde{l}_i$ es la longitud estimada en el núcleo $i$, $R_i$ es el radio de esa población y $A_i$ su número de habitantes.

Aplicando algoritmos GA-P se puede obtener una fórmula comparable en complejidad a la anterior y que se ajuste más adecuadamente a la muestra. Limitaremos los árboles a un tamaño de 20 nodos y el número de parámetros a 10. Las operaciones permitidas son la suma, la resta, la multiplicación, la división, la potencia y la raíz cuadrada. El resto de las decisiones (cuyo sentido se define en [6][3][7]) se muestra en la tabla siguiente:

| Parámetro | Decisión |
|---|---|
| Población | 100 |
| Máx gener. | 1000 (rég. p.) |
| Ext. padres | (Ver texto) |
| Cod. GA | Real |
| Cruce GA | En dos puntos |
| Cruce GP | Koza |
| P. cruce GA | 0.9 |
| P. p. internos GP | 0.9 |
| P. cruce GP | 0.8 |
| P. mutación GA | 0.01 |
| P. mutación GP | 0.01 |
| Comp. máx. GP | 20 nodos |
| Comp. máx. GP inicial | 20 nodos |
| Máx. Param. | 10 |
| Enriq. pob. inicial | 1000 individuos |
| Prob. edición | 0 |
| Prob. encaps. | 0 |
| Prob. perm. | 0 |
| Decimación | No |
| Optimización local GA | Poliedro flexible |

Para seleccionar las parejas de cromosomas que tendrán descendencia se eligen tres individuos. Los dos mejores se cruzan y, con una cierta probabilidad, se reemplaza su mejor descendiente por el tercero de ellos o bien se inserta en la población en función de su adecuación (con lo que puede ocurrir que se elimine el peor individuo de la población o que este descendiente se elimine directamente y el cruce no modifique la población). Obsérvese que esta estrategia es elitista y que intrínsecamente se realiza sobreselección. La renovación generacional es en régimen permanente.

## 5 Comparación con otros métodos

Para realizar la comparación de los métodos teóricos y de los algoritmos de modelado numérico hemos dividido la muestra en dos partes de 246 y 245 patrones. Los parámetros de cada modelo (y, en el caso de los algoritmos GA-P, también la expresión analítica de éste) se han ajustado al primero de los subconjuntos, y el error cuadrático medio de este ajuste se tabula en la columna *entrenamiento*. La estimación del error cuadrático medio se hace con el segundo conjunto (columna *test*). La columna *complejidad* recoge el número

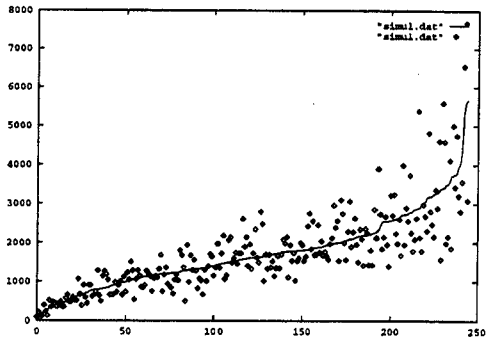


Figura 4: Residuo Modelo GA-P en conjunto de test

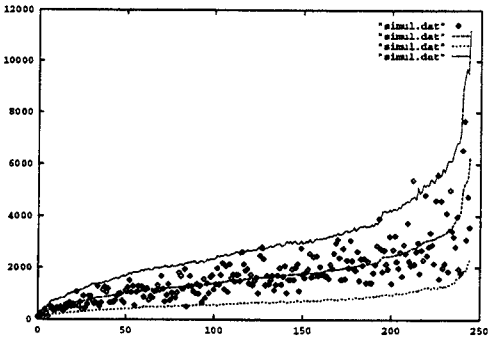


Figura 5: Bandas valores GA-P por intervalos en conjunto de test

de parámetros y de nodos en el árbol que codifica la expresión.

| Método | Entr | Test | Par. | Nod. |
|---|---|---|---|---|
| Lineal | 365 | 443 | 7 | 2 |
| Lineal, 2 Tramos | 338 | 458 | 17 | 6 |
| Exponencial (1) | 342 | 426 | 7 | 2 |
| Polinomio ord. 2 | 332 | 393 | 22 | 6 |
| Polinomio ord. 3 | 318 | 941 | 53 | 10 |
| GA-P | 317 | 367 | 20 | 4 |
| GA-P Intervalos | 341 | 424 | 8 | 2 |
| GA-P I. (corregido) | 306 | 385 | 8 | 2 |
| Red Neuronal 2-25-1 | 312 | 391 | 192[1] | - |
| Wang y Mendel | 262 | 610 | - | 22[2] |
| TSK | 272 | 462 | - | 34[2] |
| RSB [10] | 241 | 410 | - | 39[2] |

[1] Número de pesos

[2] Número de reglas borrosas

El modelo GA-P con aritmética de intervalos se ha evaluado en todos los puntos del conjunto de test en primer lugar, y a continuación se estimó el error en los puntos de test cubiertos. Se empleó un valor $\epsilon = 0.01$ y se obtuvo una estimación $1 - \hat{\alpha}^M = 0.975$ para la creencia de que el modelo cubre al resultado. El valor puntual de error mostrado en la tabla anterior se calculó reemplazando cada intervalo de valores por su punto medio. El modelo descubierto por el método GA-P con intervalos es

$$k_1 R_i k_2^{(A_i^{0.0625})}$$

con $k_1 \in [13.56, 53.47]$ y $k_2 \in [0.058, 0.064]$

## 6 Conclusiones

Los algoritmos GA-P son capaces de descubrir automáticamente una ley empírica a partir de un conjunto de puntos. El método es muy flexible y permite fijar un máximo de complejidad en la expresión, un máximo número de parámetros y un conjunto arbitrario de operaciones.

En este trabajo se ha dado un paso más y se ha adaptado este algoritmo de regresión simbólica a un problema más general: el ajuste de un modelo a un conjunto de puntos cuando éstos, por estar observados con imprecisión, se representan mediante intervalos. El método produce estimaciones multivaluadas para los parámetros que, como se ha visto, pueden convertirse fácilmente en estimaciones puntuales robustas de los mismos y aporta información acerca de la probabilidad de que el intervalo obtenido contenga al valor real que se desea predecir.

## Referencias


[1] Couso, I. (1997) *La Envolvente Probabilística. Definición y Propiedades.* Trabajo de Investigación. Universidad de Oviedo. Departamento de Estadística.

[2] Fonseca, C., Fleming, P.J. (1995) "An Overview of Evolutionary Algorithms in Multiobjective Optimization". Evolutionary Computation 3, 1-16

[3] Howard, L.; D'Angelo, D. "The GA-P: A Genetic Algorithm and Genetic Programming Hybrid" (1995) IEEE Expert. June 1995. 11-15.

[4] Iba, H.,Sato, T. De Garis, H. "System Identification approach to genetic programming". Proc First IEEE Conf on Evolutionary Computation. 401-406. vol 1

[5] Ishibuchi, H., Tanaka, H., Okada, H. (1993) "An architecture of neural networks with interval weights and its application to fuzzy regression analysis". Fuzzy Sets and Systems 57. 27-39

[6] Koza, J., Rice, J. (1994) *Genetic Programming.* MIT Press.

[7] Michaliewicz, Z. (1992) *Genetic Algorithms + Data Structures = Evolution Programs.* Sprinter-Verlag

[8] Renders, J.M., Flasse, S. P. (1996) "Hybrid Methods Using Genetic Algorithms for Global Optimization". IEEE Transactions on SMC. Part B: Cybernetics. Vol 26, NO. 2, April 1996.

[9] Sánchez, L. (1997) "Estudio de la red asturiana de baja tensión rural y urbana". Hidroeléctrica del Cantábrico I+D.

[10] Sánchez, L. (1997) "A Random Sets Based Method for Identifying Fuzzy Models". To appear in Fuzzy Sets and Systems.

# Una metodología para el desarrollo de SBC

**J.L. Castro, J.M. Zurita**
Dpto. Ciencias de la Computación e I.A.
Universidad de Granada
18071 Granada (Spain)
castro@robinson.ugr.es

**J.J. Castro-Schez**
Caja Rural de Granada
C/ Circunvalación, 2
18006.Granada

**A. Espin**
Dpto. Ingeniería Civil
Universidad de Granada
18071 Granada (Spain)
aespin@platon.ugr.es

## Resumen

Este trabajo presenta una metodología para el desarrollo de SBC difusos que, permitiendo una total automatización, sirva para abordar problemas más complejos que aquellos donde las técnicas de aprendizaje automático a partir de ejemplos tienen éxito.

**Palabras Clave:** Sistemas Basados en el Conocimiento, Adquisición del Conocimiento, Aprendizaje Automático.

## 1 INTRODUCCION

El desarrollo de Sistemas Basados en el Conocimiento (SBC) es uno de los temas prioritarios de la politica cientifica Nacional y Europea. De hecho en los últimos años el número de grupos de investigación dedicados a este tema, el número de Shells tanto comerciales como de disposición gratuita para el diseño de SBC, y el número de técnicas y herramientas para el desarrollo de SBC se ha incrementado notablemente. En la actualidad uno de los campos de mayor interés es el desarrollo de herramientas y técnicas que automatizen el diseño de SBC, especialmente en la tarea de adquisición del conocimiento. Estas tienen por objetivo ayudar al Ingeniero del Conocimiento a diseñar el SBC, e incluso algunas el poder desarrollar de forma automática el SBC, interactuando con el experto.

Una importante característica de los SBC es la necesidad de manejar la imprecisión e incertidumbre inherente en el conocimiento humano. En este sentido, la lógica difusa ha demostrado (ver [8]) ser una herramienta muy efectiva (en algunos casos indispensable) para la resolución de problemas donde la información y/o los datos estan afectados de incertidumbre e imprecisión no probabilistica. Ello es debido principalmente a tres factores:

a) la simplicidad y naturalidad como mecanismo de representación del conocimiento, al permitir trasladar expresiones del lenguaje natural a reglas difusas [16],

b) la efectividad, demostrada tanto teóricamente [Castro96] como en la práctica [8],

c) la capacidad de manejar la imprecisión y la incertidumbre de forma simple [15].

Así, resulta natural que se hayan diseñado numerosos SBC utilizando lógica difusa (ver [7], [11], [1]). Estos sistemas utilizan la lógica difusa como modelo de representación del conocimiento y la inferencia difusa para diseñar el motor de inferencia. Cabe destacar que también existen herramientas para el diseño de SBC utilizando lógica difusa, y númerosos trabajos dedicados al estudio de los Sistemas Basados en el Conocimiento difuso.

Entre los trabajos dedicados al estudio de SBC difusos, tenemos bastantes trabajos dedicados al aprendizaje automático de reglas difusas a partir de ejemplos [13], [14], [12], [6], [5], [10] (Machine Learning session). También podemos destacar que hay varios proyectos de investigación dedicados a este tema, por ejemplo el TIC96-0778 "Aprendizaje de sistemas basados en reglas difusas utilizando Algoritmos Genéticos y Redes Neuronales" y el proyecto TIC95-0453 "Aprendizaje en entornos imprecisos e inciertos. Aplicaciones".

En cuanto al diseño de herramientas que automatizen en parte o en general el desarrollo de SBC difusos, una notable aportación la tenemos en la shell MILORD realizada en el IIA del CSIC de Barcelona (ver [11] para una breve descripción). Esta shell incorpora un módulo de elicitación del conocimiento mediante el sistema EAR, Elicita-Analiza-Refina, y basado principalmente en la "Personal Construct Theory" de Kelly. La elicitación se basa en un "repertory grid" difuso donde a cada "constructo" se le asigna un valor lingüístico cuando se aplica a un elemento del dominio. Esto conduce a un "repertory grid" de "referencias cruzadas" entre "constructos" y elementos del dominio. Entonces

el sistema analiza de forma interactiva el "repertory grid" para detectar:' a) elemetos pobremente o ambiguamente definidos y b) elementos que el experto aún no ha tenido en cuenta. Estos problemas se resuelven introduciendo nuevos constructos o elementos.

En una segunda etapa se realiza un análisis lógico del "repertory grid" de la que se obtiene un conjunto inicial de reglas difusas en forma de red de inferencia. En la fase de refinamiento se presenta esta red al experto para que muestre sus desacuerdos y se usan diversas técnicas de la "Personal Construct Theory" para corregir la base de reglas. El ciclo termina cuando no hay ningún desacuerdo.

Una segunda herramienta para automatizar en parte el desarrollo de SBC difusos es KAFES [9]. Esta es una shell de adquisición del conocimiento para sistemas expertos difusos basada también en un "repertory grid" difuso, similar en cuanto a filosofía a la de MILORD, y que esta diseñada para construir el sistema experto difuso sobre la shell CLIPS. La diferencia con MILORD consiste en que no realiza el ciclo Elicita-Analiza-Refina, las reglas difusas tienen una estructura diferente, e incorpora un módulo para describir las funciones de pertenencia de las etiquetas difusas.

Otra aportación en cuanto a estas herramientas la tenemos en la metodología propuesta por Slany [10] (Applications I session)] para utilizar CheckFLIP++ como una herramienta de AC para sistemas basados en restricciones difusas.

Los trabajos y proyectos de investicación de aprendizaje automático de reglas difusas anteriormente citados, son muy útiles para resolver problemas no demasiado complejos, donde todos ellos obtienen unos resultados muy similares. Cuando los problemas poseen una estructura más compleja, estos algoritmos tienen unos resultados menos satisfactorios. En nuestra opinión ello es debido a que pretenden aprender todo el sistema con una estructura de reglas excesivamente simple y sin sacar partido de la capacidad de la lógica difusa para representar el conocimiento humano y así adquirir conocimiento de un experto. Por contra, los sistemas de adquisición del conocimiento de MILORD y KAFES no utilizan las ténicas de aprendizaje automático a partir de ejemplos. Además, el hecho de estar diseñados para desarrollar SBC que pueden ser muy complejos va en detrimento de una automatización completa, siendo necesaria la intervención de un ingeniero del conocimiento.

Este trabajo presenta una metodología para el desarrollo de SBC difusos que, permitiendo una total automatización, sirva para abordar problemas más complejos que aquellos donde las técnicas de aprendizaje automático a partir de ejemplos tienen éxito. Por supuesto, el hecho de tener por objetivo una posible automatización completa va a hacer que no sea aplicable (al menos como sistema automático) a problemas excesivamente complejos. En definitiva queremos diseñar una metodología intermedia entre las dos metodologías anteriormente citadas y que combine las dos principales facetas de ambas: la automatización y la posibilidad de abordar problemas complejos mediante adquisición del conocimiento de expertos.

## 2 PLANTEAMIENTO DEL PROBLEMA

Partimos de la hipótesis de que la metodología esta destinada a disenar un SBC para resolver un problema de clasificación (o decisión). Un problema de este tipo consiste en decidir a que clase pertenece un objeto (o que decisión tomar) a partir de cierta información de las propiedades del objeto (o de la información de entrada). La información vendrá dada por los valores que toman una serie de variables de entrada $\{X_1, ..., X_n\}$, y el resultado del sistema será a que clase pertenece entre un conjunto de clases prefijadas $y_1, ..., y_d$.

Los valores de las variables de entrada podrán ser precisos (valores concretos) o imprecisos (expresiones linguisticas que se interpretaran como conjuntos difusos). Además podran ser discretas o continuas, pero cada $X_i$ dentro de un rango conocido $[a_i, b_i]$ en el caso continuo, o $\{V_i^1, ...., V_i^{n_i}\}$ en el caso discreto.

Supondremos que cada variable $V_i$ tiene un conjunto de posibles atributos $\mathcal{L} = \{L_i/i \in I_i\}$. Este conjunto de atributos puden variar desde la familia de los subconjuntos difusos en el dominio de $V_i$, el el caso de que no partamos de un conjunto de atributos prefijados, o simplemente una subfamilia finita de esta (etiquetas) en el caso de que partamos de un conjunto de atributos prefijados.

Se podrán crear nuevas variables que sean necesarias para el correcto diseño del SBC.

La idea es diseñar el SBC como un conjunto de reglas difusas tal que en el caso de poder disparar varias reglas distintas se adopta el criterio de quedarse con la decisión de la regla con mayor grado de acoplamiento a la entrada. Las reglas tendran la forma:

> IF $X_1$ is [conjunto de atributos] and ... and $X_n$ is [conjunto de atributos] THEN Y is [decisión].

Este tipo de reglas incluyen casos de reglas muy simples como por ejemplo, para el caso de 3 variables de entrada:

> IF $X_3$ is Grande THEN Y is $y_1$.

Pues esta regla tiene la expresion equivalente:

> IF $X_1$ is [conjunto total de atributos] and $X_2$ is [conjunto total de atributos] and $X_3$ is Grande THEN Y is $y_1$.

El grado de acoplamiento se realizara como es usual en muchos casos utilizando la t-norma del mínimo. Obviamente, cualquier otra t-norma puede ser utilizada con tal fin.

# 3 METODOLOGIA PARA EL DISEÑO DEL SBC CON LA COLABORACION DE UN EXPERTO

En este caso, para diseñar el sistema se tendrá en cuenta dos tipos de información: a) Un fichero de ejemplos de la respuesta correcta para una serie de valores de entrada, y b) Un experto en el problema en cuestión.

La metodología que se propone es la siguiente:

1.- **Fase de extraccion de variables relevantes.** Se extraen las variables relevantes al problema, sus dominios y atributos, mediante la información proporcionada por el experto a traves de entrevistas:

   - Se realiza una primera entrevista al experto donde se le pide que indique como suele resolver el problema y se realiza una transcripción de la misma.
   - Se extraen los substantivos utilizados por el experto (candidatos a variables) y los adjetivos aplicados a cada sustantivo (candidatos a atributos).
   - Se realiza una segunda entrevista donde se le consulta si realmente esos sustantivos corresponden a variables utilizadas en el proceso; y los posibles atributos de cada variable, utilizando como ejemplo los adjetivos utilizados en la primera entrevista para cada atributo.
   - Se determinan las variables, su tipo, y sus posibles atributos.

2.- **Fase de elección de un conjunto representativo de ejemplos.** Se particionan los ejemplos en dos grupos, el 80% para adquisición del conocimiento y el 20% para verificación. Esto se realiza con la colaboración del experto, cuya mision sera el asegurar que en los ejemplos de verificación no son sino casos similares a alguno de los de adquisición, y que estos recogen el expectro completo de casos típicos.

3.- **Fase de adquisición del conocimiento.** Para cada ejemplo de prueba se solicita al experto una explicación del por qué de la respuesta dada. Esta explicación se traduce en una regla que soluciona y explica el ejemplo y los ejemplos similares al mismo.

   Cuando se aborda un nuevo ejemplo, se observa si ya se puede resolver con el conocimiento ya adquirido y se compara la solución y/o explicación con la proporcionada por el experto. Pueden ocurrir entonces los siguientes casos:

   - La solución y explicación coincide con la proporcionada por el experto. En este caso se pasa al siguiente ejemplo sin hacer nada.

   - La solución coincide pero la explicación no. En este caso se incluye esa nueva explicación como una nueva regla, pues puede ocurrir que nos permita resolver casos distintos.

   - Ni la solución ni la explicación coinciden. En este caso tenemos dos (o más) reglas que producen inconsistencia. La solución a este conflicto tiene dos alternativas:

     - Hay que especificar mas alguna o todas las reglas introduciendo mas antecentes o especificando mas los atributos de las variables que intervienen en el antecedente.
     - Las reglas son validas en general, pero cuando pueden dispararse conjuntamente hay que aplicar otro criterio (razonamiento no monótono). En este caso se dejan ambas reglas, y se realiza un nuevo módulo al que se accedera solo en el caso de que ambas puedan dispararse y dedicado a resolver ese caso particular. Esta solución puede ir desde el ponderar ambas reglas y quedarse con la solución proporcionada por la regla que de un mayor matching ponderado, hasta introducir nuevas reglas para estos casos particulares.

   En cualquiera de los anteriores casos puede haber la necesidad de considerar nuevas variables que hasta ese momento no habian sido consideradas.

   Cuando se ha realizado este proceso, tenemos una base de reglas que **explica todos los ejemplos de adquisición del conocimiento tal y como lo haría el experto.** Pero no se puede asegurar ni la completitud, ni la consistencia, pues puede haber entradas para las que no se disponía de ningun ejemplo donde no habría respuesta o que podría producir inconsistencia. En busca de la completitud y consistencia se realiza la siguiente etapa del proceso:

4.- **Fase de Validación.** Mediante el método de validación publicado en [2], se van extrayendo en-

tradas donde el sitema aún no produciría respuesta o entradas que producirián inconsistencia. Para cada una de estas entradas se le consulta al experto, pudiendo ocurrir nuevamente dos cosas:

- Esa entrada no se da en la vida real. En este caso no se hace nada.

- Esa entrada se puede dar en la vida real. En este caso se procede como si fuese un ejemplo de adquisición en el paso anterior.

Cuando se ha realizado este proceso, tenemos una base de reglas que es **consistente, completo,** y que **explican todos los ejemplos de adquisición del conocimiento tal y como lo haría el experto.**

5.- **Fase de Verificación.** Para cada uno de los ejemplos de verificación se dispara el sistema para ver si la respuesta y explicación del mismo coincide con la esperada. Si coincide, no se modifica nada, y si no coincide se procede como en el caso de la fase de adquisición del conocimiento cuando se produce un conflicto.

# 4 METODOLOGIA PARA EL APRENDIZAJE AUTOMATICO DEL SBC A PARTIR DE EJEMPLOS

En este caso, partimos de la base de que solo se cuenta con un fichero de ejemplos de la respuesta correcta para una serie de valores de entrada, y que tanto las variables de entrada como las de salida son conocidas.

La idea es utilizar la misma metodología que para el caso anterior sin más que cambiar la intervención del experto por un método de aprendizaje automático de reglas difusas.

La metodología que se propone es la siguiente:

1.- **Fase de diseño de etiquetas.** Se eligen las etiquetas que van a representar los atributos difusos a considerar para cada variable.

2.- **Fase de elección de un conjunto de ejemplos.** Se particionan los ejemplos en dos grupos, el 70% para aprendizaje, y el 30% para prueba.

3.- **Fase de Aprendizaje.** Para cada ejemplo de prueba se "aprende" la regla difusa más general (mas simple) que cubre dicho ejemplo mediante el algoritmo de [10], [5]. Este algoritmo proporciona una regla solo si los ejemplos contradictorios tienen un grado de acoplamiento $< \epsilon$ (reglas validas).

Cuando se aborda un nuevo ejemplo, se observa si ya se puede resolver con las reglas ya aprendidas con un grado de acoplamiento $\geq \epsilon$. Si es asi, claramente responderá lo mismo que la regla ya aprendida y se aprendería otra vez la misma regla, así que se pasa al siguiente ejemplo. Si no es así, se aprende la regla correspondiente a dicho ejemplo (si es que existe) y se comparan las respuestas de ambas reglas, pudiendo ocurrir los siguientes casos:

- Ambas respuestas coinciden. En este caso se incluye esa nueva regla, pues puede ocurrir que nos permita resolver casos distintos de la regla ya aprendida.

- Ambas respuestas son distintas. En este caso tenemos dos (o más) reglas que producen inconsistencia. La solución a este conflicto se resuelve analizando y aprendiendo nuevas reglas, que se dispararan tan solo cuando ambas reglas entren en conflicto y que por tanto se aprenden solo con los ejemplos que entren dentro de este conflicto.

4.- **Fase de Validación.** Mediante el método de validación publicado en [2], se van extrayendo entradas donde el sitema aún no produciría respuesta o se produciría inconsistencia, y se comprueba si existen ejemplos con esa entradas. Si no hay ejemplos no se hace nada. Si hay ejemplos, se adopta la respuesta con mayor número de ejemplos para esa entrada y se realiza el proceso de solución de conflictos del paso anterior.

5.- **Fase de Verificación.** Para cada uno de los ejemplos de verificación se dispara el sistema para ver si la respuesta del mismo y la explicación coincide con la esperada. Esto nos da una medida de la "bondad" del aprendizaje.

# 5 UN EJEMPLO

Hemos realizado una prueba muy simple para comprobar si la metodología propuesta podría dar buenos resultados. En concreto la prueba consistía en diseñar un SBC para clasificar la planta Iris en sus tres tipos básicos (Iris-Versicolor, Iris-Setosa e Iris-Virginica) a partir de la longitud y anchura del pétalo y del sépalo. Como base de ejemplos utilizamos el 70% de la base de datos *Iris Plant Database* creada por R.A. Fisher y utilizada en gran cantidad de publicaciones. El restante 30% lo dejamos para prueba del método. La división de los dominios de las variables se realizó en tres "términos lingüisticos": Pequeño, Mediano y Grande; establecidos dividiendo los respectivos dominos en tres partes iguales y modelizando cada término como etiquetas difusas trapezoidales.

El algoritmo de aprendizaje de reglas válidas y simples (con $\epsilon = 0.3$) obtuvo tres reglas:

1.1) Si la anchura del pétalo es pequeña entonces el tipo es Iris-setosa

1.2) Si la anchura del pétalo es grande entonces el tipo es Iris-Virginica

1.3) Si la longitud del pétalo es mediano entonces el tipo de Iris-Versicolor

Posteriormente, la fase de validación nos presentó dos conflictos reales (de los que si había ejemplos):

a) Si la anchura del pétalo es mediano y la longitud del pétalo es grande no había respuesta,

b) Si la longitud del pétalo es mediano y la anchura del pétalo es grande se producía una inconsistencia.

El análisis del conflicto a) nos llevo a introducir una nueva regla en el módulo inicial:

1.4) Si la longitud del pétalo es mediano y la anchura del pétalo es grande entonces el tipo es Iris-Virginica.

El análisis del conflicto b) nos llevo a introducir un nuevo módulo al que se accedería sólo en el caso de dispararse las reglas 1.2 y 1.3 con valores similares (diferencia $\leq 0.2$). Este nuevo módulo tendría las siguientes dos reglas:

2.1) Si la anchura del sépalo es mediano entonces el tipo es Iris-Versicolor

2.2) Si la anchura del sépalo es pequeña entonces el tipo es Iris-Virginica

El resultado de este simplísimo SBC fué muy satisfactorio. Acertaba con todos los ejemplos de prueba y fallaba solamente con uno de los ejemplos de aprendizaje. Finalmente comparamos estos resultados con los algoritmos de aprendizaje clásicos (Arbolés de Decisión, Redes Neuronales, Regresión Logística) y los métodos de aprendizaje de reglas difusas desarrollados en el Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada; todos ellos aplicados sobre esa misma partición de la base de ejemplos. El SBC desarrollado obtenia un acierto mayor que el mejor de los métodos comprobados (99.4% frente a 98.7%), y además con unas reglas mucho mas simples que las obtenidas por los métodos de extracción de reglas.

## 6 CONCLUSION

En este trabajo presentamos una metodología para el diseño de SBC difusos con las siguientes carastericticas:

- El diseño queda parcialmente mecanizado.
- Se puede aplicar para aprendizaje automático
- Utiliza razonamiento no-monótono
- El diseño es modular.

Además lo hemos aplicado a un ejemplo muy simple de aprendizaje automático donde los resultados son alentadores.

## Referencias


[1] **Campbell, J.G.** *Fuzzy Rule-Based Software Modules - Programmers' Guide.* Interactive Systems Centre, Report isc/94021/R, University of Ulster, 1995.

[2] **Castro, J.L., Trillas, E., Zurita, J.M.** *Expert Systems Validation.* Proceedings of the sixth IFSA Congress, Sao Paulo, Brasil, 1995, 41-44.

[3] **Castro, J.L., Delgado, M.** *Fuzzy Systems with defuzzification are Universal Approximators,* IEEE Trans. on Syst, Man. and Cybern. 26-01, 1996, pp. 149-152.

[4] **Castro, J.L., Trillas, E., Zurita, J.M.** *Non Monotonic Fuzzy Reasoning,* por aparecer en Fuzzy Sets and Systems. 1997a.

[5] **Castro, J.L., Castro, J.J., Zurita, J.M.** *Learning maximal structure rules in fuzzy logic for knowledge acquisition in expert systems.* Por aparecer en Fuzzy Sets and Systems. 1997b.

[6] **Gonzalez, A.** *A learning methodology in uncertain and imprecise environments,* International Journal of Intelligent Systems 10, (1995) 357-371.

[7] **Hall, L.O., Kandel, A.** *Designing Fuzzy Expert Systems,* Verlag TÜV Rheinland, Germany, 1986.

[8] **Hirota, K.** *Industrial Applications of Fuzzy Technologies,* Springer Verlag 1993.

[9] **Hwang, G.J.** *Knowledge Acquisition for fuzzy Expert Systems,* International Journal of Intelligent Systems 10, (1995) 541-560.

[10] **IPMU96** *Proceedings of the sixth IPMU Congress,* Granada, España, 1996.

[11] **Kandel, A, Ed.** *Fuzzy Expert Systems,* CRC Press, 1992.

[12] **Sestito, S., Dillon, T.,** *Knowledge Adquisition of Conjunctive Rules Using Multilayered Neural Networks,* Int. J. Intelligent Systems, 8, (1993), 779-805.

[13] **Sugeno, M., Tanaka, K.** *Succesive identification of a fuzzy model and its applications to prediction of a complex systems*, Fuzzy Sets and Systems 42 (1991) pp. 315-334.

[14] **Wang, L., Mendel, J.M.**, *Generating fuzzy rules by learning from examples*, IEEE Trans. on Syst, Man. and Cybern. 22, 1992, pp. 1414-1427.

[15] **Zadeh, L.A.**, *The role of fuzzy logic in the management of uncertainty in expert systems.* Fuzzy Sets and Systems, 11, 197-227, 1983.

[16] **Zadeh, L.A.**, *Fuzzy Logic, Neural Network and Soft Computing*, CACM 37-3, 1994.

# SISTEMA EXPERTO FUZZY PARA EVALUACIÓN Y ENTRENAMIENTO FÍSICO

**Nuria Juan Sierra**
NJS00001@teleline.es

**Dpto. Informática y Automática**
Facultad de Ciencias Físicas
Universidad Complutense de Madrid

**Matilde Santos Peñas**
msantos@eucmax.sim.ucm.es

## Resumen

En este trabajo se presenta el diseño y la implementación de un sistema experto interactivo que trabaja con información fuzzy en el ámbito del ejercicio físico como terapia. Partiendo de un análisis cualitativo sobre el estado físico de un individuo, se elabora un diagnóstico sobre esta evaluación y se genera un plan de entrenamiento personalizado dependiendo de la edad, sexo, deporte al que quiera orientar su formación, y diagnostico sobre las características físicas que haya demostrado, que le lleve a mejorar su condición física.

## 1 INTRODUCCIÓN

El objetivo de este artículo es presentar un "Sistema Experto Fuzzy para Evaluación y Entrenamiento Físico".

Funcionalmente, el sistema desarrollado proporciona una herramienta para realizar la evaluación de la condición física de una persona, y como consecuencia recomienda un plan de entrenamiento a medida. Con esta finalidad, propone inicialmente una serie de pruebas y analiza los resultados obtenidos para un individuo mediante una serie de evaluaciones fuzzy parciales referentes a rendimiento cardiovascular, grado de entrenamiento y eficiencia muscular. Como consecuencia, dependiendo de la edad, sexo, deporte al que quiera orientar su formación, y diagnostico sobre las características físicas que haya demostrado, se le propondrá un plan de entrenamiento personalizado.

A este programa se le puede denominar sistema experto porque se ha diseñado para realizar tareas tradicionalmente asignadas a expertos humanos: evaluadores y entrenadores deportivos. Este sistema de decisión, como sistema fuzzy frente a un sistema de lógica clásica, presenta la ventaja importante de permitir una representación del conocimiento afín con la vaguedad con la que trabajan estos expertos y con la naturaleza de las evaluaciones que se realizan, incorporando la imprecisión propia del leguaje humano que permite la lógica fuzzy [3], [5]. La principal fuente de información y experiencia que se ha utilizado para especificar este sistema procede de un grupo de expertos entrenadores del Club Deportivo "Real CANOE Natación Club" de Madrid.

En el presente artículo, el apartado 2 se dedica al enfoque del Entrenador, donde se realiza un profundo planteamiento del objetivo del sistema. El análisis realizado permite obtener unas especificaciones detalladas del objetivo, y justifica las razones por las que resulta factible y útil desarrollar un sistema de decisión borroso. En el apartado 3 se expone cómo se ha concebido el sistema y el diseño que se ha realizado. Dentro del apartado 4 se detallan las características esenciales de los módulos que componen el programa: Rendimiento Cardiovascular, Grado de Entrenamiento, Eficiencia Muscular. En el apartado 5 se presenta la realización del Diagnóstico de Preparación Física y Plan de Entrenamiento a Medida. Finalmente, se exponen las conclusiones, que representan una breve reflexión acerca del trabajo realizado, y síntesis de algunos puntos discutidos previamente.

## 2 EL ENFOQUE DEL ENTRENADOR

El campo de aplicación al que va destinado el "Sistema Fuzzy para Evaluación y Entrenamiento Físico" corresponde al ámbito de la Educación Física y se ha concebido como un

sistema con capacidad para **asistir** o **sustituir** en ciertas actividades al experto evaluador/entrenador.

En los diferentes ámbitos de la educación física: clubes deportivos, mantenimiento de la condición física y escuelas, la ausencia de criterios normalizados sugeridos por las administraciones origina que las personas responsables del entrenamiento físico utilicen normalmente criterios subjetivos y colectivos, basados en su experiencia personal, para llevar a cabo el plan de preparación física del colectivo total de individuos. Con este sistema se intenta buscar una línea coherente de actuación entre lo que debe hacerse y lo que es posible.

El entrenador que realiza un plan de entrenamiento, normalmente de modo subjetivo, tiene en cuenta la edad, capacidad y preparación del colectivo con el que piensa ponerlo en práctica. Esto en muchos casos resulta insuficiente, y conviene descender a la personalización, es decir, se ha de individualizar el plan o programación física diseñada.

Es interesante hacer notar que el sistema desarrollado permite sustituir al evaluador/entrenador en el sentido de que una persona que tenga acceso al sistema fuzzy de decisión puede seguir los pasos que se le indican interactivamente para **autoevaluarse**, y a continuación iniciar el Plan de Entrenamiento aconsejado por el sistema fuzzy. A su vez, el programa tiene capacidad para asistir al entrenador en el sentido de que le permite uniformizar y estandarizar las tareas de: diagnostico de personas con problemas en su formación física, emisión de criterios acerca del nivel de las personas que inician su actividad física, y control y seguimiento de las mejoras obtenidas **individualmente**. Además, existe una garantía de objetividad en los criterios seguidos.

Es necesario, a su vez, que el entrenador realice un seguimiento acerca de si las acciones o toma de decisiones que inicialmente se concluyen del conjunto de reglas borrosas del sistema son eficaces, y en función de los resultados se podrá ir actualizando el sistema desarrollado y las reglas fuzzy del mismo. Se parte de la idea de que la evaluación adquiere sentido en la medida que comprueba la eficacia del plan de entrenamiento, y posibilita el perfeccionamiento de los procedimientos y planes llevados a cabo.

La evaluación incluida en el ámbito de la educación física en el que se centra este sistema no está asociada a la noción cuantitativa de medida exacta, sino más bien, una aproximación cualitativa. Conviene en este punto hacer un poco de énfasis en la diferencia entre **medición** y **evaluación**, mientras que por medición se entiende una descripción cuantitativa del comportamiento, la evaluación comprende a la vez, la descripción cualitativa y cuantitativa, y comporta además, juicios de valor en relación con lo que se espera. Evaluar constituye pronunciarse sobre esa actuación. La medición constituye una fase de la evaluación que suministra datos, información. La evaluación, al emitir juicios de valor sobre los aspectos medidos, supera la mera recogida de datos. En este concepto de **evaluación cualitativa**, es en definitiva, en el que se basa el sistema desarrollado, y en el que incorpora la lógica borrosa [3].

Así pues, el sistema soportará las acciones que normalmente realiza el entrenador en multitud de ocasiones de un modo inconsciente: recogerá información, emitirá un juicio a partir de esta información y ciertas comparaciones, y tomará una decisión con respecto al tipo de preparación física que tendría que llevar a cabo esta persona.

Una de las razones fundamentales en las que se ha basado el siguiente trabajo para concluir la idoneidad de un sistema experto fuzzy aplicado a este campo, la educación física, es porque precisamente se trata de un ámbito en el que es importante que la evaluación se realice como un proceso sistemático: planificación, aprendizaje, evaluación, y reajuste, en el sentido de que sistemático se opone a improvisado o desordenado, e implica que unas partes están relacionadas con otras, aunque individualmente son independientes.

Otra de las razones que se podría citar es que se trata de un campo en el que, contrariamente a lo que en la práctica habitual se considera, es muy importante la personalización. Individualizar significa tener en cuenta a la persona, las diferencias individuales, y no masificar o uniformar, que supondría plantear las mismas exigencias y el mismo plan de entrenamiento a todo el colectivo al que se dirige el entrenamiento (club deportivo, gimnasio o escuela). El sistema tradicional de agrupamiento para el entrenamiento físico consiste en distribuir a las personas en grupos cerrados, establecidos de acuerdo con un criterio, normalmente de afinidad. Sin embargo, las últimas tendencias marcan las grandes diferencias y singularidades que pueden existir entre personas con una misma afinidad, y la importancia de marcar los diferentes niveles de homogeneidad dentro de personas con una misma afinidad deportiva de una forma mucho más **objetiva** y personalizada.

Los objetivos que pretenderá alcanzar nuestro sistema fuzzy son:

- **Diagnosticar** qué persona y en qué tiene dificultades, y determinar el nivel físico de partida de un individuo en relación con determinados parámetros. La finalidad es

adaptar un entrenamiento a medida que le permita progresar y corregir cualquier deficiencia en el proceso de formación.

- Emitir un **Criterio** acerca de los diferentes aspectos a considerar en cuanto al nivel físico de una persona; éstos permitirán la adopción de decisiones respecto a estrategias de educación física.
- **Clasificar** a las personas dependiendo del juicio de valor final que se haya obtenido acerca de su nivel físico con respecto a unas condiciones específicas o a su condición general.
- **Control** y seguimiento de las mejoras obtenidas individualmente, según los planes de preparación física propuestos. De este modo se puede determinar si una persona posee los niveles mínimos necesarios para abordar un plan de formación física concerniente a una categoría superior, e iniciar un nuevo ciclo de entrenamiento.

Las características básicas y especificas en las que se debería centrar la evaluación son diversas, considerando que ninguna de ellas permite una evaluación completa, ha resultado necesario la utilización conjunta y armónica de varias:

- Características de Rendimiento Cardiovascular
- Características de Grado de Entrenamiento
- Características de Eficiencia Muscular.

La evaluación de estas características se ha seleccionado porque permiten enfocarse hacia los cambios que experimenta el físico de una persona de un modo continuo y dinámico, y por tanto facilita la verificación de los logros adquiridos en función de los objetivos propuestos. En este punto central es dónde el sistema desarrollado aglutina la experiencia de los expertos entrenadores que han marcado las pautas de las especificaciones detalladas.

## 3 EL DISEÑO DEL SISTEMA

Según los expertos consultados el grado de capacidad física de un individuo depende de la integración de múltiples funciones orgánicas. Sin embargo, a grandes rasgos, los factores esenciales que tienen una incidencia directa sobre la capacidad física se pueden dividir en tres grandes grupos:

-Eficiencia orgánica
-Eficiencia muscular
-Eficiencia muscular y orgánica combinada

En nuestro sistema se han seleccionado tres módulos de evaluación cualitativa que son representativos de estos grandes grupos de factores: como representación de los factores orgánicos se ha diseñado un módulo de evaluación del rendimiento cardiovascular; asociado a la eficiencia muscular, un módulo de evaluación de la eficiencia muscular en el que a su vez intervienen diversos factores; y finalmente, como representativo del grupo que combina tanto factores orgánicos como musculares, se ha concebido un módulo de evaluación del grado de entrenamiento (Figura 1).

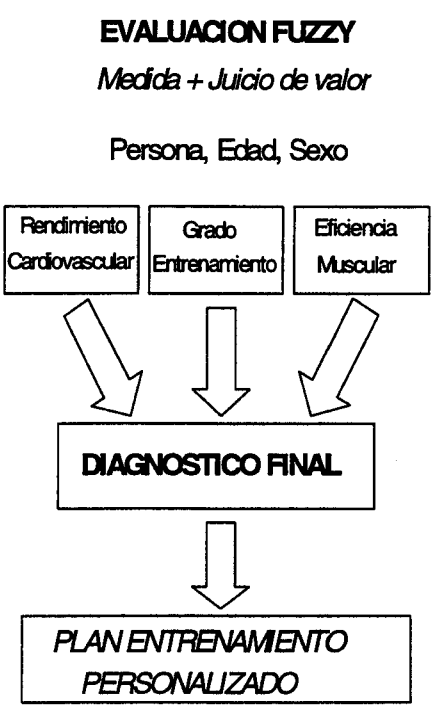



Figura 1: Diseño del Sistema

Los expertos entrenadores siempre hablan en términos cualitativos acerca de estas tres características, pero precisamente para cumplir el objetivo del sistema de decisión se tendrá que ser capaz de definir un modelo y delimitar pautas que permitan establecer criterios objetivos de en qué condiciones se dice que el rendimiento cardiovascular de una persona es mediocre, la eficiencia muscular es buena o su grado de entrenamiento pobre. Conviene señalar que incluso conceptos que podrían ser determinados cuantitativamente son discutidos en **términos cualitativos**. Esto hace muy conveniente la incorporación de la lógica fuzzy para tratar con este tipo de información, así como con los resultados de la evaluación.

Una reflexión en mayor profundidad de los expertos entrenadores permite incluso deducir que, en la mayoría de los casos en que se realiza una evaluación de la condición física de una persona, el criterio de valor que emiten de forma subjetiva está dirigido por procedimientos y reglas consecuencia de la experiencia acumulada.

Incluso los entrenadores expertos manifestaron que sería especialmente ventajoso que la agrupación o clasificación de las personas, atendiendo a la heterogeneidad de las mismas, se realizase según un criterio de homogeneidad que atienda a su rendimiento, eficiencia o habilidad, con la mínima influencia de factores subjetivos. Esto tiene la finalidad de proponer a cada grupo niveles objetivos y homogéneos en el plan de entrenamiento.

Esta evaluación se realizará con cierta periodicidad para contrastar los logros obtenidos y reclasificar las personas, dado que la agrupación única e inalterable de las personas sería una utopía ya que no todas progresarán a un mismo ritmo.

Inicialmente es importante definir la edad y el sexo de una persona, pues tanto el grado de entrenamiento como la eficiencia muscular variará substancialmente dependiendo de estas características. Estas características físicas vienen definidas por conjuntos borrosos, con sus correspondientes etiquetas y funciones de pertenencia asociadas.

El Sistema desarrollado tiene tres módulos de evaluación cualitativa importantes: *cardio_vascular, entrenamiento, y muscular,* de acuerdo al grupo de factores que el experto entrenador declaró como más representativos de la condición física de una persona. Se implementan como *macro_reglas borrosas*, ya que cada una hace referencia a una serie de relaciones borrosas entre las entradas (características físicas) y la salida (evaluación).

El diagnostico final sobre la condición física general de una persona se determina a través del predicado *evaluar (Persona, Deporte, perfil (CardioVascularRendimiento, GradoEntrenamiento, EficienciaMuscular), DiagnosticoFinal),* el cual evalúa el perfil físico del individuo a través del resultado de los tres módulos de evaluación cualitativa. El Plan de Entrenamiento más apropiado a las características básicas y específicas de la persona (su perfil) es entonces definido mediante el predicado *plan (Persona, DiagnosticoFinal, PlanEntrenamiento).*

En *evaluacion_fisica* es importante destacar las características de la formulación del problema a alto nivel y la modularidad de la evaluación parcial planteada como objetivo, de modo que cada módulo se puede desarrollar independientemente, sin afectar al resto del sistema [2].

Las especificaciones de la aplicación no exigen trabajar obligatoriamente con estructuras de datos concretas sino que se utiliza la abstracción de datos. Una estructura *perfil (C,E,M)* representa el perfil físico de la persona en cuanto al rendimiento cardiovascular C, grado de entrenamiento E, y eficiencia muscular M.

La relación básica a alto nivel en el programa desarrollado es:

*EVALUACION_FISICA:*
***Nombre de la persona:***
***Edad:***
*Sexo(hombre/mujer):*
***cardio_vascular****(Persona, CardioVascularRendimiento),*
***entrenamiento****(Persona, Edad, Sexo, GradoEntrenamiento),*
***muscular****(Persona, Edad, Sexo, Deporte, EficienciaMuscular),*
***evaluar****(Persona, Deporte, perfil (CardioVascularRendimiento, GradoEntrenamiento, EficienciaMuscular), DiagnosticoFinal)*
***plan****(Persona, DiagnosticoFinal, PlanEntrenamiento).*

# 4 CARACTERÍSTICAS DE LOS MÓDULOS DE EVALUACIÓN

Antes de introducir cada módulo, es conveniente señalar dos comentarios acerca del modo en que se ha realizado la evaluación. El primero es que la terminología utilizada corresponde a la comúnmente manejada en el entorno de los entrenadores y evaluadores, dentro del campo de aplicación al que va destinado el sistema: la educación física. De esta forma, el sistema presenta de modo independiente cierta autodocumentación para los usuarios a los que va dirigido, y permite pensar y discutir las depuraciones del mismo en términos de conceptos inherentemente asociados al dominio objetivo. El segundo comentario se refiere a que la aparente ingenuidad de las reglas de evaluación puede resultar engañosa, sin embargo gran cantidad de conocimiento y experiencia respaldan estos sencillos modelos. Seleccionar unos valores erróneos para estos modelos de evaluación puede significar un fracaso del programa.

## 4.1 MODULO DE RENDIMIENTO CARDIOVASCULAR

Es indudable que el entrenamiento produce modificaciones en la adaptación del organismo al esfuerzo. Este módulo está orientado a evaluar las características cardiovasculares y respiratorias de una persona. No dependerá ni del sexo ni de la edad de una persona. Según el criterio de los expertos consultados, el perfil cardiovascular de una persona, en cuanto a su capacidad para realizar ejercicio físico, viene

determinado fundamentalmente por tres factores que son: las pulsaciones en reposo, durante y después de un ejercicio físico.

El sistema propone una serie de pruebas para obtener la información necesaria para establecer un criterio acerca del rendimiento cardiovascular de una persona. La relación adecuada entre las mediciones del pulso realizadas antes, después y durante las pruebas, determinará mediante reglas fuzzy proporcionadas por el experto evaluador/seleccionador, una medida cualitativa del rendimiento cardiovascular de la persona (excelente, bueno, mediocre, pobre o malo).

### 4.2 MODULO DE GRADO DE ENTRENAMIENTO

El modulo de evaluación del grado de entrenamiento combina tanto factores musculares como orgánicos. Al igual que el módulo de evaluación anterior, el sistema propone una serie de ejercicios que le permitirán obtener los datos necesarios para establecer un criterio acerca de la capacidad de entrenamiento (Resistencia Anaeróbica) de una persona.

Los resultados obtenidos en los ejercicios por una persona darán un criterio acerca de su grado de entrenamiento dependiendo de la edad y del sexo de la persona. El grado de entrenamiento podrá evaluarse cualitativamente de modo fuzzy (malo, pobre, medio, bueno, excelente.)

### 4.3 MODULO DE EFICIENCIA MUSCULAR

La eficiencia muscular es la tercera característica de gran importancia para determinar el rendimiento de un individuo en actividades físicas. Normalmente los expertos entrenadores hablan en términos cualitativos acerca de esta característica: "*Tiene una eficiencia muscular excelente para atletismo*".

En este módulo se pronostica el rendimiento de un individuo en un determinado deporte. Con este objetivo, el sistema realiza una evaluación fuzzy de la eficiencia muscular a partir de un conjunto de ejercicios propuestos por los expertos entrenadores para obtener la información necesaria. Según los conocimientos y experiencia de los entrenadores consultados, la eficiencia muscular depende esencialmente de una serie de factores musculares, y reflexionando acerca de su experiencia y conocimientos, finalmente manifestaron que se podría establecer una asociación entre estos aspectos y los resultados que se obtengan en distintos ejercicios a realizar. A continuación se presenta una lista posible:

| **Factor Muscular** | **Ejercicio** |
|---|---|
| Potencia anaerobica | Carrera 5x10 |
| Flexibilidad | Flexión de tronco |
| Fuerza muscular | Flexión de brazos |
| Potencia muscular | Salto de longitud |
| Fuerza-Resistencia muscular | Abdominales |

En base a esta información se podrá realizar la selección de integrantes para un equipo de competición. La predisposición para la práctica de ciertas modalidades deportivas requiere la consideración de datos y ciertos factores de peso definidos por el seleccionador o evaluador, cuyos valores previsibles se superponen con frecuencia a las aptitudes manifestadas en el momento de la evaluación. Dependiendo del deporte a practicar, la influencia o **peso** de unos factores musculares será más predominante que otros, las pautas a seguir será direccionadas por la experiencia y la base de conocimientos del sistema experto. Los deportes que se han incluido en el sistema experto desarrollado son: piragüismo, atletismo, rugby y mantenimiento.

## 5 DIAGNOSTICO DE PREPARACIÓN FÍSICA Y PLAN DE ENTRENAMIENTO A MEDIDA

La evaluación final corresponde al diagnostico de preparación física en el que se intenta determinar las características importantes que la persona posee al comienzo del proceso de entrenamiento, y de este modo se delimitan los objetivos concretos a alcanzar. Como consecuencia de la evaluación final se podrá:

- Identificar debilidades o dificultades en el nivel físico de las personas de modo individualizado.
- Asegurar que en función de las tres evaluaciones parciales que se han realizado se ubicará a la persona en el nivel adecuado, y se localizará el punto de partida óptimo para que resulte factible alcanzar aquellos objetivos que son prioritarios en su perfil.

Sería conveniente destacar que los tres módulos de evaluación reflejan el punto de vista de los expertos entradores que han marcado las pautas de las especificaciones generales del sistema. Dentro del campo de la educación física habrá personas que confíen en que estas reglas que se han dado son unas buenas directrices de medida de la condición física, pero existirán otros expertos de la materia que consideren que existe otro tipo de parámetros más representativos de la condición física de una persona. Aunque resulte anecdótico dentro de un mismo campo de aplicación, con las características de la educación física, en la que no existen pautas generales objetivas de evaluación, resulta realmente difícil conseguir un consenso entre personas que trabajan el mismo campo.

La tarea de formular unas reglas que permitan diagnosticar la condición física general de la persona, basada en los resultados parciales de los tres módulos independientes de evaluación fuzzy, representa una de las partes más complicadas [1]. La principal dificultad reside en representar el conocimiento del experto que es relevante. De modo más inmediato, existen tres diagnósticos claros y evidentes:

- Personas con una condición física óptima, que se podría denominar *condicion_atletica.*
- Personas con unas características dentro de la media, se podría clasificar como *condicion_deportiva_media.*
- Personas con unas características por debajo de la media y que se clasificaría como *condicion_deportiva_con deficiencias.*

Sin embargo, profundizando un poco más en el análisis de las situaciones reales que se plantean, se identificaron dos casos especiales que por la delicadeza de la situación convendría considerar: el primer caso correspondería a personas cuya condición cardiovascular fuera especialmente deficiente, en cuyo caso sería altamente recomendable realizar una consulta médica antes de iniciar cualquier actividad física con cierta continuidad, *consulta_medica_antes de_incorporación.* El segundo caso estaría asociado a personas con un desarrollo muscular bastante avanzado, pero con problemas aeróbicos, el diagnostico de esta condición se podría denominar como *enfasis_en_ejercicios_aerobicos.*

Si se hace un repaso, en total había tres factores a ser considerados que eran: rendimiento cardiovascular, grado de entrenamiento y eficiencia muscular. Cada factor tenía un valor cualitativo, por ejemplo, el rendimiento cardiovascular podía variar entre excelente, bueno, mediocre, pobre o malo. El sistema a desarrollar claramente se enfrenta con un modelo particular de un problema general: Encontrar un resultado a partir de una escala ordinal basada sobre los resultados cualitativos de diversas escalas ordinales borrosas (rendimiento cardiovascular, grado de entrenamiento y eficiencia muscular).

Existe un problema potencial al utilizar escalas ordinales debido al gran número de casos individuales que se necesitan especificar. Si cada uno de los tres módulos de evaluación tiene m posibles resultados, hay $3^m$ casos que se tienen que considerar. En general no es viable tener una regla separada para cada posibilidad. El tener tantas reglas no origina únicamente un problema de espacio, sino que la búsqueda involucrada para encontrar la regla adecuada puede ser prohibitivo. Por ello se tomó la decisión de trasladar únicamente al sistema los casos más comunes que se dan en situaciones reales, analizados previamente por los expertos entrenadores.

En el programa se utiliza la estructura *regla(Condiciones,Conclusión),* dónde *Condiciones* es una lista de condiciones bajo las cuales se aplica la regla objetivo y *Conclusión* es la conclusión de la regla. Una condición tiene la forma *condicion(Factor,Relacion,Evaluacion),* incidiendo en que la evaluación resultante del factor denominado *Factor* soporta la relación identificada por *Relación* con respecto a la evaluación dada por *Evaluación.*

Por ejemplo la regla que concluye la condición atlética vendría dada por:

*si cardio_vascular es excelente y entrenamiento es bueno y muscular es excelente* **entonces** *condicion_atletica.*

Esta regla se lee: "Si el rendimiento cardiovascular de una persona es excelente, el grado de entrenamiento es como mínimo bueno, y la eficiencia muscular es excelente, la persona en cuestión tiene una condición física atlética". El interprete para las reglas está formulado de modo no determinista.

A partir de la ubicación de la persona en el grupo adecuado, como consecuencia del diagnostico final, el módulo Plan de Entrenamiento a Medida define el ritmo del proceso de entrenamiento adecuado a las características de la persona. Este módulo en realidad lo que hace es presentar el *PlanEntrenamiento* correspondiente a un *DiagnosticoFinal,* para lo que se ha definido un plan de entrenamiento a medida asociado a cada *DiagnosticoFinal* posible.

# 6 CONCLUSIONES

El sistema experto prototipo desarrollado es una composición de métodos y estilos. Reglas heurísticas de pensamiento, junto con algoritmos simples se utilizan en los módulos de evaluación. Existe un lenguaje de reglas, con un interprete para expresar resultados en términos de valores a partir de escalas ordinales borrosas.

La principal fuente de información y experiencia que se ha utilizado para especificar este programa procede de un grupo de expertos entrenadores del club deportivo CANOE. Se ha superado la dificultad de extracción explícita del conocimiento y consenso entre expertos, tanto en la definición de valores lingüísticos como en la especificación de reglas, mediante un arduo esfuerzo en reuniones de puesta en común y utilizando estrategias basadas en ejemplos en la especificación de puntos concretos.

Actualmente "El Sistema Fuzzy para Evaluación y Entrenamiento Físico" se encuentra en una fase piloto de evaluación de la bondad de los resultados que se obtienen. Una vez superada dicha fase, se realizará una segunda version con los refinamientos de las retroalimentaciones obtenidas y mejoras propuestas por los grupos de preparados físicos que están trabajando con él.

A través del presente trabajo se ha podido valorar muy de cerca las ventajas que presenta un lenguaje de programación lógica, tipo declarativo. Entre estas cabría mencionar la facilidad que ofrece en la **especificación** de un sistema. La programación particularmente a alto nivel permite clarificar las ideas y conceptos de un sistema, de modo que una implementación de este tipo no se reduce a una herramienta de codificación, sino que abarca el ámbito de la especificación.

**Agradecimientos**

Se agradece especialmente la colaboración prestada por el grupo de preparadores físicos del Real CANOE Natación Club de Madrid. Especialmente a la Sección de Rugby cuya labor fue coordinada por Luis Javier Portillo Yábar.

**Referencias**

[1] D. Dubois, H. Prade, L. Ughetto (1996). "Coherence of Fuzzy Knowledge Bases", Proc. of Fifth IEEE Int. Conf. on Fuzzy Systems, vol III, pp. 1858-1864.

[2] G. Guida, and C. Tasso (1995). "Design and Development of Knowledge-Based Systems", Ed. Wiley&Sons.

[3] M.M. Gupta, and T. Yamakawa (1988). "Fuzzy logic in knowledge-based systems, decision and control", Ed. M.M. Gupta and T. Yamakawa.

[4] M.M.Gupta, and N.K. Sinha (1995). "Intelligent control systems: concepts and applications", Ed. IEEE Press.

[5] N. Vadiee, T. Akbarzadech (1996). "Analogous Fuzzy Rule-Based Expert Systems", Proc. of Fifth IEEE Int. Conf. on Fuzzy Systems, vol III, pp. 1852-1857.

# Múltiple Clasificación en Nivel de Medida aplicada al Reconocimiento de Textos

**Aitziber Sanz[1], Javier Echanobe[1],José R. González de Mendívil[2], José R. Garitagoitia[3]**

(1) Departamento de Electricidad y Electrónica, Universidad del País Vasco,
P.O. Box 644, 48080 BILBAO, SPAIN; e.mail:aitziber@we.lc.ehu.es
(2)Departamento de Automática y Computación, Universidad Pública de Navarra,
Campus Arrosadia s/n, 31006 PAMPLONA, SPAIN
(3) Departamento de Matemáticas e Informática, Universidad Pública de Navarra,
Campus Arrosadia s/n, 31006 PAMPLONA, SPAIN

## Resumen

Las técnicas de Múltiple Clasificación han demostrado su capacidad de mejorar el rendimiento de los Clasificadores de Caracteres Aislados (ICC) mediante la combinación de sus salidas. Dicha combinación se ha efectuado mayoritariamente a nivel abstracto o de rango, no confiriendole tanta importancia al desarrollo de métodos de Múltiple Clasificación en el nivel de medida.
En este artículo se presenta una modificación de técnicas de Múltiple Clasificación capaces de aceptar información expresada en el nivel de medida proveniente de Clasificadores de Caracteres Aislados. El resultado de esta combinación está expresado también en el nivel de medida dado que se impone el uso de los Sistemas Deformados. Un Sistema Deformado constituye una herramienta fuzzy que, incorporando Procesamiento Contextual, es capaz de trabajar con este tipo de información.
Para evaluar el rendimiento de las técnicas de Múltiple Clasificación adaptadas, se ha desarrollado un experimento de reconocimiento de textos manuscritos. Las tasas de reconocimiento obtenidas muestran que esta técnicas obtienen una notable mejoría respecto a cada uno de los sistemas de reconocimiento de textos individuales.

**Palabras Clave:** Reconocimiento de Patrones, Múltiple Clasificación, Sistemas Deformados, Procesamiento Contextual, Carácter Fuzzy.

## 1 INTRODUCCIÓN

El concepto de combinación de Múltiples Clasificadores (MC) ha supuesto un nuevo enfoque en el desarrollo de sistemas de Reconocimiento de Textos altamente fiables, habiéndose obtenido resultados que indican que la combinación de varios clasificadores mejora el rendimiento obtenido por cada clasificador aislado [1]-[3]. Básicamente, un MC consiste en un conjunto de clasificadores aislados (ICC) y una función de decisión combinada [2]. Cada ICC emplea un descriptor particular del carácter de entrada y efectúa decisiones sobre la pertenencia de dicho carácter a un conjunto dado de clases. Se combinan todas las decisiones en una decisión final, constituyendo ésta la salida del sistema.

La información obtenida a la salida del ICC puede venir dada en tres niveles [1]: (i)Nivel abstracto: si sólo se obtiene información relativa a la clase más representativa a la que pertenece el carácter; (ii) Nivel de rango: se obtiene una lista ordenada de las clases de caracteres, donde la primera clase de la lista es la más representativa, y así sucesivamente y, (iii) Nivel de medida: el clasificador asigna a cada una de las clases un valor que indica el grado de proximidad del carácter a esa clase.

Por otra parte, los sistemas de reconocimiento que incluyen técnicas de Procesamiento Contextual (PC) [4] han demostrado ser capaces de recuperar gran parte de los errores producidos por los ICC. En experiencias anteriores [5]-[7] se ha comprobado que los Sistemas Deformados (SD) son una herramienta fuzzy que, aceptando como entrada los resultados de un ICC expresados en forma de carácter fuzzy en el nivel de medida, e incluyendo el Contexto Lexicográfico en términos de un diccionario, producen excelentes resultados de rendimiento.

Hasta la fecha, se ha concedido la mayor importancia al desarrollo de métodos de MC que combinan clasificadores en el nivel abstracto o de rango, no realizándose tantos experimentos en el nivel de medida [8]. Además, estas combinaciones se han efectuado en el nivel final de toma de decisiones, es decir, una vez que ya se ha efectuado el reconocimiento. En este artículo, se presenta una adaptación de una serie de técnicas de MC a experiencias en las que se trabaja en el nivel de medida. El MC implementado acepta como entradas los caracteres fuzzy expresados en el nivel de medida provenientes de varios ICC, y genera como salida un carácter fuzzy,

también en el nivel de medida. Este carácter fuzzy generado por MC sirve de entrada al SD, quien es el que finalmente efectúa el reconocimiento. Para probar la efectividad las modificaciones de estas técnicas se proponen una serie de experimentos en los que se observa una mejoría en las tasas de recuperación de errores respecto a las propias del SD.

Este artículo está organizado en las siguientes secciones: En la segunda sección se desarrolla el Sistema de MC en el nivel de medida.. En la tercera sección se describen los experimentos y pruebas realizados. En el último apartado se exponen las conclusiones y las nuevas perspectivas.

# 2 MÚLTIPLE CLASIFICACIÓN EN EL NIVEL DE MEDIDA

En esta sección se presenta una formulación de un Sistema de Reconocimiento de Textos en el que se aplican técnicas de Múltiple Clasificación en el nivel de medida. En la Figura 1 puede apreciarse una representación del sistema. Dicho sistema está constituido por varios Clasificadores de Caracteres Aislados (ICCs), un Sistema Deformado y un Múltiple Clasificador en el nivel de medida.

## 2.1 CLASIFICADOR DE CARACTERES AISLADOS

En este caso se ha construido como ICC una red neuronal basada en el modelo del Perceptrón Multicapa [9]. con una capa de entrada y una capa de salida. La capa de entrada posee 64 unidades o neuronas, que se corresponden con las componentes del vector que representa a un carácter. La capa intermedia también cuenta con 64 neuronas. Finalmente, la capa de salida tiene 26 unidades que se ajustan de acuerdo con los 26 caracteres que contiene nuestro alfabeto.

El aprendizaje del ICC se realiza con un alfabeto patrón definido previamente. Para cada carácter patrón se elige una unidad de la capa de salida que tomará el valor uno, mientras que el resto de las unidades tomarán el valor cero. De esta forma cada unidad de la capa de salida se identifica con un carácter patrón.

En el proceso de clasificación, cuando un vector correspondiente a la imagen de un carácter es presentado a la entrada de la red neuronal, el carácter reconocido es aquel que se corresponde con el patrón cuya unidad asociada de la capa de salida tiene el valor más alto. Este valor puede interpretarse como la "proximidad" de la imagen al carácter patrón (1 representa un total acuerdo, 0 representa un total desacuerdo). Los valores de todas las unidades de salida representan, de la misma manera, la "proximidad" de la imagen a cada uno de los patrones asociados a dichas unidades.

Si se considera esta salida en el nivel de medida, puede construirse [7] un **carácter fuzzy** $\tilde{y}$: conjunto fuzzy asociado al carácter de entrada x sobre el alfabeto $\Sigma$;

$$\tilde{y} = \left\{(\mu_y(\alpha_i), \alpha_i), \alpha_i \in \Sigma, \mu_y(\alpha) \in [0,1]\right\} \quad (1)$$

## 2.2. PROCESAMIENTO CONTEXTUAL MEDIANTE SISTEMAS DEFORMADOS

Un Sistema Deformado [10] es capaz de tratar con caracteres fuzzy junto con el contexto lexicográfico . Este sistema se define como un autómata cuyo comportamiento está regido de forma "fuzzy" por entradas fuzzy. En este caso, el autómata representa un diccionario y las entradas fuzzy son caracteres fuzzy [7].

Una forma de implementar un diccionario es mediante un Autómata de Estados [11]. Como el diccionario D es finito. el autómata propuesto es finito y determinista. Sea $A \neq (\Sigma, Q, q_0, \delta, \lambda, \Delta)$ un Autómata de Estados finito y determinista de tipo Moore que acepta el diccionario D; los elementos del autómata se definen como:(i) $\Sigma$ es el alfabeto; (ii) Q es el conjunto de estados; (iii) $q_0 \in Q$ es el estado inicial; (iv) $\delta$ es la función de transición definida como $\delta : Q \times \Sigma \times Q \rightarrow \{0, 1\}$, siendo $\delta(q, \alpha, q') = 1$ si existe la transición de q a q' mediante el carácter $\alpha$, y $\delta(q, \alpha, q') = 0$ si no existe tal transición; (iv) $\Delta$ el alfabeto de salida, donde $\Delta = D \cup \{\varepsilon\}$ ($\varepsilon$ denota la cadena vacía); (v) $\lambda$ la función de salida definida como $\lambda: Q \rightarrow \Delta$. Dado un diccionario específico D, los elementos arriba presentados pueden ser calculados de la siguiente forma: Sea Q un conjunto finito de estados. Inicialmente $\forall q, q' \in Q$: $\delta(q, \alpha, q') = 0$ y $\lambda(q) = \varepsilon$. Dado un diccionario de palabras $\omega = \alpha_1..\alpha_t..\alpha_m$, una secuencia de estados $q_0..q_t..q_m$ puede ser definida por construcción, siendo $q_0$ el estado inicial $q_t \in Q$ con $1 \leq t \leq m$, y $\delta(.,.,.)$ se actualiza mediante $\delta(q_{t-1}, \alpha_t, q_t) = 1$ con $1 \leq t \leq m$. Finalmente, $\lambda(q_m) = \omega$. Este método se restringe de forma que el autómata obtenido sea determinista.

Cuando las entradas están constituidas por secuencias de caracteres fuzzy, es posible utilizar el autómata definido anteriormente utilizando una interpretación fuzzy. El Sistema deformado para el autómata A se define como la tupla $Ad = ((\Sigma, \tilde{y}), (Q,S), q_0, \delta, \lambda, (\Delta,W))$ donde $\Sigma$, Q y $\Delta$ son restringidos de manera fuzzy por los conjuntos fuzzy , $\tilde{y}$ S and W, respectivamente; $\delta$ es la función de transición cuyo dominio es restringido de forma fuzzy por S x $\tilde{y}$ x S; $\lambda$ es la función de salida cuyo dominio es restringido de manera fuzzy por S y su rango por W. Considérense los conjuntos genéricos fuzzy $\tilde{y}$, S y W definidos como $\tilde{y} = \{(\mu_{\tilde{y}}(\alpha), \alpha), \alpha \in \Sigma\}$; $S = \{(\mu_S(q), q), q \in Q\}$; $W = \{(\mu_W(\omega), \omega), \omega \in \Delta\}$, de universos $\Sigma$, Q y $\Delta$ , respectivamente. Las ecuaciones de estado del Sistema Deformado son las siguientes: Sea $(S_t, \tilde{y}_t, S_{t+1})$ el t-

ésimo paso del sistema (una vez que las ecuaciones el término $\mu_{S_t}(q_t)$ es simplificado por $\mu_S(q_t)$);

(a) Dado $\delta(q_t, \alpha_t, q_{t+1})$ entonces $\mu_S(q_{t+1}) \geq \min(\mu_S(q_t), \mu_{\tilde{y}}(\alpha_t), \delta(q_t, \alpha_t, q_{t+1}))$.

(b) Dado $\omega = \lambda(q_t)$ entonces $\mu_W(\omega) \geq \mu_S(q_t)$.

El cálculo del conjunto fuzzy W, que constituye la salida del Sistema Deformado, se efectúa simultáneamente a partir de las ecuaciones (a) y (b). Por consiguiente, una vez que se alcanza el final de la secuencia de caracteres fuzzy para una palabra dada de entrada, la palabra observada a la salida del Sistema Deformado será aquella palabra del diccionario que presente el máximo valor de función de pertenencia en el conjunto fuzzy W: $Z \in D$ / $\forall \omega \in D: \mu_W(Z) \ \mu_W(\omega)$.

### 2.3 Adaptación de Técnicas de MC con entrada y salida en nivel de medida

Se han adaptado técnicas de MC que no requieren independencia estadística de clasificadores: principio de mayoría de votos [1], método BKS [12], estimación de Borda [13] y rango más alto [13], técnicas que a su vez no han sido aplicadas anteriormente al tratamiento de información en el nivel de medida. Como salida de estos métodos modificados se obtiene un nuevo carácter fuzzy que servirá de entrada a la etapa de procesamiento contextual via SD.

**1) Mayoría de Votos (MV)** Según esta técnica de MC, el carácter resultante es aquél que recibe el mayor número de votos. En esta experiencia se considera además el caso de empate, primando la decisión del clasificador más potente, es decir, aquél que produce mejor tasa de reconocimiento. Para generar el carácter fuzzy, se elige entre los caracteres obtenidos a la salida de las redes que coincidan con el más votado, aquel carácter que presente el mayor valor de función de pertenencia para la letra reconocida. Para representar los caracteres rechazados se utiliza el carácter fuzzy comodín, que es aquél para el cual el valor de la función de pertenencia de todas sus componentes es 1.

**2) Método BKS (BKS)** El Espacio de Conocimiento del Comportamiento (del inglés Behaviour Knowledge Space) es un espacio k-dimensional donde en cada dimensión (unidad focal) se almacena la intersección de las decisiones de k clasificadores acerca de un mismo carácter de entrada. La MC para un carácter se efectúa eligiendo en la correspondiente unidad focal el carácter prototipo de la clase más representativa. El carácter fuzzy se construye de la misma forma que en el método anterior, se elige, de entre las salidas de los ICC que coinciden con el carácter modelo, aquél que presenta el mayor valor de función de pertenencia para la letra reconocida.

**3) Método de la Estimación de Borda (CB).** La estimación de Borda para un carácter es el número de caracteres que se encuentran por debajo de él cuando los ICC producen su salida en el nivel de rango. Se procede al cálculo de dicha estimación para cada carácter según las salidas dadas por los ICC. Con los valores obtenidos, se elabora un vector donde la componente relativa al carácter es la suma de sus estimaciones de Borda correspondientes. Con el vector así obtenido, se procede a la construcción del caracter fuzzy mediante una normalización del mismo y una reordenación en el nivel de medida.

**4) Método del Rango más Alto (RA.)** A partir de la información obtenida de los ICC, se construye un vector en el que en cada componente se almacena el rango más alto que ha alcanzado cada carácter del alfabeto. Seguidamente, se procede a una normalización del mismo, construyéndose el caracter fuzzy que constituye la salida en el nivel de medida de este MC.

## 3. Pruebas y Resultados

Las pruebas experimentales se han realizado sobre un conjunto de cincuenta textos manuscritos escritos en lengua inglesa, que han sido escritos por cincuenta autores diferentes. Se ha impuesto una única restricción al tipo de letra : se trata de letras mayúsculas manuscritas no solapadas.

Los ICC han sido entrenados mediante cinco cuerpos de aprendizaje diferentes, para así obtener distintos clasificadores $ICC_1,...,ICC_5$.

El rendimiento del sistema se mide en términos del porcentaje de palabras del texto bien reconocidas a la salida. La **Tasa de Reconocimiento** corresponde a la fórmula

$$\%t_{rec} = \frac{\text{número de palabras reconocidas}}{\text{número total de palabras}} \text{x}100 \quad (2)$$

Además, la **Tasa de Reducción del Error** mide la mejora en la capacidad de reconocimiento introducida por la incorporación del Contexto Lexicográfico. Se expresa comparando el sistema con y sin Procesamiento Contextual mediante el Sistema Deformado. Por lo tanto, puede definirse esta tasa como:

$$\%t_{red} = \left(1 - \frac{\%t_{err}(ICC + DS)}{\%t_{err}ICC}\right)\text{x}100 \quad \text{con}$$

$$\%t_{err} = 100 - \%t_{rec} \quad (3)$$

Además, la tasa de Reducción del Error mediante MC se define como la mejora en la capacidad de reconocimiento producida por la aplicación de las técnicas de MC adaptadas, con relación a cada ICC.

$$\%t_{MCred} = \left(1 - \frac{\%t_{err}(ICC + MC + DS)}{\%t_{err}ICC}\right)x100 \quad (4)$$

En las tablas 1 2 y 3 pueden apreciarse los resultados obtenidos, y cómo se producen mejoras, tanto en el proceso de reconocimiento como en la reducción del error cometido por los ICC.

En la Tabla 1 se presentan las tasas de reconocimiento de los clasificadores individuales $ICC_i$ y cada uno de ellos junto con el SD. Además, se muestra la tasa de reducción del error producida por la incorporación del segundo. Ha de notarse que las tasas de reconocimiento de los $ICC_i$ son tan pequeñas debido a que están referidas a palabras completas y no a caracteres, y por lo tanto no se ha introducido ningún tipo de contexto.

En la Tabla 2 se muestran las tasas de reconocimiento de las técnicas adaptadas de MC al nivel de medida. Puede verse que con estas técnicas se obtienen mejores resultados que con los sistemas por separado. En la Tabla 3 se presentan las tasas de reducción del error de las MC. En todos los casos, la tasa de reducción del error es más alta que en los sistemas sin MC.

## 4 Conclusiones

En este artículo se ha presentado una modificación de técnicas de Múltiple Clasificación con objeto de generar, a partir de una combinación de salidas de clasificadoresen el nivel de medida, información expresada también en el nivel de medida. Esta aplicación no es producida en el nivel de toma de decisiones, es decir, una vez que ya se ha efectuado el reconocimiento final, sino que se efectúa en un nivel intermedio justo antes del procesamiento contextual.

Mediante la adaptación de estas técnicas de MC se procede a la construcción de caracteres fuzzy, los cuales sirven de entrada a la etapa de procesamiento contextual por los Sistemas Deformados. De esta forma se consigue incrementar las tasas de reconocimiento y la tasa de reducción del error.

Este trabajo permite deducir, por otra parte, que pueden desarrollarse nuevos métodos de MC con salida en el nivel de medida y, dado que uno de los objetivos es la construcción de caracteres fuzzy que sirvan como entrada a una etapa posterior de PC, en experiencias posteriores se puede utilizar, como funciones de composición, operaciones de carácter fuzzy tipo max-pros, max-min, media global o media global cuadrática.

**Agradecimientos**

Los autores agradecen al Gobierno Vasco por la financiación económica del presente trabajo.

**Referencias**

[1] L. Xu, A. Krzyzak, C.Y. Suen, 'MethoSD of Combining Multiple Classifiers and Their Applications to Handwritting Recognition', *IEEE Trans. Systems, Man and Cybernetics*, Vol. 22, n° 3, pp. 418-435, May/June 1992.

[2] T.K. Ho, J.J. Hull, S.N. Srihari, 'Decision Combination in Multiple Classifier Systems', *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 16, n° 1, pp. 66-75, 1994.

[3] S. Madhvanath, V. Govindaraju, 'Serial Classifier Combination for Handwritten Word Recognition', Proc. Third Int. Conf on Doc. Anal. and Rec. ICDAR95, Montreal, Canada, Aug. 14-16, 1995.

[4] J.J. Hull, S.N. Srihari, R. Choudhari, 'An Integrated Algorithm for Text Recognition: Comparison with a Cascaded Algorithm', *IEEE Trans. Pattern Anal. and Machine Intell.* Vol.5, pp.384-395, 1983.

[5] J. Echanobe, R. Reina, J.R.Garitagoitia, J.R. González de Mendívil, 'Deformed Systems in Text Recognition', Int. Conf. on Artificial Neural Networks, Sorrento, Italy, May 26-29, 1994.

[6] J. Villadangos, A. Sanz, J.R. González de Mendívil, F. Fariña, "Recuperación de Errores de segmentación y de Clasificación utilizando Sistemas Deformados', V Congreso Español sobre Tecnologías y Lógica Fuzzy, 20-22 Sep. 1995, Murcia, España.

[7] J. Echanobe, J. R. González de Mendívil, J.R. Garitagoitia, C.F. Alastruey, 'Deformed Systems for Contextual Postprocessing', *Fuzzy Sets and Systems,* in Press.

[8] Y.S. Huang, K. Liu, C.Y. Suen, 'The Combination of Multiple Classifiers by a Neural Network Approach', *Int Jour. of Pattern Rec. and Art. Intell.* Vol.9, n°3, pp.579-597, 1995.

[9] B. Widrow, M.A. Lehr, '30 years of Adaptive Neural Networks: Perceptron, Madaline and Back-propagation', *Proc. IEEE*, 78, pp. 1415-1442, 1990.

[10] C.V. Negoitia, D.A. Ralescu, 'Application of Fuzzy Sets to System Analysis', Birkaueser, 1975.

[11] J. Hopcroft, J. Ullman, 'Introduction to Automata Theory, Languages and Computation', Addison-Publishing Company, Reading Massachussets, 1979.

[12] Y.S. Huang, C.Y. Suen, 'An Optimal Method of Combining Multiple Classifiers For Unconstrained Handwritten Numerals Recognition', Proc. III Int. Work. on Frontiers in Handwriting Recognition IWFHR93, Buffalo, N.Y. May 25-27, 1993.

[13] T.K. Ho. 'A Theory of Multiple Classifiers Systems and its Aplication to Visual Word Recognition', Doctoral Dissertation, Dept. of Computer Science, State Univ. of N.Y.May,1992.

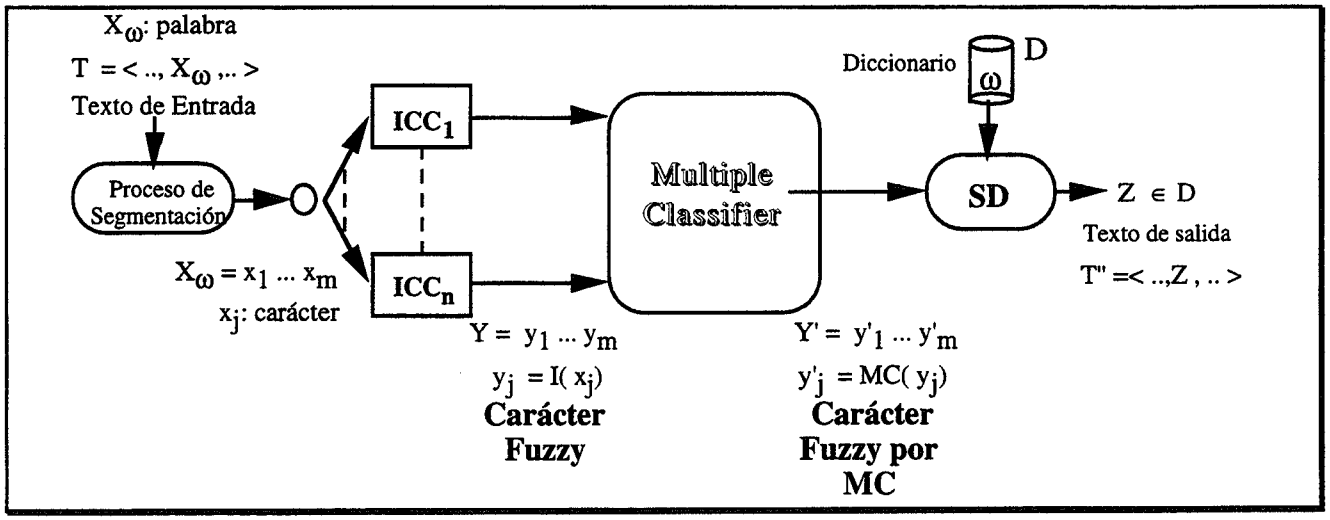



**Figura 1.** MC en el nivel de medida.

Tabla 1. Tasas de Rec y Red. Error

| | | | | | |
|---|---|---|---|---|---|
| $\%t_{rec}$ $ICC_i$ | 17.582 | 13.503 | 14.645 | 13.178 | 11.631 |
| $\%t_{rec}$ $ICC_i$+SD | 87.351 | 88.464 | 86.491 | 86.685 | 79.456 |
| $\%t_{red}$ | 85.164 | 87.269 | 84.905 | 85.265 | 77.648 |

Tabla2 Tasas de Rec. de MC con SD

| MVSD | BKSSD | CBSD | HRSD |
|---|---|---|---|
| 89.301 | 89.524 | 91.730 | 92.483 |

Tabla 3 Reducción error de los MC respecto a ICC

| $\%t_{MCred}$ | MVSD | BKSSD | CBSD | HRSD |
|---|---|---|---|---|
| $ICC_1$ | 87.582 | 87.803 | 90.451 | 91.326 |
| $ICC_2$ | 88.209 | 88.359 | 90.877 | 91.785 |
| $ICC_3$ | 88.060 | 88.277 | 90.847 | 91.692 |
| $ICC_4$ | 88.291 | 88.468 | 91.003 | 91.855 |
| $ICC_5$ | 88.336 | 88.538 | 91.033 | 91.911 |

**Notacion:** **$ICC_i$** denota al i-ésimo ICC
**$ICC_i$+SD** denota al i-ésimo ICC con Procesamiento Contextual con Sistema Deformado
**MVSD** MC ppor Mayoría de Votos+ Sistema Deformado
**BKSSD** MC por método BKS + Sistema Deformado
**CB** MC por estimaci'ón de Borda + Sistema Deformado
**HR** MC por Rango más Alto + Sistema Deformado

# Cooperación Mediante Comunicación de Conceptos Borrosos

**Vicente Matellán**
e-mail: vmo@ia.uc3m.es

**José M. Molina**
e-mail: molina@ia.uc3m.es

**Lorenzo Sommaruga**
e-mail: los@ia.uc3m.es

Laboratorio de Agentes Inteligentes (LAI)
Departamento de Informática
Universidad Carlos III de Madrid

## Resumen

Este artículo presenta un ejemplo de cooperación borrosa entre un grupo de robots como una solución alternativa a los protocolos clásicos utilizados en las arquitecturas de cooperación entre robots. Para ello se ha procedido a integrar trabajos previos realizados en el Laboratorio de Agentes Inteligentes (LAI) sobre comportamientos borrosos y sobre arquitecturas de cooperación. Se han considerado tres tipos diferentes de protocolos borrosos y se han probado tanto en un simulador como sobre un robot real. Por último, se presenta la descripción de los experimentos realizados así como los resultados obtenidos.

**Palabras Clave:** Robots autónomos, comunicación, cooperación, coordinación.

## 1 Introducción

Dentro del campo de la cooperación entre robots, nuestra aproximación fundamentalmente aborda el problema de la coordinación desde dos puntos de vista complementarios: la comunicación y el comportamiento. La coordinación de robots puede conseguirse mediante la comunicación explícita entre los individuos, habitualmente realizada mediante mensajes. En segundo lugar, la coordinación global de un grupo puede verse también influenciada implicitamente por los efectos del comportamiento individual de cada robot, lo cual se manifiesta mediante sus acciones y los cambios que éstas producen en el mundo percibido por los otros robots. Estos métodos directos e indirectos de coordinación son mecanismos complementarios muy útiles para obtener cooperación intencional de alto nivel. Además, dichos mecanismos pueden integrarse en un modelo único.

Este artículo presenta un protocolo borroso para la coordinación de robots como una solución alternativa a los protocolos utilizados en otras arquitecturas, tanto fuzzy [5], como tradicionales [4], que se pueden encontrar en la literatura como [6], etc.. Esta clase de protocolos de coordinación proporcionan también un comportamiento suave y flexible gracias a la representación borrosa del conocimiento y al razonamiento borroso. En realidad, las inexactitudes e incertidumbres de los datos proporcionados por los sensores y por la ejecución de las tareas del robot, a'si como los errores de hardware, los errores en las comunicaciones, etc., se tratan fácilmente con una aproximación borrosa. En concreto, la adopción de una filosofía borrosa se puede utilizar para borrosificar tanto el receptor de un mensaje como su contenido, lo que proporciona una comunicación más suave. Más aún, puede utilzarse para realizar una fusión de los comportamientos de los robots, no solo al bajo nivel de las tareas reactivas, como el control de los motores, sino en el alto nivel de la cooperación intencional.

La integración de trabajos previos [2] sobre comportamientos borrosos y arquitecturas de cooperación [7], se han considerado la base para conseguir una arquitectura para la cooperación de robots. La idea principal es crear dos niveles en la arquitectura: un nivel reactivo, que se encarga de todas las actividades *instintivas* de un robot (ej. moverse, girar, evitar obstáculo, etc.); y un nivel de coordinación, que controla las actividades inteligentes, como el comportamiento cooperativo y que influye en el bajo nivel.

En el resto de este artículo se presenta un análisis de la comunicación entre robots, desde la perspectiva de los principios de la lógica borrosa. Así, se describe un experimento que presenta la influencia potencial de la comunicación borrosa en el comportamiento de un robot móvil real.

## 2 Protocolo Fuzzy

Algunos trabajos previos han mostrado que los robots pueden aprovechar las ventajas de las teorías del razonamiento borroso. En este sentido, hemos utilizado [2] controladores borrosos para tratar la incertidumbre que se produce ante las distintas situaciones a las que se enfrenta un robot autónomo. Estos *comportamientos* borrosos se han diseñado como parte de la arquitectura global.

Esta arquitectura se diseñó para coordinar robots en tareas cooperativas. Ésto implica la necesidad de comunicar información entre los robots. Dicha comunicacación se basaba en un protocolo fijo que intercambiaba información clásica. En este capítulo se presenta una nueva aproximación a la construcción de este protocolo, permitiendo la transmisión de información borrosa.

En este trabajo, el protocolo propuesto no será un intercambio cerrado de conceptos *crisp*, idénticamente definidos en todos los robots. Donde el término *crisp* querrá decir que el valor de la variable, o de cualquier otra información intercambiada entre los robots, se define por sus valores numéricos exactos. En nuestro protocolo esto no será asi, lo que nos enfrentará con algunos problemas semánticos.

Expliquemos estos conceptos mediante un ejemplo sencillo. Supongase que un robot, que percibe el mundo mediante sus sensores, obtiene cierta información sobre la anchura de un objeto. Este concepto (anchura) se ha definido en su controlador fuzzy mediante una *variable difusa* llamada *anchura*. El valor de dicha variable se obtiene mediante el correspondiente proceso de borrosificación y se expresa mediante los valores de activación de las variables *lingüísticas* definidas sobre el rango de la variable. Por ejemplo, la variable *anchura* se puede definir por el conjunto de equiquetas {MUY GRANDE, GRANDE, PEQUEÑO y MUY PEQUEÑO}. El conocimiento obtenido sobre lo anchura del objeto se expresará en terminos borrosos como MUY GRANDE (0.8), GRANDE (0.5), PEQUEÑO (0.2) y MUY PEQUEÑO (0), donde 0.8, 0.5, 0.2 y 0 representan el grado de pertenencia de las medidades de los sensores a las variables lingüísticas.

A la hora de definir las etiquetas hay que tener en cuenta el dominio de la variable. Por ejemplo, al definir la variable *anchura* el rango de la varible puede ser por ejemplo de 0 a 1023. En muchos casos este rango viene determinado por los requisitios físicos del sistema. Por ejemplo, el rango de la variable que representa la distancia a un objeto medida por un sensor, viene determinada por el rango del sensor. En el caso de nuestro robot (el micro-robot Khepera [3]) el rango de los sensores de infrarojos es un valor entre 0-1023,

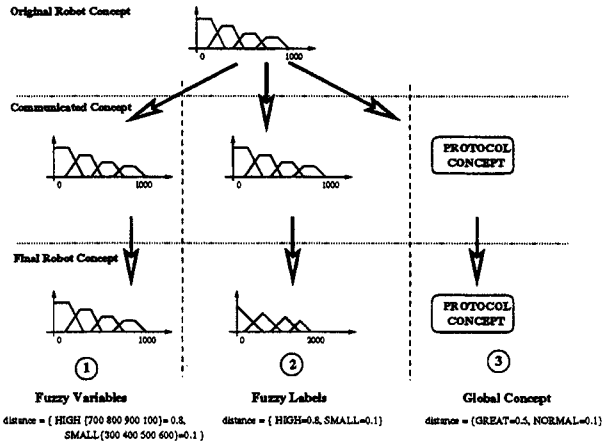



Figura 1: Tres protocolos de comunicación

luego el rango de la variable será 0 - 1023.

Ahora, si nuestro robot quire compartir su conocimiento sobre la anchura del objeto, consideraremos tres posibles métodos de comunicación (según la Figura 1):

1. Comunicación de la variables difusas.
2. Comunicación de las etiquetas lingüísticas.
3. Comunicación de *conceptos del protocolo.*

El primer método se basa en la comunicación de variables difusas almacenadas como un conjunto de valores de las etiquetas lingüísticas. Estas etiquetas habrán sido definidas mediante un conjunto de *funciones de pertenencia* sobre el dominio de la variable. Esto es, el valor de activación de las etiquetas se calcula mediante dichas funciones, que usualmente es una función linear.

Si se envia la variable completa, las etiquetas de todos los robots usarán el mismo dominio y las mismas funciones de pertenencia. En ese caso, el receptor obtiene la misma información borrosa que el emisor. En el ejemplo, querría decir que se ha enviado la definición de las etiquetas { MUY GRANDE, GRANDE, PEQUEÑO y MUY PEQUEÑO }, enviando sus funciones de pertenencia, y los valores de activación de las mismas MUY GRANDE (0.8), GRANDE (0.5), PEQUEÑO (0.2) y MUY PEQUEÑO (0).

El segundo método utiliza etiquetas definidas sobre diferentes rangos de las variables. Esto quiere decir que cada robot usará las mismas etiquetas para definir sus variables difusas (*anchura* en el ejemplo), es decir { MUY GRANDE, GRANDE, PEQUEÑO y MUY PEQUEÑO }, pero el dominio será diferente. Por ejemplo, otro robot utilizará un rango 0 - 2048 en vez del 0 - 1023 del anterior.

Esto quiere decir que el robot receptor usa las etiquetas utilizando sus propias funciones de pertenencia. Esto hace que los robots no compartan la misma información, lo que intercambian es la *impresión* subjetiva. Por ejemplo, la variable "anchura" para un robot pequeño (por ejemplo de 6 cm. de diametro) se puede calificar como MUY GRANDE un objeto de más de 6 cm. (rango de 6 - $\infty$). Si esta información se transmite a un robot más grande, por ejemplo con un diámetro de 30 cm., donde se ha definido el mismo concepto MUY GRANDE como objetos de más de 30cm (rango de 30 - $\infty$), la información realmente intercambiada será diferente.

Para resolver este problema, sería posible ajustar el conocimiento dentro de cada robot. Siguiendo con el ejemplo anterior, si el robot grande sabe que el robot que le envía la información es más pequeño, entonces la información recibida se traduciría desde MUY GRANDE a PEQUEÑO dentro del receptor. De esta forma, lo que se ha transmitido realmente es la impresión del emisor, pero lo que obtiene el receptor es la información real (y la impresión del otro robot, como información adicional). Este mecanismo nos lleva a un tercer método que utilizará conceptos, en forma de etiquetas lingüísticas, compartidos e independientes.

En el tercer método se utilizará un concepto que resulte de la interpretación de la etiqueta lingüística que el robot quiere transmitir. Esto quiere decir que definirán conceptos globales como protocolo de comunicación. Continuando con el ejemplo, los valores de la variable difusa serían traducidos en valores de activación de un nuevo conjunto de etiquetas {ENORME, GRANDE, NORMAL, PEQUEÑO, DIMINUTO}. A continuación, una de estas etiquetas sería la que se transmitiese.

Se puede hacer una analogía con la comunicación humana: una persona tiene una imagen perfecta de una situación que ha vivido, pero cuando comunica esta información a otra utiliza una serie de reglas para traducir sus experiencias en palabras. Esta traducción implica una reducción de la información almacenada en su cerebro. Por su parte, la persona que recibe las palabras transmitidas las traduce en ideas mentales usando sus propias reglas de traducción, obteniendo su imagen particular.

En el ejemplo, la información contenida en la variable difusa se resumirá en el concepto GRANDE, que será la información que se transmitirá. De esta forma todos los robots tienen los mismos conceptos definidos para la comunicación y pueden adaptar esta información limitada de acuerdo a sus características particulares.

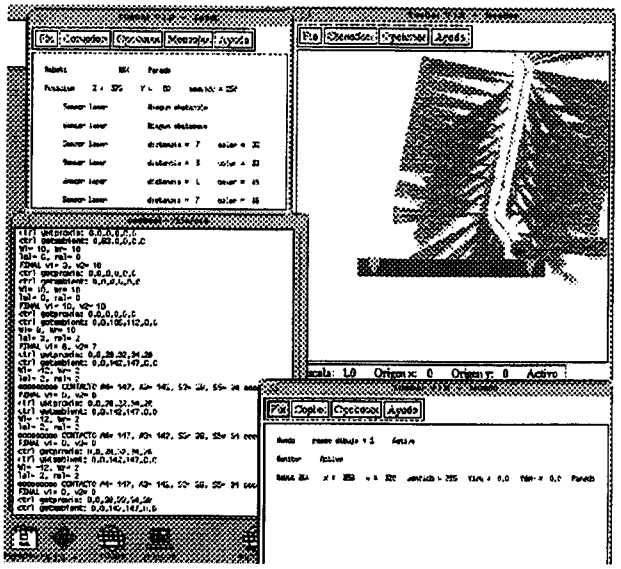


Figura 2: El entorno simulado

## 3 Experimentos

Para probar las diferentes alternativas de comunicación enunciadas en el apartado anterior se han realizado experimentos utilizando tanto un simulador como robots reales. El simulador utilizado ha sido SimDAI [8], la versión distribuida del simulador ERA [7]. SimDAI permite la simulación de un grupo de robots simulados independientes, que se ejecutan en diferentes ordenadores, los cuales desarrollan tareas simples en un mundo definido por el usuario y que es común a los distintos robots. El simulador también proporciona un mecanismo para que los robots se comuniquen.

Los robots reales utilizados han sido dos robots Khepera [3]. Estos robots tienen 5.5 cm de diámetro, dos motores independientes (uno para cada una de sus dos ruedas) y 8 sensores de infrarojos. Dichos sensores pueden medir tanto la distancia a un objeto como la luz ambiente. Estos robots pueden funcionar de forma autónoma (con su propia batería y micro-procesador) o conectados a una estación de trabajo.

Para realizar los experimentos en el simulador SimDAI se ha diseñado un mundo simulado (Figura 2) que imita el entorno real de simulación (Figura 3). Es interesante reseñar que se han utilizado los mismos controladores borrosos para controlar los robots simulados que los robots reales, lo que prueba la idoneidad del simulador SimDAI para este tipo de experimentos.

La siguiente sección describe solamente los experimentos con los robots reales, pues los simulados se han utilizado únicamente para problar la viabilidad de éstos, que son los verdaderamente significativos.

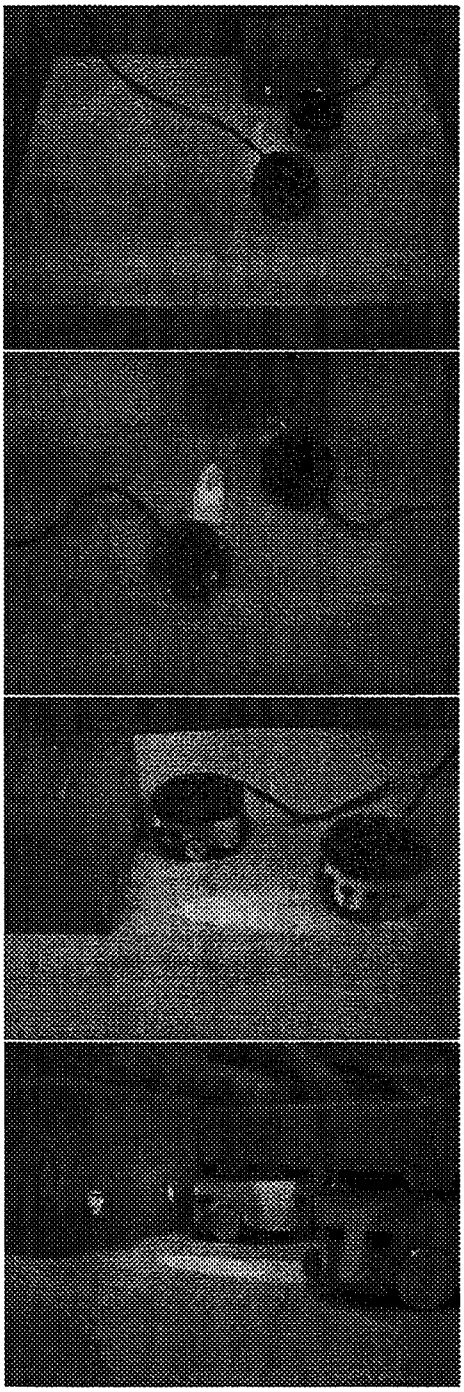


Figura 3: El experimento real

### 3.1 Descripción de los experimentos

Los experimentos realizados intentan mostrar la influencia de la comunicación borrosa en el comportamiento coordinado. Así, el objetivo común de los robots será empujar un objeto, de forma coordinada entre dos robots. Esta tarea puede dividirse en dos subtareas principales, prepararse para empujar y empujar. Para probar las diferentes alternativas de comunicación consideramos suficientemente relevante la primera subtarea. Esta subtarea se ha dividido en tres fases:

1. Un robot encuentra el objeto a empujar.
2. El robot se alinea con el objeto y envía al otro robot una descripción de su alimeamiento.
3. El segundo robot trata de alinearse exactamente como el primero.

La primera fase se puede realizar mediante un controlador difuso sencillo, como alguno de los desarrollados previamente [2], o mediante otro tipo de controladores. Por ejemplo, una versión del controlador reactivo propuesto por V. Braitenberg [1] se ha utilizado para alcanzar la posición correcta de alineación.

Una vez que el robot ha encontrado el objeto, se alinea con él de una determinada forma. Esta fase necesita un control más preciso que la fase anterior, pues el alineamiento determinará si la tarea cooperativa puede ser realizada correctamente o no. Utilizando un robot real como el Khepera, la definición del alineamiento habrá que realizarla en función de las medidas de los sensores que se utilizan en el controlador de alineación.

En este sentido, hemos realizado la definición del alineamiento en función de variables borrosas definidas para cada sensor del robot Khepera. Así, suponemos que un robot, que usa un controlador estándar, se ha alineado cuando sus sensores de proximidad y sus sensores de luz ambiente devuelven unos valores borrosos concretos. En ese momento el robot informará al otro *como* se ha alineado. Será en esta comunicación donde probaremos las tres alternativas de comunicación enumeradas anteriormente.

Además de las tres alternativas borrosas, se ha probado también método de comunicación tradicional, que consiste en el intercambio de conceptos no-borrosos. Es decir, el emisor le dice al otro robot los valores exactos que han medido sus sensores. Esta situación sería la ideal si los dos robots fuesen físicamente idénticos, incluyendo la sensibilidad de los sensores, y si no se produjesen errores en las medidas.

Para mostrar el rendimiento de las diferentes soluciones en primer lugar hemos realizado el experimento

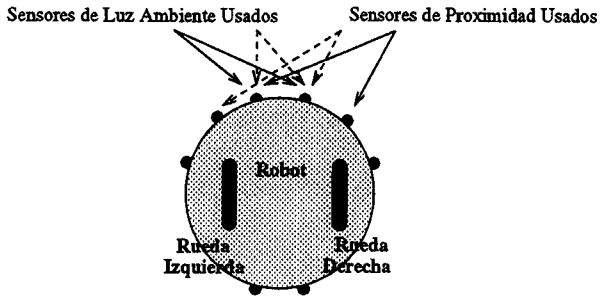



Figura 4: Distribución de los sensores usados en el experimento.

descrito en el simulador SimDAI. En el caso del entorno simulado la subtarea de alineamiento se tiene que realizar en una posición determinada del objeto a empujar. Esta posición se indica en el objeto simulado mediante un color distintinto del propio objeto, como muestra la Figura 2.

A continuación el experimiento se ha realizado sobre el entorno real mostrado en la Figura 3. Para ello se ha construido el objeto, que aparece en la figura, con dos luces en su interior. Estas luces son los puntos en los cuales los robots se tienen que alinear. Cuando uno de los robots se ha alineado envía el entorno percibido por sus sensores al otro robot utilizando uno de los métodos comentados.

En el experimento con los robots reales el entorno se percibe mediante los sensores de luz ambiente y de proximidad. Para simplificar el controlador, solamente se han utilizado los sensores que se muestran en la Figura 4. Dependiendo del lado en el que el robot se tenga que alinear los sensores de proximidad usados son los marcados con líneas de puntos (para alinearse a la izquierda) o los indicados por líneas continuas (alinearse a la derecha).

Dado que los sensores del Khepera son muy sensibles al entorno externo, se realizaron diversas pruebas preliminares para elegir una configuración significativa, ajustando las posiciones iniciales de los robots y la luz ambiente del laboratorio. La configuración final elegida hace que los robots encuentren fácilmente el objeto, pues el objetivo del experimento era probar las comunicaciones y no los controladores de movimiento.

### 3.2 Resultados Experimentales

En el experimento final, se ha medido la distancia al objeto para saber como se han utilizado los valores comunicados en el controlador. En algunas situaciones el tiempo empleado por el segundo robot para alinearse también es una información significativa. Por ejemplo, en el caso utilizar el método tradicional (los va-

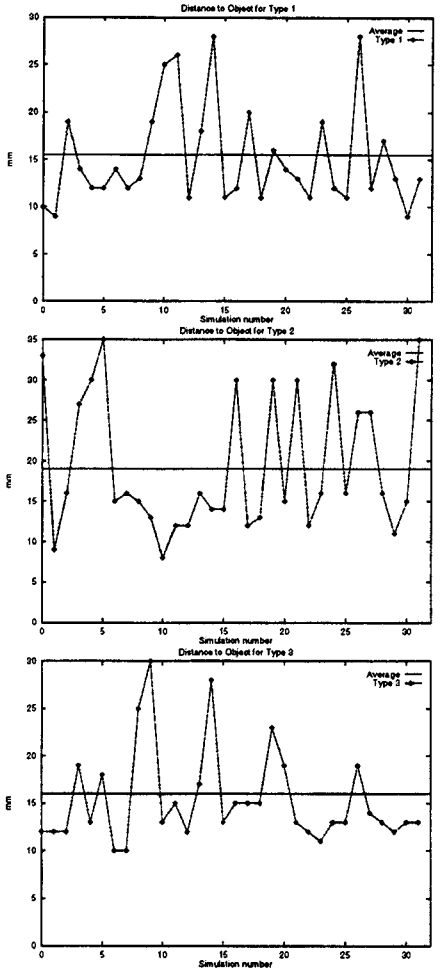



Figura 5: Resultados gráficos de los experimentos

lores numéricos medidos por sus sensores), el segundo robot es incapaz de alinearse en un tiempo razonable. Ello se debe a que en el mundo real hay muy pocas probabilidades de que los sensores del segundo robot midan *exactamente* lo mismo que los correspondientes del primer robot. De hecho, este problema fue el origen de este trabajo.

Sin embargo, para el caso de los mecanismos difusos el tiempo empleado no es significativo para discriminar entre las distintas opciones. En la mayoría de los casos sólo mide la calidad del controlador y su tolerancia a las perturbaciones del entorno (sobre todo a las variaciones de luz ambiente).

Se ha realizado el mismo número de pruebas para cada tipo de comunicación y se han medido los parámetros relevantes. Los resultados obtenidos se muestran en las gráficas de la Figura 5. La distancia media al objeto para los tipos de comunicación uno, dos y tres fueron, respectivamente, 15.125, 19.375 y 15.375. De esta forma se puede apreciar como el primer método es el más exacto, es decir, es el que detiene al robot de forma más similar a como se alineó el primer robot. Este resultado era predecible, pues es el método que pasa más información al controlador.

El segundo es el menos exacto en el alineamiento ya que las etiquetas se definen de forma distinta, sucediendo por ejemplo que una distancia realmente próxima resulte mayor que una comunicada como lejana.

El tercer método tiene un rendimiento similar al primero. Ello es debido a que los dos robots comparten la misma interpretación de un concepto. Este método presenta un buen rendimiento, sobre todo si se tiene en cuenta que es más tolerante al ruido externo. Además, este método es el que utiliza menos recursos de comunicación, puesto que sólo necesita intercambiar un concepto.

El último caso, la comunicación de valores numéricos, produce, como ya hemos comentado, resultados desastrosos, siendo incapaz de alinear al robot, finalizando al consumir el tiempo fijado como máximo.

## 4 Conclusiones y Trabajos Futuros

En este se han presentado tres métodos de comunicación entre robots móviles con el objetivo de realizar una tarea conjunta. La comunicación se ha basado en el intercambio de mensajes borrosos en lugar de los clásicos valores numéricos. Como conclusión se puede decir que los tres métodos borrosos permiten una comunicación entre los robots más flexible, suave y fiable. Además, permiten que los robots incrementen la semántica de los mensajes y obtengan mejores resultados a la hora de alinearse que con el envío de los valores numéricos crudos.

De entre los tres métodos examinados, hemos concluido que el preferible (en condiciones normales) es el tercero, puesto que proporciona un rendimiento aceptable con el menor uso de recursos de comunicaciones. Así mismo, este método es el más tolerante ante perturbaciones externas.

Para contrastar la validez de estos resultados hemos previsto realizar un número similar de experimentos para el desarrollo de tareas más complejas.

## Referencias


[1] **V. Braitenberg** *Vehicles. Experiments in Synthetic Psychology.* MIT Press, Cambridge, MA, 1984.

[2] **V. Matellán, J.M: Molina, J. Sanz, C. Fernández**, *Learning Fuzzy Reactive Behaviors for Autonomous Robots*, 4rd European Workshop on Learning Robots, B-Learn Technology Transfer Workshop. Karlsruhe, Germany, December 1995.

[3] **F. Mondada, F. P. Ienne**, *Mobile Robot Miniaturization: A tool for investigation in control algorithms.* Proceedings of the Third International Symposium on Experimental Robotics. Kyoto, Japan, 1993.

[4] **F. R. Noreils**, *Toward a robot architecture integrating cooperation between mobile robots: Application to indoor Environment* The International Journal of Robotics Research, 12 (1):79-98. 1993.

[5] **P. Reignier**, *Fuzzy logic techniques for mobile robot obstacle avoidance*, Robotics and Autonomous Systems, 12, pag. 143-153, 1994.

[6] **L. E. Parker**, *Heterogeneous Multi-Robot Cooperation.* Ph. Thesis, Massachusetts Institute of Technology, May, 1994.

[7] **L. Sommaruga, N. Patón, E. M. Montero**, *An Architecture for Autonomous Robots Simulation.* Third International Symposium on Intelligent Robotic Systems, Pisa, (Italy). 1995.

[8] **L. Sommaruga, Ignacio Merino, V. Matellán, José Manuel Molina**, *A Distributed Simulator for Intelligent Autonomous Robots.* Fourth International Symposium on Intelligent Robotic Systems, Lisbon, (Portugal). 1996.

# UTILIZACIÓN DE TÉCNICAS DIFUSAS PARA LA DETECCIÓN DE LESIONES PEQUEÑAS DE ESCLEROSIS MÚLTIPLE

**F. Xavier Aymerich**
Unitat de R. M. Vall d'Hebron (I.D.I.)
P.Vall d'Hebron 119-129
08035 Barcelona,Spain
idirmr@ar.vhebron.es

**Pilar Sobrevilla**
Dept. Mat. Aplicada II
U.P.C.
Pau Gargallo 5
08028 Barcelona,Spain
pilars@grec.upc.es

**Jaume Gili**
Unitat de R. M. Vall d'Hebron (I.D.I.)
P.Vall d'Hebron 119-129
08035 Barcelona,Spain
gili.idi@bcn.servicom.es

**Eduard Montseny**
Dept. Eng. Informàtica
U.R.V.
Carretera Salou s/n
43006 Tarragona,Spain
emontsen@etse.urv.es

## Resumen

Este trabajo presenta una aplicación de los algoritmos en los que se utilizan técnicas difusas. El estudio se centra en la automatización del análisis de imágenes obtenidas a través de una técnica no invasiva, como es la resonancia magnética, de pacientes que presentan esclerosis múltiple, y en concreto de las lesiones más pequeñas. La característica incertidumbre que presentan estas lesiones en su definición hace que un enfoque difuso pueda ser una buena solución al problema.

El diseño del algoritmo propuesto se basa en la definición de un conjunto de reglas que permite la extracción de características y posterior análisis de los datos. La fuzzificación de los datos se ha resuelto a partir de la utilización de funciones de densidad de probabilidad. El resultado final es la obtención, a partir reglas, de operadores OWA que permiten un elevado grado de detección de este tipo de lesiones.

El diseño que se propone resuelve el problema de las falsas detecciones que se generan, considerando diferentes filtros implementados a partir de nuevas reglas.

**Palabras clave:** visión por computador, técnicas difusas, resonancia magnética, esclerosis múltiple.

## 1 INTRODUCCIÓN

La esclerosis múltiple es una enfermedad del sistema nervioso central que se caracteriza por la destrucción de la mielina que envuelve los axones. Esta desmielinización comporta la inhibición de la transmisión neural y provoca la aparición de múltiples síntomas clínicos como son la pérdida de la capacidad de coordinación, reducción de la capacidad motriz y alteraciones de la visión entre otros [1,5].

La resonancia magnética (RM) constituye una técnica de diagnóstico clínico que gracias a su excelente capacidad de diferenciación de tejidos blandos se utiliza habitualmente en la confirmación del diagnóstico clínico de la esclerosis múltiple [2,8]. De hecho, en la primera mitad de los años 80, período en el que se inició la aplicación clínica de la RM, la diagnosis de la esclerosis múltiple se convirtió en uno de sus primeros campos de aplicación.

En un estudio craneal por RM, la enfermedad se manifiesta a través de la aparición de alteraciones en la intensidad habitual de las regiones donde se produce esta desmielinización. Esta variación permite observar una serie de placas o lesiones, generalmente de pequeñas dimensiones y con una forma elíptica. Estas lesiones se localizan habitualmente sobre la denominada substancia blanca y predominantemente en la periferia de la región ventricular.

La detección de este tipo de lesiones es importante no únicamente a nivel de confirmar la existencia de esclerosis múltiple en un paciente sinó en el estudio de su evolución. Este punto es especialmente importante en lo referente al estudio de respuesta del individuo a determinados fármacos. Se debe tener en cuenta que en la actualidad no existe ningún fármaco que permita alcanzar la curación del enfermo, por lo que actualmente se realizan numerosos ensayos clínicos para hallar fármacos que permitan luchar contra la enfermedad, y es en este punto donde un estudio evolutivo de un paciente integrado en un ensayo clínico adquiere especial interés. Esto ha propiciado que desde principios de los años 90 se haya orientado gran parte de la investigación en este campo a la segmentación de las lesiones con el objetivo de cuantificar el volumen lesional de un paciente.

La detección de las lesiones es un proceso complejo dado que existe un elevado grado de incertidumbre en su definición que afecta de forma notable a la cuantificación final [2]. Esto ha hecho que en numerosas ocasiones se haya optado por métodos de segmentación semiautomáticos [5,9,10], aunque lo ideal es la automatización del análisis para evitar de este modo el error inter-observador y poder comparar mejor los resultados [2,4]. Esta automatización también es importante porque en una exploración de estas características se precisa analizar muchas imágenes, y las diferencias que se pueden observar al realizarse un estudio evolutivo pueden ser pequeñas entre dos exploraciones consecutivas.

El grado de incertidumbre que se observa depende en gran medida de las características de cada imagen y de factores como el ruido y la incidencia de la inhomogeneidad del campo magnético sobre la imagen, siendo especialmente crítico en las lesiones más pequeñas, que presentan un elevado grado de vaguedad en su contorno.

Teniendo en cuenta lo anterior se propone la implementación de un algoritmo siguiendo un conjunto de reglas basadas en la metodología utilizada por los neuroradiólogos para la detección de lesiones, en el que se ha introducido un análisis difuso de la información.

## 2 ESTRUCTURA DEL ALGORITMO

En el presente estudio se consideran imágenes del cráneo obtenidas por RM con potenciaciones en densidad y T2. La adquisición de estas imágenes se realiza en un Siemens Magnetom 1.5T, utilizando una secuencia $SE_{2200/12-80}$ con una matriz de 192x256 sobre un FOV de 250 mm considerando secciones de 5 mm de grosor separadas 1 mm. Previamente a la adquisición de estas imágenes se ha administrado contraste al paciente.

El diagnóstico de esclerosis múltiple en el paciente a partir de las imágenes adquiridas anteriormente, se basa en la observación, por parte del neuroradiólogo, de comportamiento hiperintenso respecto a su entorno. Esta hiperintensidad está asociada a la desmielinización de los axones permitiendo observar en ambas imágenes, para la zona lesional, niveles de gris elevados respecto a su entorno.

La diferenciación de estas regiones respecto a otras que pueden presentar comportamientos similares conduce a establecer una serie de reglas que permiten caracterizar el denominado comportamiento hiperintenso. Las reglas a considerar son las siguientes:

-Las lesiones se localizan únicamente en el parénquima encefálico.

-Las lesiones presentan comportamiento hiperintenso tanto en la imagen potenciada en T2 como en densidad.

-La imagen potenciada en T2 permite apreciar mejor las lesiones, pero esta potenciación presenta muchas regiones con comportamiento hiperintenso que no corresponden a lesiones. Esto no se observa tanto en la imagen densidad.

-Las lesiones presentan formas elípticas, aunque si se produce superposición por proximidad entre lesiones pueden observarse formas muy diversas. Su tamaño no acostumbra a superar los 9 pixels para su eje mayor, mientras que el mínimo se puede situar en 3 ó 4 pixels.

-Si la lesión se sitúa en la frontera entre los ventrículos y el parénquima sólo se considera la presencia de lesión si se observa una variación de intensidad de nivel de gris muy importante respecto a su entorno (situación menos habitual) o si la lesión se extiende describiendo una variación importante respecto al contorno de la región ventricular (generalmente se observa ortogonalidad).

Las reglas anteriores son aplicables a las diferentes lesiones que se puedan observar con independencia de su tamaño. Las lesiones pequeñas, objeto de este estudio, se definen como aquellas que se pueden incluir en el interior de una ventana de 5x5.

El algoritmo que se utiliza para la detección a partir de las reglas anteriores, consta de las siguientes partes:

1.-Normalización de los valores de nivel de gris de las imágenes originales (rehistogramación no lineal).

2.-Extracción y evaluación de las características.

3.-Fuzzificación.

4.-Análisis de datos mediante reglas.

5.-Defuzzificación.

6.-Filtrado.

De forma previa a la normalización de los valores de nivel de gris se realiza la adquisición de 19 imágenes por estudio con el equipo de RM de 1.5T descrito, utilizando la secuencia $SE_{2200/12-80}$ con los parámetros citados anteriormente. Su análisis se realiza en una estación de trabajo SGI Indigo2 XZ. La normalización de los valores se realiza entonces mediante una rehistogramación no lineal que consiste en asociar el valor 0 a los primeros

16000 pixels de valor más bajo y 242 a los 100 de valor más elevado; el resto de los pixels se distribuyen de forma lineal entre estos dos valores de acuerdo con su valor en la escala de niveles de gris.

# 3 EXTRACCIÓN DE CARACTERÍSTICAS

Se han considerado tres características que se basan en el nivel de gris asociado a los pixels internos de las lesiones y el de los pixels situados en la corona exterior a éstas. Se toman, junto con los valores anteriores, diferentes valores relativos obtenidos entre los valores interiores y exteriores.

En la extracción de características se han considerado 7 pacientes seleccionados de estudios obtenidos utilizando la secuencia $SE_{2200/12\text{-}80}$. Las diferentes lesiones que se han observado han sido marcadas por diferentes radiólogos. De éstas se han seleccionado únicamente aquellas que mostraban claramente la placa circunscrita a una ventana de 5x5. Para realizar la extracción de los valores se superpone una ventana de 7x7 sobre las posibles lesiones pequeñas y se adquieren valores únicamente cuando ningún pixel del interior de la lesión ocupa las filas o columnas exteriores de la ventana de 7x7. Se adquieren dos tipos de valores: los que corresponden al interior de la lesión, y los que son claramente de su exterior, creando una corona alrededor de los valores de lesión. La recogida de datos se realizó para las dos potenciaciones consideradas, obteniéndose más valores en el caso de las imágenes T2 dado que las placas se observan con mayor claridad.

A partir del espacio de características se determina la presencia de comportamiento hiperintenso en una región a mediante la aplicación de las siguientes reglas:

1.-En una región existe un mayor comportamiento hiperintenso si el nivel de gris de los pixels que la componen es alto.

2.-En una región existe un mayor comportamiento hiperintenso si existe un número suficientemente elevado de radios en los que se observa que la diferencia entre el valor central y el valor de los pixels del radio es relativamente elevada, aunque de valor claramente inferior al valor central.

3.- En una región existe un mayor comportamiento hiperintenso si existe un número suficientemente elevado de radios en los que se observa que el valor central supera con cierta claridad valores de los pixels del radio.

A fin de implementar las reglas anteriores se consideran las siguientes características para las dos potenciaciones:

-Nivel de gris.

-Cociente de la diferencia entre valor de lesión y valor de contorno respecto al valor de lesión.

-Cociente entre valor de lesión y valor de contorno.

Para la evaluación de las características anteriores se centra, en cada pixel, una ventana de 7x7. La primera característica corresponde al nivel de gris del pixel central. La segunda y la tercera característica se evalúan para cada uno de los ocho radios direccionales, tomándose para cada característica y radio los tres valores posibles, dado que no se conoce a priori por cual de los tres pixels, del radio, pasa la corona de la lesión.

# 4 FUZZIFICACIÓN

A cada una de las características anteriores se le asocia un conjunto difuso (se obtienen, por tanto seis conjuntos difusos: $fs_{1T2}$, $fs_{2T2}$, $fs_{3T2}$, $fs_{1D}$, $fs_{2D}$, $fs_{3D}$). El grado de pertenencia de cada pixel a estos conjuntos se obtiene, tal y como introdujeron Montseny-Sobrevilla [6], a partir de funciones de densidad de probabilidad que se han obtenido a partir de los valores del conjunto de diseño para cada característica. A partir de estos modelos se realiza una fuzzificación que permite transformar el espacio de características en una función que toma valores en [0,1]. Esto se aplica a las 3 características consideradas y a los dos tipos de imágenes (densidad y T2), por lo que se dispone de 6 funciones.

Los valores obtenidos, para las tres características y las dos potenciaciones, se fuzzifican de acuerdo con las funciones definidas en el párrafo anterior. De esta forma, se obtiene el grado de pertenencia, del pixel central, a los tres conjuntos difusos. Para la segunda y tercera características la obtención del grado de pertenencia de cada pixel a los correspondientes conjuntos difusos se realiza de la siguiente forma:

-Los tres valores correspondientes a cada radio se agregan utilizando el operador OWA [1,0,0].

-Los ocho valores obtenidos en el paso anterior, correspondientes a los distintos radios, se agregan utilizando el operador OWA [0,0,0,1,0,0,0,0].

# 5 ANÁLISIS DE LOS DATOS MEDIANTE REGLAS

Para el análisis de los datos considera la regla introducida en el apartado 2: "las lesiones presentan comportamiento hiperintenso tanto en la imagen T2 como en densidad", así como la definición de lesión pequeña. Esto se ha traducido en tres reglas de implicación:

-Si el pixel (i,j) pertenece a los conjuntos difusos $fs_{1T2}$, $fs_{2T2}$, $fs_{3T2}$, entonces el pixel pertenece a $fs_{T2}$.

-Si el pixel (i,j) pertenece a los conjuntos difusos $fs_{1D}$, $fs_{2D}$, $fs_{3D}$, entonces el pixel pertenece a $fs_D$ .

-Si el pixel (i,j) pertenece a los conjuntos difusos $fs_D$, $fs_{T2}$, entonces el pixel pertenece a $fs_H$ (conjunto difuso de pixels con comportamiento hiperintenso).

El grado de pertenencia de cada uno de los pixels a los nuevos conjuntos difusos se obtiene mediante los operadores OWA:

-[0,1,0] para el conjunto difuso $fs_{T2}$.

-[0,1,0] para el conjunto difuso $fs_D$.

-[0,1] para el conjunto difuso $fs_H$.

# 6 DEFUZZIFICACIÓN

El proceso de defuzzificación permite la obtención de las agrupaciones de pixels que pueden corresponder a lesiones pequeñas. Para ello se realiza un $\alpha$-corte ($\alpha$=0.5) sobre el conjunto difuso $fs_H$.

El resultado de la aplicación de este proceso es la detección de toda un conjunto de agrupaciones de pixeles en la región objeto de análisis. Entre estas agrupaciones aparecen algunas que corresponden a lesiones no pequeñas (según la definición fijada) y a regiones donde no se observa lesión. Estos dos grupos son considerados falsas detecciones.

# 7 POSTFILTRADO

La eliminación de las falsas detecciones se basa en una nueva implementación de reglas que consideran los aspectos diferenciadores respecto a las lesiones pequeñas reales, así como la definición mediante reglas de comportamiento hiperintenso (apartado 2). De esta forma las reglas que se consideran son las siguientes:

-Se consideran lesiones aquellas agrupaciones que presenten un número mínimo de pixels.

-Se consideran lesiones pequeñas aquellas agrupaciones que no superen los 25 pixels.

-Las lesiones se localizan únicamente en el parénquima encefálico.

-Si la lesión se sitúa en la frontera entre los ventrículos y el parénquima sólo se considera la presencia de lesión si se observa una variación de intensidad de nivel de gris muy importante respecto a su entorno (situación menos habitual) o si la lesión se extiende describiendo una variación importante respecto al contorno de la región ventricular (generalmente se observa ortogonalidad).

-Las lesiones no pueden estar situadas en la separación de los dos hemisferios en los que se puede dividir el cerebro.

-Las lesiones no pueden estar situadas en los surcos subaracnoideos de la convexidad de hemisferios cerebrales.

Las dos primeras reglas se implementan mediante filtros que analizan el tamaño de las agrupaciones. Para implementar el resto de las reglas se tiene en cuenta las localizaciones de las distintas agrupaciones obtenidas.

# 8 RESULTADOS

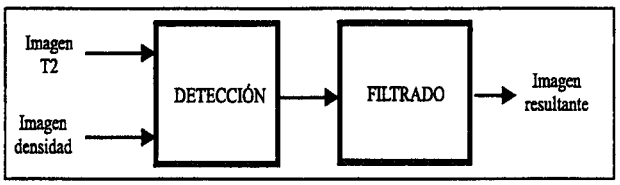



Figura 1: Proceso de obtención de los resultados

A fin de analizar los resultados, el algoritmo se ha descompuesto en dos fases (figura 1), la primera de las cuales engloba las partes 1 a 5 del algoritmo descrito en el apartado 2. Siguiendo este proceso, el algoritmo se ha aplicado sobre 63 imágenes obteniéndose el siguiente resultado:

-Detecciones correctas: 91.43 %.

-Falsas detecciones (media por imagen): 203.66

donde se puede observar que:

a)casi la totalidad de las lesiones pequeñas son detectadas.

b)es necesario eliminar falsas detecciones.

Para eliminar las falsas detecciones, en primer lugar se ha considerado el filtrado atendiendo al tamaño de las lesiones, mediante la utilización de dos tipos de filtros:

A: Se consideran lesiones pequeñas aquellas agrupaciones que constan de 3 a 25 pixels.

B: Se consideran lesiones pequeñas aquellas agrupaciones que constan de 4 a 25 pixels.

obteniéndose el resultado que se muestra en la tabla 1.

Tabla 1: Niveles de detección según filtrado por tamaño de las agrupaciones de pixels.

| | **Detección** | **Filtrado A** | **Filtrado B** |
|---|---|---|---|
| Detecciones correctas | 91.43 % | 82.86 % | 62.86 % |
| Falsas detecciones (media por imagen) | 203.66 | 78.75 | 59.95 |

En segundo lugar, se ha considerado el filtrado atendiendo a la localización de la agrupaciones. De este modo se ha tenido en cuenta:

C: Las lesiones no pueden estar situadas en la separación de los dos hemisferios en los que se puede dividir el cerebro.

D: Las lesiones no pueden estar situadas en los surcos subaracnoideos de la convexidad de hemisferios cerebrales.

Obteniéndose el resultado que muestra la tabla 2.

Tabla 2: Niveles de detección según filtrado por tamaño y por localización.

| | **Detección** | **Filtrado A** | **Filtrado A,C y D** |
|---|---|---|---|
| Detecciones correctas | 91.43 % | 82.86 % | 80 % |
| Falsas detecciones (media por imagen) | 203.66 | 78.75 | 20.35 |

Las imágenes 2 a 5 muestran el proceso de obtención de los resultados a partir de las dos imágenes a analizar (densidad y T2) para un caso concreto, considerando dos situaciones de filtrado.

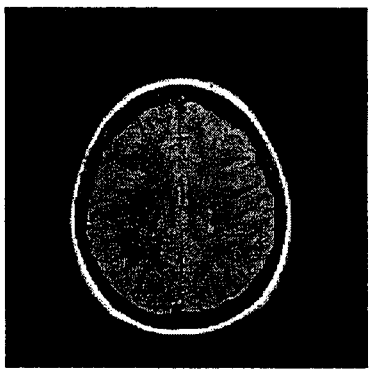


Figura 2: Imagen potenciada en densidad.

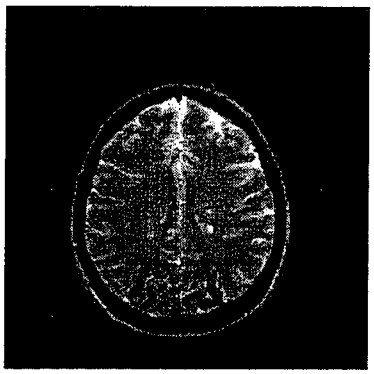


Figura 3: Imagen potenciada en T2.



Figura 4: Resultado de aplicar detección y filtrado A.



Figura 5: Resultado de aplicar detección y filtrado A, C y D.

## 9 CONCLUSIONES

El presente estudio muestra el diseño de un algoritmo para la detección de lesiones pequeñas en esclerosis múltiple a partir de imágenes de RM. La implementación realizada no detecta únicamente aquellas lesiones para las que se diseñó, sinó que permite detectar otras de dimensiones superiores dado que no se han impuesto restricciones de forma.

Este estudio ha permitido determinar en 3 pixels el tamaño mínimo límite para la detección de las lesiones, por debajo del cual no se aprecian mejoras en los resultados, pero si un notable aumento en el número de falsas detecciones.

La aplicación de técnicas difusas ha permitido, por una parte, formalizar el comportamiento hiperintenso permitiendo la detección de lesiones pequeñas. Por otra, ha facilitado la implementación de las reglas que utilizan los neuroradiólogos para la detección de estas lesiones.

Si bien en estos momentos el número de falsas detecciones es excesivo, se ha observado que, mediante la implementación de nuevas reglas de filtrado basadas en la localización, podrán ser eliminadas.

En estos momentos se está trabajando en la mejora del algoritmo en las siguientes líneas:

-Mejora de la capacidad de detección mediante la optimización del uso de la función de distribución de probabilidad.

-Aumento de los niveles de filtrado a partir de la máxima segmentación posible de la región de interés.

-Implementación de funciones de validación que permitan mejorar la detección.

**Agradecimientos**

Este trabajo ha sido parcialmente subvencionado a través de la ayuda CICYT ref. BIO95-0916-C02-01. Asimismo, los autores desean agradecer la colaboración del personal del I.D.I..

**Referencias**

[1] M.K. Edwards and J.M. Bonnin. White Matter Diseases. Magnetic Resonance Imaging of the Brain and Spine. S.W. Atlas. Raven Press 1991. Pág. 467-477.

[2] M. Filippi, M.A. Horsfield, P.S. Tofts, F. Barkhof, A.J. Thompson and D. H. Miller. Quantitative assessment of MRI lesion load in monitoring the evolution of multiple sclerosis. Brain 1995; 118, 1601-1612.

[3] M..M. Gupta, G.K. Knopf. Fuzzy logic and uncertainty in image processing. Proceedings SPIE, vol. 1902, pág. 210-223, 1993.

[4] E.F. Jackson, P.A. Narayana, J.S. Wolinsky, and T.J. Doyle. Accuracy and reproducibility in volumetric analysis of multiple sclerosis lesions. . J. Comput. Assist. Tomogr., Vol. 17(2), 1993; pág. 200-205.

[5] J.R. Mitchell, S.J. Karlik, D.H. Lee, A. Fenster. Computer-assisted Identification of Multiple Sclerosis Lesions in MR Imaging Volumes in the Brain. JMRI 1994; 4: 197-208.

[6] E. Montseny, P. Sobrevilla. Aplicación de técnicas difusas al diseño de algoritmos para visión por computador. VI Congreso Español Sobre Tecnologías y Lógica Fuzzy (estylf 96) Oviedo; pág. 207-212.

[7] K. Pal et al.. Image Description and Primitive Extraccion Using Fuzzy Sets. IEEE Trans. SCM 13, 1983.

[8] F. Pannizzo, M.J.B. Stallmeyer, J. Friedman, R.J. Jennis, J. Zabriskie, C. Plank, R. Zimmerman, J.P. Whalen and P.T. Cahill. Quantitative MRI Studies for Assessment of Multiple Sclerosis. Magn. Res. Med. 1992; 24, 90-99.

[9] S. Samarasekera, J. K. Udupa, Y. Miki, L. Wei, and R.I. Grossman. A new computer-assisted method for the quantification of enhancing lesions in multiple sclerosis. J. Comput. Assist. Tomogr., Vol. 21, No. 1, 1997; pág. 145-151.

[10] D.A.G. Wicks, P. S. Tofts, D.H. Miller, G.H. du Boulay, A. Feinstein, R.P. Sacares, I. Harvey, R. Brenner and W.I. McDonald. Volume measurement of multiple sclerosis lesions with magnetic resonance images. A preliminary study. Neuroradiology 1992; 34, 475-479.

[11] R.R. Yager. Families of OWA operators. Fuzzy Sets and Systems n. 59, pág 125-148, 1993.

# Una primera aproximación a la unificación fuzzy vía relaciones de compatibilidad*

**T. Alsinet**
Departament d'Informàtica
Universitat de Lleida (UdL)
Ap. Correos 471, 25080 Lleida, Spain
e_mail:tracy@eup.udl.es

## Resumen

Este trabajo surge de la necesidad de incorporar el tipo de datos *conjunto fuzzy* en un entorno de programación lógica basado en una familia de lógicas infinitamente valuadas. Para ello, introducimos el concepto de unificador fuzzy más general y proporcionamos un algoritmo para su construcción. Dicho algoritmo utiliza una relación de compatibilidad o de proximidad fuzzy basada en la interpretación de la función de pertenencia de un conjunto fuzzy como un grado de similitud entre los elementos del universo sobre el cual está definido y su conjunto de prototipos.

**Palabras clave:** relaciones de compatibilidad, unificación fuzzy, variables lingüísticas, constantes fuzzy.

## 1 Introducción

Los lenguajes de programación lógica han sido aplicados a distintas áreas como la inteligencia artificial y las bases de datos deductivas. Entre estos lenguajes Prolog es el más representativo; sin embargo, no permite representar ni tratar información vaga, imprecisa e incompleta. Con objeto de solucionar dicho problema han sido propuestas distintas soluciones [3, 6, 8, 11, 12, 14, 15, 19] Todas ellas implementan procedimientos de prueba para el razonamiento fuzzy y difieren unas de otras en la lógica subyacente.

Aparentemente, la incorporación de constantes fuzzy a los lenguajes de programación lógica fue tratada por primera vez a principios de los ochenta por Cayrol et al. [5] con objeto de incorporar valores fuzzy a un procedimiento de pattern matching. Posteriormente, en el lenguaje Fril [4] se implementó un procedimiento de unificación semántica basado en la asignación de masas. Recientemente, Virtanen [18] presentó un algoritmo de unificación fuzzy basado en relaciones de igualdad fuzzy y Rios-Filho et al. [16] introdujeron la distinción entre conocimiento general y específico, dando lugar a una unificación dicotómica basada en medidas de inclusión y de semejanza. Finalmente, Dubois et al. [7] han desarrollado un algoritmo de unificación basado en la medida de necesidad de un evento fuzzy $A$ a partir de la información fuzzy $B$.

El presente trabajo surge como necesidad de incorporar el tipo de datos *conjunto fuzzy* a un entorno de programación lógica basado en una familia de lógicas infinitamente valuadas [1]. Así, la sección 2 establece la sintaxis de los programas soportados por el algoritmo de unificación fuzzy y define la semántica asociada con los términos. La sección 3 define la medida de similitud o grado de proximidad entre dos conjuntos fuzzy utilizada por el sistema. La sección 4 introduce las definiciones de sustitución, de unificador y de unificador fuzzy más general, desarrolla un algoritmo para la construcción del unificador fuzzy más general de un par de fórmulas atómicas y, finalmente, presenta el grado de compatibilidad asociado. La última sección establece las conclusiones y futuras líneas de trabajo.

## 2 Sintaxis de la lógica fuzzy

Nuestro objetivo es incorporar el tipo de datos *conjunto fuzzy* a un sistema de programación lógica basado en una familia de lógicas infinitamente valuadas. Para ello, hemos considerado los aspectos sintácticos y semánticos que intervienen en el proceso de unificación cuando los términos de las fórmulas atómicas tienen una componente semántica como es el caso de las constantes fuzzy y variables lingüísticas.

**Definición 1 (variable lingüística)** *Caracterizamos una variable lingüística mediante una quíntupla* $(v, T_L, \Omega, g, m)$ *en la cual* $v$ *es el nombre de la vari-*

Trabajo financiado parcialmente por la CICYT, proyecto SMASH (TIC96-1038-C04-03)

*able, $T_L$ es el conjunto de términos lingüísticos de $v$ definidos sobre el universo $\Omega$, $g$ es una regla sintáctica para la generación de términos lingüísticos y $m$ es una regla semántica que asigna a cada término lingüístico un conjunto fuzzy definido sobre el universo $\Omega$.*

**Definición 2 (constante fuzzy)** *Una constante fuzzy es un término lingüístico.*

Las constantes fuzzy manejadas por el sistema de unificación son conjuntos fuzzy normales trapezoidales. Utilizaremos el nombre de un conjunto fuzzy para denotar su función de pertenencia.

**Definición 3 (término)** *Un término es una constante, una constante fuzzy, una variable o una variable lingüística.*

**Definición 4 (fórmula atómica)** *Si $p$ es un símbolo de predicado de aridad $n$ y $t_1, \ldots, t_n$ son términos, entonces $p(t_1, \ldots, t_n)$ es una fórmula atómica. Si $p(t_1, \ldots, t_n)$ es una fórmula atómica y $\alpha \in [0,1]$, entonces $(p(t_1, \ldots, t_n); \alpha)$ es una fórmula atómica multivaluada.*

**Definición 5 (programa)** *Un programa $P$ se compone de un conjunto finito de hechos y reglas multivaluados definidos de la siguiente forma:*

- **hecho:** *es una fórmula atómica multivaluada*
- **regla:** *es un par $(q \leftarrow p_1, \ldots, p_k; \alpha)$ donde $q, p_1, \ldots, p_k$ son fórmulas atómicas y $\alpha \in [0,1]$.*

**Definición 6 (semántica de los término)**

- *La semántica de una constante es el valor que denota.*
- *La semántica de una constante fuzzy es el conjunto fuzzy que denota.*
- *La semántica de una variable se define como el valor que toma dicha variable en un programa. Representamos dicha componente semántica como una lista de nombres de constantes fuzzy y la llamamos estructura semántica de la variable.*
- *La semántica de una variable lingüística se compone de la quíntupla $(v, T_L, \Omega, g, m)$ que la caracteriza y del valor que toma dicha variable en un programa. Al igual que en el caso de las variables, representamos esta segunda componente semántica como una lista de nombres de constantes fuzzy y la llamamos estructura semántica de la variable lingüística.*

## 3 Relaciones de compatibilidad

El problema de la unificación en lógica clásica es puramente sintáctico, es decir, dos constantes unifican si son sintácticamente iguales (denotan el mismo valor). Cuando afrontamos el problema de la unificación fuzzy debemos considerar la componente semántica de las constantes fuzzy y de las variables lingüísticas resultando necesario definir alguna medida que determine el grado de similitud, o el grado de proximidad entre dos conjuntos fuzzy.

**Definición 7 (relaciones de equivalencia o similitud fuzzy)** *Una relación binaria fuzzy $S$ definida sobre el universo $W$, $S : W \times W \rightarrow [0,1]$, es una relación de $*$-similitud, donde $*$ es una t-norma, si satisface las siguientes propiedades [10]:*

1. *reflexiva: $S(w,w) = 1$*
2. *simétrica: $S(w,w') = S(w',w)$*
3. *$*$-transitiva: $S(w,w') * S(w',w'') \leq S(w,w'')$*

El algoritmo de unificación fuzzy desarrollado por Virtanen [18] define un grado de igualdad aproximada, $\cong$, a través de una relación de inclusión aproximada, $\overset{\subseteq}{\approx}$, basada en la implicación y conjunción de Łukasiewicz. Concretamente, el grado de igualdad de dos términos $w$ y $w'$ se obtiene en base a la siguiente expresión: $[w \cong w'] = w \overset{\subseteq}{\approx} w' \wedge w' \overset{\subseteq}{\approx} w$ [9].

De forma intuitiva, cualquier relación de igualdad fuzzy deberá ser definida vía alguna relación de inclusión restringiendo la capacidad de unificación por la propiedad $*$-transitiva de la relación. La solución que proponemos consiste en utilizar relaciones de proximidad fuzzy.

**Definición 8 (relaciones de compatibilidad fuzzy)** *Una relación binaria fuzzy $C$ definida sobre el universo $W$, $C : W \times W \rightarrow [0,1]$, es una relación de compatibilidad o de proximidad si cumple las propiedades reflexiva y simétrica.*

**Definición 9 (clase de $\alpha$-compatibilidad)** *Una clase de $\alpha$-compatibilidad es un subconjunto $A$ de $W$ tal que para todo $w, w' \in A$ se cumple $C(w,w') \geq \alpha$.*

Sea $A \subseteq \Omega$ un subconjunto no vacío de $\Omega$. Una relación de $*$-similitud $S$ permite definir un conjunto fuzzy normal no vacío $A^*$ de los elementos *cercanos* a $A$ de la siguiente forma [10]: $A^*(w) = max_{w' \in A} S(w,w')$.

Por otro lado, por el teorema de Valverde [17], referente a la representación de relaciones de $*$-similitud a través de conjuntos fuzzy y basada en la residuación de la t-norma $*$, cualquier conjunto fuzzy normal

$F$ definido sobre un universo $\Omega$ puede ser interpretado como una fuzzyficación a partir de una relación de $*$-similitud generada por $F$ y un subconjunto no vacío $A$ de $\Omega$:

$$A = \{w \mid F(w) = 1\}$$

$$S(w, w') = min(F(w) \Rightarrow F(w'), F(w') \Rightarrow F(w))$$

Dicho resultado proporciona una justificación formal al hecho de que el grado de pertenencia de un elemento $w$ a un fuzzy set $F$, $F(w)$, puede ser interpretado como el grado de similitud de $w$ y los prototipos de $F$, siendo los prototipos de $F$ el conjunto $A$.

Basándonos en la interpretación anterior de un conjunto fuzzy, procedemos a definir la relación de compatibilidad utilizada por el algoritmo de unificación de la sección 4.

**Definición 10 (grado de similitud media entre prototipos)** *Sean $A$ y $B$ dos conjuntos fuzzy normales trapezoidales definidos sobre el universo $\Omega$ con conjuntos de prototipos $A'$ y $B'$, respectivamente. Definimos el grado de similitud media de los prototipos de $A$ respecto los prototipos de $B$, $SM(A'/B')$, como el valor medio de la función de pertenencia del conjunto fuzzy $B$ en el intervalo $[min(A'), max(A')]$.*

El algoritmo de unificación desarrollado se basa en una relación binaria de compatibilidad, de proximidad o de tolerancia fuzzy, $C$, definida sobre $\mathcal{F}(\Omega)$, $C : \mathcal{F}(\Omega) \times \mathcal{F}(\Omega) \rightarrow [0, 1]$, donde $\mathcal{F}(\Omega)$ denota el conjunto de todos los conjuntos fuzzy normales trapezoidales que pueden ser definidos sobre el universo $\Omega$.

**Definición 11 (grado de compatibilidad entre dos conjuntos fuzzy)** *Sean $A$ y $B$ dos conjuntos fuzzy normales trapezoidales definidos sobre el universo $\Omega$ con conjuntos de prototipos $A'$ y $B'$, respectivamente. Definimos el grado de compatibilidad de $A$ y $B$ como $C(A, B) = T(SM(A'/B'), SM(B'/A'))$, donde $T$ es una t-norma.*

## 4 Unificación fuzzy

En esta sección introducimos las definiciones de sustitución, de unificador y de unificador fuzzy más general. Desarrollamos un algoritmo, basado en la relación de compatibilidad entre conjuntos fuzzy introducida en la sección anterior, para la construcción del unificador fuzzy más general y presentamos el grado de compatibilidad asociado.

**Definición 12 (términos de sustitución)** *Un término de sustitución de una variable es una variable, una constante o una variable lingüística. Un término de sustitución de una variable lingüística es una variable lingüística.*

**Definición 13 (sustitución fuzzy)** *Una sustitución fuzzy $\sigma$ consta de dos componentes:*

- *componente sintáctica: es una función finita de las variables en los términos de sustitución de las variables y de las variables lingüística en los términos de sustitución de las variables lingüísticas. Representamos la componente sintáctica de una sustitución $\sigma$ como $\theta = \{x_1/t_1, ..., x_n/t_n\}$, donde $x_1, ..., x_n$ son variables y variables lingüísticas sintácticamente distintas y para todo $i$, $1 \leq i \leq n$, la variable o variable lingüística $x_i$ es sintácticamente distinta al término de sustitución $t_i$.*
- *componente semántica: es una función finita de las variables, variables lingüísticas y variables ficticias en las estructuras semánticas. Una variable ficticia es una variable asociada a un par de contantes fuzzy. Representamos la componente semántica de una sustitución $\sigma$ como $CS = \{x_1 \rightarrow l_1, ..., x_n \rightarrow l_n\}$, donde $x_1, ..., x_n$ son variables, variables lingüísticas o variables ficticias sintácticamente distintas y $l_1, ..., l_n$ son estructuras semánticas.*

En lo sucesivo denotaremos una sustitución fuzzy $\sigma$ con componente sintáctica $\theta$ y componente semántica $CS$ como $\sigma(\theta, CS)$.

Las sustituciones operan sobre expresiones. Una expresion es un término o una secuencia de fórmulas atómicas.

**Definición 14 (instancia fuzzy de una expresión)** *Dada una expresión $E$ y una sustitución fuzzy $\sigma(\theta, CS)$, llamaremos instancia fuzzy de la expresión $E$, que notaremos por $E\sigma$, al resultado de aplicar secuencialmente los dos siguientes pasos:*

1. *sustitución sintáctica: consiste en aplicar la componente sintáctica $\theta$ de la sustitución fuzzy $\sigma$ a la expresión $E$, el resultado de la cual se obtiene remplazando simultáneamente cada ocurrencia en $E$ de una variable y variable lingüística del dominio de $\theta$ por el correspondiente término de sustitución. Representaremos la expresión resultante de aplicar la componente sintáctica $\theta$ a la expresión $E$ como $E\theta$.*
2. *sustitución semántica: consiste en aplicar la componente semántica $CS$ de la sustitución fuzzy $\sigma$ a la expresión $E\theta$, el resultado de la cual se obtiene*

*enlazando cada variable y variable lingüística de $E\theta$ con estructura semántica en $CS$ con dicha estructura.*

**Definición 15 (composición de sustituciones fuzzy)** *Sean $\sigma_1(\theta_1, CS_1)$ y $\sigma_2(\theta_2, CS_2)$ dos sustituciones fuzzy siendo $\theta_1 = \{x_1/t_1, ..., x_n/t_n\}$ y $\theta_2 = \{y_1/s_1, ..., y_m/s_m\}$. La composición de $\sigma_1$ y $\sigma_2$ es una sustitución $\sigma(\theta, CS)$, donde*

- *$\theta$ se obtiene como resultado de eliminar del conjunto $\{x_1/t_1\theta_2, ..., x_n/t_n\theta_2, y_1/s_1, ..., y_m/s_m\}$ los pares $x_i/t_i\theta_2$ tales que $t_i\theta_2$ es $x_i$ y los pares $y_i/s_i$ tales que $y_i \in \{x_1, ..., x_n\}$.*
- *$CS$=$(CS_1 \backslash CS_2) \cup CS_2$*

**Definición 16 (igualdad entre sustituciones fuzzy)** *Dos sustituciones fuzzy $\sigma_1(\theta_1, CS_1)$ y $\sigma_2(\theta_2, CS_2)$ son iguales si $\theta_1 = \theta_2$ y $CS_1 = CS_2$.*

**Definición 17 (sustitución fuzzy más general)** *Una sustitución fuzzy $\sigma_1$ es más general que una sustitución fuzzy $\sigma_2$ si existe alguna sustitución fuzzy $\sigma_3$ tal que $\sigma_2 = \sigma_1\sigma_3$.*

**Definición 18 (unificador fuzzy)** *Sean $A$ y $B$ dos fórmulas atómicas con el mismo símbolo de predicado $p$ de aridad $n$, y $\sigma(\theta, CS)$ una sustitución fuzzy. Si $A\sigma$ es de la forma $p(s_1, ..., s_n)$ y $B\sigma$ es de la forma $p(t_1, ..., t_n)$, diremos que $\sigma$ es un unificador fuzzy de $A$ y $B$ si para todo $i$, $1 \leq i \leq n$, se cumple alguna de las siguientes condiciones:*

- *$s_i$ y $t_i$ son sintáctica y semánticamente iguales*
- *$s_i$ es una variable o una variable lingüística y $t_i$ es una constante fuzzy, o viceversa, y $t_i$ pertenece a la estructura semántica de $s_i$*
- *$s_i$ y $t_i$ son constantes fuzzy y existe en $CS$ una variable ficticia con estructura semántica asociada $s_i$ y $t_i$*

**Definición 19 (unificador fuzzy más general)** *Un unificador fuzzy $\sigma$ de un par de fórmulas atómicas $A$ y $B$ es su unificador fuzzy más general, si es más general que cualquier otro unificador fuzzy de $A$ y $B$.*

**Proposición 1** *Existe un algoritmo el cual para cualquier par de fórmulas atómicas produce como salida su unificador fuzzy más general si son fuzzy unificables y, en otro caso, devuelve la no existencia de un unificador fuzzy.*

Para el desarrollo del algoritmo seguiremos la presentación de Apt [2] basada en el algoritmo original de Herbrand el cual encuentra la solución a un conjunto finito de ecuaciones de términos. Este algoritmo fue presentado por primera vez por Martelli y Montanari [13].

**Algoritmo 1 (Algoritmo de unificación fuzzy)**

**Entrada:** dos fórmulas atómicas con símbolo de predicado $p$ de aridad $n$ de la forma $p(s_1, ..., s_n)$ y $p(t_1, ..., t_n)$ tales que no tienen ninguna variable ni variable lingüística en común.

**Salida:** el unificador fuzzy más general $\sigma(\theta, CS)$ si son fuzzy unificables y, en otro caso, *error*.

**Inicializaciones:**

- A partir de las fórmulas atómicas $p(s_1, ..., s_n)$ y $p(t_1, ..., t_n)$ construir un conjunto de pares de términos de la forma $\{(s_1, t_1), ..., (s_n, t_n)\}$, en los cuales cada variable y variable lingüística mantiene el enlace a la estructura semántica asociada en las fórmulas atómicas
- $CS := \emptyset$

**Método:**

Selecionar cualquier par $(s_i, t_i)$ del conjunto de pares de términos y ejecutar de entre las siguientes acciones la que corresponda hasta que el conjunto de pares de términos y $CS$ no se modifiquen

1. $s_i$ y $t_i$ son constantes:
   **si** ($s_i$ y $t_i$ denotan el mismo valor)
   . Eliminar el par $(s_i, t_i)$ del conjunto
   **si no** devuelve *error*
2. $s_i$ es una variable lingüística o una constante fuzzy y $t_i$ es una constante o, viceversa devuelve *error*
3. $s_i$ no es una variable y $t_i$ es una variable o bien, $s_i$ es una constante fuzzy y $t_i$ es una variable lingüística: remplazar $(s_i, t_i)$ por $(t_i, s_i)$
4. $s_i$ es una variable y $t_i$ es una constante:
   **si** (vacía(estructura semántica de $s_i$))
   **si** ($s_i$ tiene otra ocurrencia en el resto de pares de términos del conjunto)
   . Aplicar la sustitución $(\{s_i/t_i\}, \emptyset)$ al resto de pares de términos del conjunto
   **fin si**
   **si no** devuelve *error*
5. $s_i$ y $t_i$ son constantes fuzzy:
   **si** ($s_i$ y $t_i$ no denotan la misma constante fuzzy)

**si** ($\neg\exists$ una variable ficticia para el par $(s_i, t_i)$ en $CS$)

**si** (universo($s_i$)=universo($t_i$))

. $CS$:=$CS \cup \{(s_i, t_i) \rightarrow [s_i|t_i]\}$, siendo $(s_i, t_i)$ el nombre de la variable ficticia y $[s_i|t_i]$ la lista de constantes fuzzy asociada

**si no** devuelve *error*

**fin si**

Eliminar el par $(s_i, t_i)$ del conjunto

6. $s_i$ es una variable y $t_i$ es una variable o una variable lingüística, o bien, $s_i$ y $t_i$ son variables lingüísticas:

**si** ($s_i$ y $t_i$ son variables lingüísticas con distinto conjunto de términos lingüísticos)

. devuelve *error*

**si no**

**si** (vacía(estructura semántica de $s_i$))

**si** ($s_i$ tiene otra ocurrencia en el resto de pares de términos del conjunto)

. Aplicar la sustitución $(\{s_i/t_i\}, \emptyset)$ al resto de pares de términos del conjunto

**fin si**

**si no**

**si** ($t_i$ es una variable o $s_i$ es una variable lingüística, y vacía(estructura semántica de $t_i$))

. Remplazar $(s_i, t_i)$ por $(t_i, s_i)$

**si no**

**si** (universo($s_i$)=universo($t_i$))

. $CS$:=$(CS \backslash \{s_i \rightarrow l_{s_i}, t_i \rightarrow l_{t_i}\}) \cup \{s_i \rightarrow [\,|\,], t_i \rightarrow \texttt{concatenar}(l_{t_i}, l_{s_i})\}$, siendo $l_{s_i}$ y $l_{t_i}$ las listas de constantes fuzzy de las estructuras semánticas de $s_i$ y $t_i$, respectivamente

. Aplicar la sustitución $(\{s_i/t_i\}, CS)$ al resto de pares de términos del conjunto

. Aplicar la sustitución $(\emptyset, CS)$ al par $(s_i, t_i)$

**si no** devuelve *error*

7. $s_i$ es una variable o una variable lingüística y $t_i$ es una constante fuzzy:

**si** ($t_i \notin$ estructura semántica de $s_i$)

**si** (universo($s_i$)=universo($t_i$))

. $CS$:=$(CS \backslash \{s_i \rightarrow l_{s_i}\}) \cup \{s_i \rightarrow \texttt{insertar}(l_{s_i}, t_i)\}$

. Aplicar la sustitución $(\emptyset, CS)$ al resto de pares de términos del conjunto

**si no**

. devuelve *error*

**fin si**

Eliminar el par $(s_i, t_i)$ del conjunto

**Tratamiento final:** obtención de la componente sintáctica del unificador fuzzy más general $\sigma(\theta, CS)$. Si $\{(x_1, u_1), ..., (x_k, u_k)\}$ es el conjunto de pares de términos final, entonces $\theta$:=$\{x_1/u_1, ..., x_k/u_k\}$

□

**Definición 20 (grado de compatibilidad)** *Sea $\sigma(\theta, CS)$ el unificador fuzzy más general de un par de fórmulas atómicas $A$ y $B$. $CS = \{v_1 \rightarrow l_1, ..., v_n \rightarrow l_n, vl_1 \rightarrow l_{n+1}, ..., vl_m \rightarrow l_{n+m}\}$, donde $v_1, ..., v_n$ son variables y variables ficticias, $vl_1, ..., vl_m$ son variables lingüísticas y $l_1, ..., l_n, l_{n+1}, ..., l_{n+m}$ son listas de constantes fuzzy de la forma $l_i = [cf_{i1}|cf_{i2}, ..., cf_{ik}]$. Definimos el grado de compatibilidad de $A$ y $B$ vía el el unificador fuzzy más general $\sigma$, que notaremos por $Comp_\sigma(A, B)$, como el valor real en el intervalo $[0, 1]$ calculado en base a la siguiente expresión:*

$$Comp_\sigma(A, B) = T(C_v(v_1), T(C_v(v_2), ...,$$
$$T(C_v(v_n), T(C_{vl}(vl_1), T(C_{vl}(vl_2), ...,$$
$$T(C_{vl}(vl_{m-2}), T(C_{vl}(vl_{m-1}), C_{vl}(vl_m)))...)))...)),$$

*donde*

$$C_v(v_i) = C_l(l_i) = T(C_c(cf_{i1}, l_i), T(C_c(cf_{i2}, l_i), ...,$$
$$T(C_c(cf_{i(k-2)}, l_i), C_c(cf_{i(k-1)}, l_i))...)) \; y$$

$$C_{vl}(vl_i) = max_{tl_{ij}} C_{tl}(tl_{ij}, l_{n+i}),$$

*siendo $tl_{ij}$ el término lingüístico $j$ de la variable lingüística $vl_i$.*

*Finalmente,*

$$C_c(cf_{ij}, l_i) = T(C(cf_{ij}, cf_{i(j+1)}),$$
$$T(C(cf_{ij}, cf_{i(j+2)}), ...,$$
$$T(C(cf_{ij}, cf_{i(k-1)}), C(cf_{ij}, cf_{ik}))...)) \; y$$

$$C_{tl}(tl_{ij}, l_{n+i}) = T(C(tl_{ij}, cf_{(n+i)1}),$$
$$T(C(tl_{ij}, cf_{(n+i)2}), ...,$$
$$T(C(tl_{ij}, cf_{(n+i)(k-1)}), C(tl_{ij}, cf_{(n+i)k}))...))$$

*siendo $C$ y $T$ la relación de compatibilidad fuzzy y la t-norma, respectivamente, introducidas en la definición 11.*

La interpretación de la expresión $C_{vl}(vl_i)$, $1 \leq i \leq m$, se corresponde con el hecho de remplazar cada ocurrencia de la variable $vl_i$ en las fórmulas atómicas $A$ y $B$ por el término lingüístico de $vl_i$ que produzca

el unificador más general con mayor grado de compatibilidad de las fórmulas atómicas resultantes del remplazo. Por otro lado, cada lista $l_i$, $1 \leq i \leq n$, constituye una clase de $C_l(l_i)$-compatibilidad sobre el conjunto de constantes fuzzy de la lista.

## 5 Conclusiones y futuras líneas de trabajo

Hemos presentado un algoritmo para la construcción del unificador fuzzy más general de un par de fórmulas atómicas basado en la semántica de los términos. Para ello, hemos introducido la estructura semántica asociada con las variables y variables lingüísticas de un programa y hemos desarrollado una medida de la compatibilidad o proximidad de dos conjuntos fuzzy, basada en la interpretación de la función de pertenencia de un conjunto fuzzy como un grado de similitud entre los elementos del universo y su conjunto de prototipos. Además, hemos formalizado las definiciones de sustitución, unificador y unificador fuzzy más general proporcionando, finalmente, el grado de compatibilidad o proximidad de dos fórmulas atómicas vía su unificador fuzzy más general.

En cuanto a las futuras líneas de trabajo, debemos considerar la incorporación de funciones al algoritmo de unificació fuzzy presentado y la integración de dicho algoritmo al entorno de programación lógica basado en una familia de lógicas infinitamente valuadas [1].

Finalmente, algunas de las aplicaciones del sistema se sitúan en el campo de la comunicación en lenguaje natural con robots, en el área de las bases de datos deductivas, en sistemas de diagnóstico y en la implantación de agentes de diálogo en sistemas multiagente.

## Referencias


[1] T. Alsinet and F. Manyà. A declarative programming environment for infinitely-valued logics. In *Proceedings of IPMU'96 Conference*, pages 1205–1210, Granada, Spain, 1996.

[2] K.R. Apt. Logic programming. In Jan Van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, chapter 10. Elsevier, Amsterdam, The Netherlands, 1990.

[3] J.F. Baldwin. Evidential support logic programming. *Fuzzy Sets and Systems*, 24:1–26, 1987.

[4] J.F. Baldwin, T.P. Martin, and B.W. Pilsworth. *Fril: Fuzzy and Evidential Reasoning in Artificial Intelligence.* John Whiley & Sons Inc., 1995.

[5] M. Cayrol, H. Farreny, and H. Prade. Fuzzy pattern matching. *Kybernetes*, 11:103–116, 1982.

[6] D. Dubois, J. Lang, and H. Prade. POSLOG, an inference system based on possibilistic logic. In *Proceedings North American Fuzzy Information Processing Society Congress*, pages 177–180, 1990.

[7] D. Dubois, H. Prade, and S. Sandri. Possibilistic logic augmented with fuzzy unification. In *Proceedings of IPMU'96 Conference*, pages 1009–1014, Granada, Spain, 1996.

[8] G. Escalada-Imaz and F. Manyà. A linear interpreter for logic programming in multiple-valued propositional logic. In *Proceedings of IPMU'94 Conference*, pages 943–949, Paris, France, 1994.

[9] S. Gottwald. Defining distances of fuzzy sets through t-norms. In *Proceedings 1st EUFIT'93*, pages 351–357, Aachen, Germany, 1993.

[10] P. Hájek and L. Godo. Deductive systems of fuzzy logic (a tutorial). Technical Report IIIA 97-10, Institut d'Investigació en Intel.ligència Artificial, CSIC, Bellatera, Spain, 1997.

[11] M. Ishizuka and N. Kanai. Prolog-ELF incorporating fuzzy logic. In *Proceedings of the Ninth IJCAI*, pages 701–703, 1985.

[12] D. Li and G. Liu. *A Fuzzy Prolog Database System.* Research Studies Press and John Wiley and Sons, 1990.

[13] A. Martelli and U. Montanari. An efficient unification algorithm. *ACM Trans. on Programming Languages and Systems*, 4(2):258–282, 1982.

[14] T.P. Martin, J.F. Baldwin, and B.W. Pilsworth. The implementation of FPPROLOG: A fuzzy prolog interpreter. *Fuzzy Sets and Systems*, 23:119–129, 1987.

[15] M. Mukaidono, Z. Shen, and L. Ding. Fundamentals of fuzzy prolog. *International Journal of Aproximate Reasoning*, 3:179–193, 1989.

[16] L.G. Rios-Filho and S.A. Sandri. Contextual fuzzy unification. In *Proceedings 5th IFSA'95 Congress*, pages 81–84, São Paulo, Brazil, 1995.

[17] L. Valverde. On the structure of F-indistinguishability operators. *Fuzzy Sets and Systems*, 17:313–328, 1985.

[18] H.E. Virtanen. Fuzzy unification. In *Proceedings of IPMU'94 Conference*, pages 1147–1152, Paris, France, 1994.

[19] T.J. Weigert, J. Tsai, and X. Liu. Fuzzy operator logic and fuzzy resolution. *Journal of Automated Reasoning*, 10:59–78, 1993.

# UNA MEDIDA DE LA BONDAD DE LA INFERENCIA USANDO MEDIDAS DIFUSAS*

**M.C. Garrido**†
Dpto. de Informática y Sistemas
Universidad de Murcia
mgarrido@dif.um.es

**J.M. Cadenas**
Dpto. de Informática y Sistemas
Universidad de Murcia
jcadenas@dif.um.es

**A. Ruiz**
Dpto. de Informática y Sistemas
Universidad de Murcia
aruiz@dif.um.es

## Resumen

En este trabajo se presentan una serie de resultados que se pueden obtener a partir del estudio de los elementos de una decisión en una medida difusa, cuando trabajamos en dominios discretos.

Estos resultados giran en torno a medir con qué grado de confianza nos creemos, que los valores inferidos a partir de una regla de decisión en la medida difusa, son los mismos que los inferidos a partir de una regla de decisión aplicada a la función de probabilidad real que modela un sistema.

**Palabras clave:** Intervalos de probabilidad, medidas difusas, probabilidad imprecisa, regla de decisión, proceso de inferencia, intervalos de confianza.

## 1 INTRODUCCIÓN

Muchos fenómenos en los que se manifiesta la regularidad estadística, son susceptibles de ser descritos mediante un modelo probabilístico, es decir, que las variables que intervienen en el mismo pueden considerarse como variables aleatorias y ser observadas repetidas veces.

Estas variables aleatorias $z = (z_1, z_2, \ldots, z_n)$ siguen una función de probabilidad, $p(z)$, que dependerá de uno o de más parámetros.

Con frecuencia se suele modelar el comportamiento de un sistema mediante la función de probabilidad que hemos comentado anteriormente, $p(z)$.

A partir de $p(z)$ se pueden realizar distintos procesos de razonamiento, como por ejemplo, realizar inferencia.

*Trabajo parcialmente soportado por el proyecto CICYT TIC 95-1019.

†Autor para correspondencia.

Un proceso de inferencia, de una manera genérica, podemos decir que consiste en obtener los valores de ciertas variables desconocidas $y \subset z$, a partir del valor observado del resto de variables $x \subset z$, donde $x = z - y$.

De una forma más detallada, el proceso de inferencia, cuando trabajamos con variables discretas, será el siguiente:

Dada la observación $x = x_i$

1. Calculamos $p(y_j/x = x_i) = \frac{p(x_i, y_j)}{p(x_i)}$, $\forall y_j \in \Omega_y$, donde $\Omega_y$ es el conjunto de todos los posibles valores que podrá tomar el subconjunto de variables que forman $y$.
2. Aplicamos la siguiente regla de decisión:

   Elegir el $y_j$ con mayor $p(y_j/x = x_i)$. Es decir, el valor $y_j$ que tiene una mayor probabilidad a posteriori, dada la observación $x = x_i$.

La regla de decisión aplicada en este proceso, está formada por los siguientes elementos:

$$p(y_j/x = x_i), \quad \forall y_j \in \Omega_y$$

Es decir, si $|\Omega_y|$ es el cardinal de $\Omega_y$, la regla de decisión estará formada por $|\Omega_y|$ elementos de decisión.

Esta regla de decisión minimiza la probabilidad media de error, [4].

Sin embargo, en la mayoría de las ocasiones $p(z)$ es desconocida, y por tanto, para realizar el proceso de inferencia, se hace uso de otra información como puede ser:

- Una aproximación de $p(z)$, a partir de un conjunto de observaciones de los valores tomados por las variables que constituyen $z$, [4], [5].
- Mediante parejas de medidas que acotan una probabilidad, es decir, mediante una acotación de $p(z)$ que notamos por $\hat{p}(z)$, [9], [8].

• ···

En este trabajo nos centraremos en la segunda posibilidad.

Es conocido que las medidas difusas son formalismos matemáticos que permiten expresar la incertidumbre asociada al razonamiento humano y que pueden ser considerados como una generalización de las probabilidades al relajar la condición de aditividad.

Sin embargo, cuanto más general es el formalismo, más complejo es su tratamiento, por lo que se introducen nuevas condiciones que restringen algo más el campo pero que permiten desarrollar medidas difusas más manejables, [8].

En este trabajo no vamos a centrar en la medida difusa de Intervalos de Probabilidad (IP), que combina una expresividad razonable con una eficiente computación, [7]. En concreto, estudiaremos los elementos de decisión de una regla dentro de la medida difusa IP, para obtener a partir de ellos algunos resultados de interés.

En la sección 2 mostramos la forma de los elementos de decisión de una regla en la medida IP, así como los razonamientos que podemos realizar a partir de ellos.

En la sección 3 se introducen algunos casos especiales que se pueden presentar al estudiar las decisiones y qué podemos deducir en tales casos.

En la sección 4 presentamos un procedimiento que se puede seguir para realizar el estudio de dichos elementos, terminando con la presentación de algunos ejemplos en la sección 5.

## 2 ESTUDIO DE LOS ELEMENTOS DE DECISIÓN EN LA MEDIDA DIFUSA IP

De acuerdo a la medida difusa IP, [7], vamos a definir $\hat{p}(z)$ como un conjunto de intervalos de probabilidad propio

$$\hat{p}(z) = I(p(z))^{\alpha} = \{[l(z_j), u(z_j)],\ \forall z_j \in \Omega_z\}$$

donde $\Omega_z$ es el dominio del conjunto de variables $z$, que cumplen

$$0 \leq l(z_j) \leq p(z_j) \leq u(z_j) \leq 1,\ \forall z_j$$

con un grado de confianza $\alpha$, es decir, con una probabilidad de que $p(z_j) \in [l(z_j), u(z_j)], \forall z_j$ igual a un cierto valor $\alpha$. Suponemos que este conjunto de intervalos de probabilidad se obtiene a partir de un conjunto de observaciones, suficientemente representativo, obtenido del sistema en estudio, [1], [2], [3].

Cuando trabajamos con $\hat{p}(z)$, el proceso de inferencia es el siguiente:

Dado la observación $x = x_i$

1. Calculamos $[l(y_j/x = x_i), u(y_j/x = x_i)],\ \forall y_j \in \Omega_y$.
2. Aplicamos una de las dos reglas de decisión siguientes:
   - **Regla de decisión máximo pesimista**: con la cual se elige el $y_j$ con mayor $l(y_j/x = x_i)$.
   - **Regla de decisión máximo optimista**: con la cual se elige el $y_j$ con mayor $u(y_j/x = x_i)$.

Por lo tanto, podemos tomar como elementos que constituyen las decisiones, los intervalos $[l(y_j/x = x_i), u(y_j/x = x_i)]$, [6], [7], donde:

$$l(y_j/x = x_i) \leq p(y_j/x = x_i) \leq u(y_j/x = x_i),\ \forall y_j$$

**Definición 1** *Dado un conjunto de intervalos de probabilidad* $I(p(z))^{\alpha} = \{[l(z_j), u(z_j)],\ \forall z_j \in \Omega_z\}$, *donde* $|\Omega_z| = m$, *se define su grado de solapamiento,* $S(I(p(z))^{\alpha})$ *como:*

$$S(I(p(z))^{\alpha}) = \sum_{k=1}^{m} \sum_{h=k+1}^{m} f(k,h)$$

*donde*

$$f(k,h) = \begin{cases} u(z_k) - l(z_h) & si\ u(z_k) > l(z_h) \geq l(z_k) \\ u(z_h) - l(z_k) & si\ u(z_h) > l(z_k) > l(z_h) \\ 0 & en\ otro\ caso \end{cases}$$

■

Veamos ilustrativamente el concepto de grado de solapamiento en la figura 1.

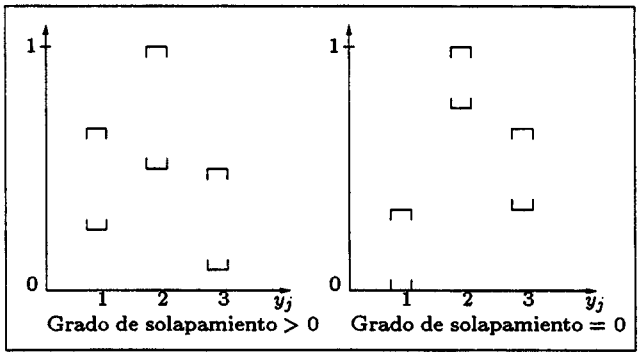



Figura 1: Grado de Solapamiento.

**Proposición 1** *Dado el conjunto de intervalos de probabilidad* $I(p(y/x = x_i))^\alpha = \{[l(y_j/x = x_i), u(y_j/x = x_i)], \forall y_j \in \Omega_y\}$, *donde* $|\Omega_y| = m$, *si el grado de solapamiento,* $S(I(p(y/x = x_i))^\alpha)$, *es cero, se puede asegurar con un grado de confianza* $\alpha$, *que la decisión tomada con las reglas máximo pesimista y máximo optimista, es la misma que la tomada con la regla de decisión de mínima probabilidad media de error aplicada a la función de probabilidad real* $p(z)$.

Demostración

El conjunto $I(p(y/x = x_i))^\alpha$ está formado por todos los elementos de una decisión concreta en la medida IP donde $x_i$ es un valor fijo. Por cuestiones de comodidad vamos a notar $[l(y_j/x = x_i), u(y_j/x = x_i)]$ como $[l(y_j), u(y_j)]$.

Entonces, si $S(I(p(y/x = x_i))^\alpha) = 0 \Rightarrow f(k,h) = 0,\ k = 1 \ldots m,\ h = k+1 \ldots m$ por la definición 1 y por tanto:

$$\begin{array}{l} l(y_h) > u(y_k) > l(y_k) \\ \acute{o} \qquad\qquad\qquad\qquad \forall k,h \\ l(y_k) > u(y_h) > l(y_h) \end{array} \qquad (1)$$

Supongamos que la decisión tomada por la regla de mínima probabilidad de error aplicada a $p(y/x = x_i)$ es $y_k$, ya que éste es el valor de $y$ con mayor $p(y/x = x_i)$. Por lo tanto, el intervalo asociado que acota $p(y_k/x = x_i)$, será $[l(y_k), u(y_k)]$ con

$$l(y_k) \leq p(y_k/x = x_i) \leq u(y_k)$$

Por (1) tenemos que todos los intervalos están desolapados y, por lo tanto, la decisión máximo pesimista o máximo optimista es también $y_k$, con grado de confianza $\alpha$.

■

Si el grado de solapamiento es 0, podemos asegurar con un determinado grado de confianza, que la inferencia realizada con la medida difusa IP, es tan buena o correcta como la inferencia realizada con la función de probabilidad real si ésta fuera conocida.

## 3 CASOS ESPECIALES DE SOLAPAMIENTO

Según lo comentado en la sección anterior, el valor del grado de solapamiento que más nos interesa, es decir, a partir del cual podemos hacer afirmaciones, es $S(I(p(y/x = x_i))^\alpha) = 0$.

Sin embargo, hay determinadas situaciones en las que se producen solapamientos de intervalos pero éstos no repercuten en los resultados vistos en la proposición 1.

Mostramos algunos ejemplos de estos solapamientos de forma gráfica en la figura 2.

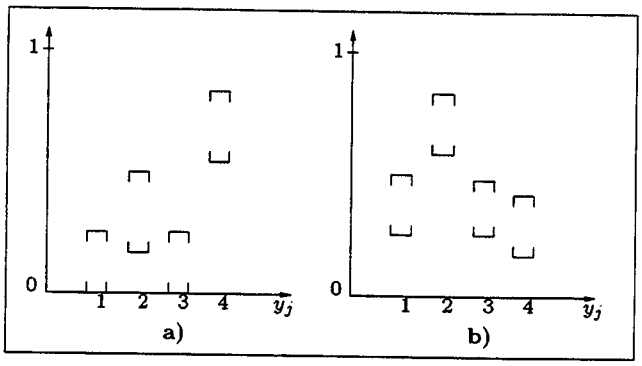



Figura 2: Casos especiales de Solapamiento.

En las dos situaciones de la figura 2, el solapamiento se produce en intervalos que no serán seleccionados por ninguna de las reglas de decisión con las que estamos trabajando. En la figura 2,a) el valor inferido será $y_j = 4$ y su intervalo $[l(4/x = x_i), u(4/x = x_i)]$ está totalmente desolapado. Lo mismo ocurre en la figura 2,b) para $y_j = 2$.

Por lo tanto, si:

$$S(I(p(y/x = x_i))^\alpha) > 0$$

y se cumple que el $y_j$ que se infiere hace que:

$$l(y_j/x = x_i) \geq u(y_h/x = x_i), \quad \forall y_h \neq y_j$$

podemos asegurar con un grado de confianza $\alpha$, que la decisión tomada con las reglas máximo pesimista y máximo optimista, es la misma que la tomada con la regla de decisión de mínima probabilidad media de error aplicada a la función de probabilidad real $p(z)$.

## 4 PROCEDIMIENTO PARA CALCULAR EL GRADO DE SOLAPAMIENTO

Vamos a ver un procedimiento para calcular el grado de solapamiento total (ST) de todas las posibles decisiones de inferencia de un sistema determinado.

$$ST = \sum_{i=1}^{t} S(I(p(y/x = x_i))^\alpha), \quad x_i \in \Omega_x$$

donde $t = |\Omega_x|$.

Para ello disponemos de un conjunto de observaciones, $Z$, de tamaño $N \geq 30$, que constituyen un conjunto

suficientemente representativo del comportamiento del sistema[1].

Inicialmente formamos una base de ejemplos, $BD$, con 30 datos del conjunto anterior elegidos aleatoriamente y calculamos el valor de $ST$ que se obtiene con esta base de ejemplos. En pasos sucesivos iremos repitiendo el mismo proceso, pero añadiendo nuevos datos, elegidos aleatoriamente de $Z$, a $BD$. Antes de pasar a describir el procedimiento, vamos a tratar una serie de cuestiones a tener en cuenta:

- La condición de parada del procedimiento se dará cuando el grado de solapamiento total, ST, se estabilice ($ST$ toma el mismo valor u oscila en torno al mismo valor en varias iteraciones) o $BD$ contenga todos los ejemplos de $Z$. La estabilización de $ST$ se puede producir en 0 ó en un valor distinto de 0. En este último supuesto, habrá que detectar los solapamientos especiales y suprimirlos de ST.

- Al ir incrementando el tamaño de la base de ejemplos $BD$, hay que tener en cuenta que los intervalos de confianza se hacen más pequeños, puesto que hay menos imprecisión (son decrecientes con el número de ejemplos).

  Esto nos puede llevar a una disminución engañosa del grado de solapamiento, en la que no disminuye el solapamiento de los intervalos, sino su tamaño.

  Para evitar este problema, vamos a normalizar el solapamiento de cada decisión de la siguiente manera:

$$\frac{S(I(p(y/x=x_i))^\alpha)}{S_{max}(I(p(y/x=x_i))^\alpha)}$$

  donde $S_{max}(I(p(y/x = x_i))^\alpha)$ se calcula a partir de los intervalos $[l(y_j/x = x_i), u(y_j/x = x_i)]$, $\forall y_j$, transformados de la siguiente forma:

$$[l(y_j/x=x_i), u(y_j/x=x_i)] =$$

$$[l(y_j/x=x_i) + (u(y_{max}/x=x_i) - u(y_j/x=x_i)),$$
$$u(y_{max}/x=x_i)]$$

  donde $y_{max}$ es el $y_j$ que tiene mayor valor $u(y_j/x = x_i)$.

  De una forma gráfica, la transformación de los intervalos $I(p(y/x = x_i))^\alpha$ para obtener $S_{max}$ se muestra en la figura 3.

Una vez tratadas estas cuestiones pasamos a ver el procedimiento de cálculo que se muestra en la tabla 1.

[1]La estimación de los intervalos de confianza se realiza mediante una aproximación Normal. Este método de estimación requiere que el número de muestras no sea muy pequeño. De ahí que tomemos inicialmente $N \geq 30$.

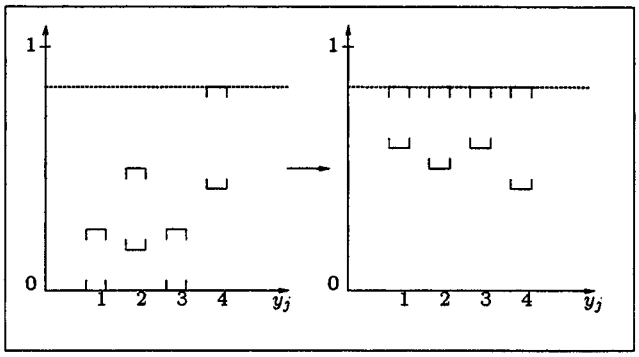



Figura 3: Transformación de intervalos.

Tabla 1: Procedimiento de cálculo.

1. $\alpha$ = nivel de confianza requerido;
2. Tamaño$_{BD}$=15;
3. Incremento$_{BD}$=15;
4. Generar una base de datos, $BD$, con Tamaño$_{BD}$ ejemplos;
5. Repetir
   a. Tamaño$_{BD}$=Tamaño$_{BD}$ + Incremento$_{BD}$;
   b. Añadir a la base de datos $BD$, Incremento$_{BD}$ datos;
   c. Obtener $\hat{p}(z)$ a partir de $BD$;
   d. Para cada $x_i$

$$ST = ST + \frac{S(I(p(y/x=x_i))^\alpha)}{S_{max}(I(p(y/x=x_i))^\alpha)}$$

   hasta que $ST$ se estabilice o Tamaño$_{BD} \geq N$;
6. Eliminar solapamientos especiales.

Con el procedimiento de la tabla 1, las afirmaciones que podremos realizar son las siguientes:

1. Si la parada del procedimiento se produce por estabilización de $ST$, podemos concluir que:
   (a) Si $ST = 0$, aseguramos con un nivel de confianza $\alpha$, que a partir del conjunto de ejemplos $BD$, los valores inferidos por $\hat{p}(z)$ son los mismos que se inferirían si conocieramos la función de probabilidad real del sistema, $p(z)$.
   (b) Si $ST \neq 0$, habrá que considerar cada decisión por separado, estudiando el valor $S(I(p(y/x = x_i))^\alpha)$:
      i. Si $S(I(p(y/x = x_i))^\alpha) = 0$, la conclusión es la misma que en (1a), pero particularizada para $x = x_i$.

ii. Si $S(I(p(y/x = x_i))^\alpha) > 0$, no podemos concluir nada acerca de esta regla de decisión, aunque aumentemos el número de datos de $BD$.

2. Si la parada del procedimiento se produce porque $\text{Tamaño}_{BD} \geq N$, consideraremos el valor $S(I(p(y/x = x_i))^\alpha)$ para cada decisión $x = x_i$ por separado:

   En los casos en los que $S(I(p(y/x = x_i))^\alpha)$ se estabiliza en un valor, el razonamiento es el mismo que en los apartados (1bi) y (1bii). En caso contrario, no podemos realizar ninguna afirmación acerca de esa decisión a partir del conjunto $Z$, ya que serían necesarios más datos.

Al proceder de esta manera, hemos visto que puede ocurrir que $ST$ se estabilice cuando $BD$ contiene menos ejemplos que $Z$. Esto nos indica que el subconjunto $BD$, es igual de representativo que $Z$, pero de menor tamaño.

# 5 EJEMPLOS DE DECISIONES CON DISTINTOS GRADOS DE SOLAPAMIENTO

Supongamos un sistema en estudio, cuyo comportamiento está descrito por dos variables aleatorias $(x_1, x_2)$, con $\Omega_{x_1} = \{a, b, c\}$ y $\Omega_{x_2} = \{1, 2, 3\}$ y que la función de probabilidad que siguen estas variables es la mostrada en la tabla 2.

Tabla 2: Probabilidad real.

| $(x_1, x_2)$ | $p(x_1, x_2)$ |
|---|---|
| $(a, 1)$ | 0 |
| $(a, 2)$ | 0 |
| $(a, 3)$ | 0.1 |
| $(b, 1)$ | 0.2 |
| $(b, 2)$ | 0.3 |
| $(b, 3)$ | 0.1 |
| $(c, 1)$ | 0.1 |
| $(c, 2)$ | 0 |
| $(c, 3)$ | 0.2 |

A partir de un conjunto de 30 y 500 observaciones de dichas variables, obtenemos la acotación $\hat{p}(x_1, x_2)$, con un grado $\alpha = 95\%$, que mostramos en la tabla 3.

A partir de estos datos, pasamos a estudiar la decisión $x_1 = c$, cuyos resultados mostramos en las tablas 4 y 5.

El símbolo * en las tablas indica el resultado inferido según la probabilidad real y las probabilidades imprecisas correspondientes a cada columna.

Tabla 3: Intervalos de probabilidad.

| | N=30 | N=500 |
|---|---|---|
| $(x_1, x_2)$ | $[l(x_1, x_2), u(x_1, x_2)]$ | $[l(x_1, x_2), u(x_1, x_2)]$ |
| $(a, 1)$ | [0,0.141324] | [0,0.00950496] |
| $(a, 2)$ | [0,0.141324] | [0,0.00950496] |
| $(a, 3)$ | [0.0017424,0.19053] | [0.065339,0.117194] |
| $(b, 1)$ | [0.0840465,0.39131] | [0.1831,0.257318] |
| $(b, 2)$ | [0.205426,0.560923] | [0.27391,0.356996] |
| $(b, 3)$ | [0.026188,0.276767] | [0.074048,0.128315] |
| $(c, 1)$ | [0.0630346,0.35451] | [0.0775552,0.13274] |
| $(c, 2)$ | [0,0.141324] | [0,0.00950496] |
| $(c, 3)$ | [0.0435963,0.31648] | [0.147884,0.21714] |

Tabla 4: Con 30 ejemplos.

| | | N=30 |
|---|---|---|
| $x_2$ | $p(x_2/x_1 = c)$ | $[l(x_2/x_1 = c), u(x_2/x_1 = c)]$ |
| 1 | 0.33333 | * [0.121037,0.890492] |
| 2 | 0 | [0,0.569959] |
| 3 | * 0.66667 | [0.0808185,0.833884] |
| | | S=0.87212 |

En la tabla 5, el solapamiento S=0, indica que con un grado de confianza del 95%, la decisión tomada es la misma que la real.

Con la decisión $x_2 = 3$, como podemos ver en la tabla 6, se presenta un caso de solapamiento especial, que permite aplicar la Proposición 1 aunque haya solapamiento de intervalos.

# 6 CONCLUSIONES

En este trabajo hemos visto una medida del grado de seguridad con el que podemos afirmar que las decisiones tomadas para inferir valores, cuando trabajamos con una acotación de la probabilidad obtenida a partir de un conjunto de ejemplos suficientemente representativo del comportamiento de un sistema, son iguales a las que tomaría la probabilidad real que describe dicho sistema, si ésta fuera conocida.

También hemos presentado un procedimiento para realizar el estudio de las decisiones, con el que hemos visto que se puede detectar si, dado un conjunto inicial de ejemplos, $Z$, podemos reducirlo y obtener los mismos resultados que al trabajar con el conjunto completo.

Hay que indicar que, si bien este último resultado no tendría aplicación en inferencia en dominios discretos, sí que lo tendría en dominios continuos.

Tabla 5: Con 500 ejemplos.

| $x_2$ | $p(x_2/x_1 = c)$ | N=500 $[l(x_2/x_1 = c), u(x_2/x_1 = c)]$ |
|---|---|---|
| 1 | 0.33333 | [0.254948,0.473018] |
| 2 | 0 | [0,0.0404562] |
| 3 | * 0.66667 | * [0.509718,0.736829] |
| | | S=0.0 |

Tabla 6: Decisión $x_2 = 3$.

| $x_1$ | $p(x_1/x_2 = 3)$ | N=500 $[l(x_1/x_2 = 3), u(x_1/x_2 = 3)]$ |
|---|---|---|
| $a$ | 0.25 | [0.159055,0.345576] |
| $b$ | 0.25 | [0.181321,0.375698] |
| $c$ | * 0.5 | * [0.37592,0.609042] |
| | | S=0.289479 |

En dominios discretos, la información proporcionada por los ejemplos queda representada mediante la distribución de probabilidad muestral.

En dominios continuos, hay técnicas de inferencia que no se basan en modelos matemáticos para simular el comportamiento del sistema, sino que trabajan directamente con los datos (por ejemplo, la técnica de $k$ de $N$ vecinos más cercanos, Parzen windows [4], [5]). En estos casos es interesante reducir el conjunto de datos con los que hay que trabajar, manteniendo los mismos resultados.

La extensión del estudio realizado en este trabajo a dominios continuos, será objeto de futuros trabajos.

## Referencias


[1] C.R. Blyth and H.A. Still (1983), Binomial confidence intervals.*Journal of the American Statistical Association 78 (381)* Pág. 108-116.

[2] G.F. Clopper and E.S. Pearson (1934), The use of confidence or fiducial limits ilustrated in the case of the binomial. *Biometrika 26* Pág. 404-413.

[3] E.L. Crow (1956), Confidence intervals for a proportion. *Biometrika 43* Pág. 423-435.

[4] R.O. Duda, P.E. Hart (1973), Pattern classification and scene analysis. *Wiley, New York.*

[5] K. Fukunaga (1990), Introduction to Statistical Pattern Recognition. *Academic Press.*

[6] M.C. Garrido, J.M. Cadenas, A. Ruiz (1997), Inferencia con probabilidades imprecisas en dominios discretos. *Technical Report TR DIS 2/97. Dpto. Informática y Sistemas. Universidad de Murcia.*

[7] J.F. Huete (1995), Aprendizaje de redes de creencia mediante detección de independencias: modelos no probabilísticos. *Tesis doctoral. Universidad de Granada.*

[8] J.F. Verdegay (1991), Tratamiento de la incertidumbre mediante medidas difusas. *Algunos aspectos del tratamiento de la información en Inteligencia Artificial. Universidad de Granada.* Pág. 155-172.

[9] P. Walley (1991), Statistical Reasoning with Imprecise Probabilities. *Chapman and Hall.*

# TEST DEL ALGORITMO FUZZY-MINIMALS CON DATOS REALES

**Antonio Flores-Sintas**[1]
Dpto. de Nuevas Tecnologías
Caja de Ahorros de Murcia
Gran Vía,23
30005-Murcia
e-mail: aflores@cajamurcia.es

**José M. Cadenas**
Dpto. de Infor. y Sistemas
Facultad de Informática
Universidad de Murcia
30071-Espinardo,Murcia
e-mail: jcadenas@dif.um.es

**Fernando Martin**
Dpto. de Infor. y Sistemas
Facultad de Informática
Universidad de Murcia
30071-Espinardo,Murcia
e-mail: fmartin@dif.um.es

**RESUMEN**

Una función de pertenencia y una función objetivo deducidas de las propiedades geométricas asociadas a la métrica definida por la matriz de covarianzas de una muestra han sido recientemente propuestas. Aplicando los resultados teóricos usando la distancia Euclídea, se ha obtenido un nuevo algoritmo denominado Fuzzy-Minimals que detecta los posibles prototipos de los grupos de una muestra. En este trabajo testeamos dicho algoritmo utilizando datos tomados del entorno empresarial.

**Palabras Clave:** funciones de pertenencia, análisis cluster.

## 1. INTRODUCCION.

Recientemente ([2],[3],[4]), hemos realizado un estudio sobre las propiedades geométricas locales asociadas a la métrica derivada de la matriz de covarianzas. La métrica Euclidiana es un caso particular de esta métrica. Como consecuencia del estudio, aparece una función objetivo del tipo suma de los cuadrados de los errores que está deducida de la función acción asociada a la perturbación producida por la muestra y medida por la métrica, y una función de pertenencia a los grupos independiente de la existencia de otros grupos. La probabilidad de pertenencia se calcula en función del grado de pertenencia. Además, utilizando la distancia Euclídea, la función objetivo permite ser tratada sin conocimiento previo del número de grupos, dando origen a un nuevo algoritmo que hemos denominado algoritmo Fuzzy Minimals (a/FM) [3]. Por otra parte, el estudio realizado nos ha llevado también a modificar el algoritmo Fuzzy C-Means (a/FCM) [4].

La idea básica presentada en [2] y [3] es que la métrica en cualquier punto del dominio de discurso está definida por la matriz de covarianzas en dicho punto. Sea $X=\{x_1,x_2,...,x_n\}\subset \mathbf{R}^F$, una muestra $F$-dimensional de $n$ patrones. La función objetivo deducida ([2] y [3]) para un grupo (la muestra) es la siguiente:

$$J_{(m)}(Mh)=\sum_{x\in X}\mu_{xm}d^2_{xm}\sqrt{g_{(x)}} \qquad (1)$$

siendo $d^2_{xm}$ la distancia de Mahalanobis ([1] y [5]) entre el patrón $x$ y la media $m$; $g_{(x)}$ el determinante de la inversa de la matriz de covarianzas en $x$, y

$$\mu_{xm}=1\Big/\left(1+d^2_{xm}\right) \qquad (2)$$

la función de pertenencia de $x$ respecto de $m$. $g_{(x)}$ está relacionado con el determinante de la matriz de covarianzas en la media de la siguiente forma:

$$g_{(x)}=g_{(m)}\Big/\left(1+d^2_{xm}\right) \qquad (3)$$

por lo que (2) se puede escribir como sigue:

$$J_{(m)}(Mh)=\sum_{x\in X}\mu^{3/2}_{xm}d^2_{xm}\sqrt{g_{(m)}} \qquad (4)$$

Si utilizamos distancia Euclídea, $g_{(x)}=1$, y la expresión (1) se convierte en:

[1] Autor para Correspondencia

$$J_{(v)}(Ec) = \sum_{x \in X} \mu_{xv} d_{xv}^2 \qquad (5)$$

La diferencia entre (1) y (5) es que (1) sólo está definida para la media del grupo, mientras que (5) está definida para cualquier punto $v \in \mathbf{R}^F$.

$J_{(v)}(Ec)$ la podemos utilizar para determinar los posibles prototipos de los grupos en que podemos dividir una muestra. El criterio para obtenerlos es el mismo que el utilizado en el (a/FCM), es decir, los prototipos serán los que hagan mínima a $J_{(v)}(Ec)$, y el número de prototipos es el número de mínimos diferentes de $J_{(v)}(Ec)$. En [2] y [3] hemos obtenido:

$$v = \sum_{x \in X} \mu_{xv}^2 x \Big/ \sum_{x \in X} \mu_{xv}^2 \qquad (6)$$

Sin embargo, hay una diferencia cualitativa importante entre la distancia Euclídea y la distancia de Mahalanobis que debemos tener en cuenta. Mientras que la distancia de Mahalanobis está normalizada, la distancia Euclídea no lo está. Esto quiere decir que si realizamos la transformación $x^i \rightarrow rx^i$, siendo $x^i$ las características del patrón $x$, las distancias Euclídeas entre los patrones serán diferentes y por consiguiente los mínimos de $J_{(v)}(Ec)$ también serán diferentes, pero nosotros esperamos que el resultado del proceso de clustering sea el mismo. Por consiguiente debemos establecer un criterio para normalizar la distancia Euclídea. El criterio que hemos propuesto en [2] y [3] es que la densidad de la muestra sea la unidad. De esta manera, el factor $r$ de la transformación anterior está determinado por la siguiente expresión:

$$\frac{1}{nr^F} \sum_{y \in X} \sqrt{g_{(y)}} \sum_{x \in X} \mu_{xy} = 1 \qquad (7)$$

$\sum_{x \in X} \mu_{xy}$ representa la "masa" asociada a un patrón $y \in X$, y $\sqrt{g_{(y)}}/r^F$ es la inversa del volumen determinado por la matriz de covarianzas en $y$, después de la transformación $x^i \rightarrow rx^i$. De (7), $r$ se puede determinar utilizando cualquier algoritmo que resuelva ecuaciones no lineales.

La función de pertenencia se transformará también, quedando de la siguiente forma:

$$\mu_{xv}(FM) = 1/\left(1 + r^2 d_{xv}^2\right) \qquad (8)$$

las letras FM entre paréntesis indican que es la función de pertenencia utilizada en el algoritmo Fuzzy Minimals. Los antecedentes del algoritmo que se han resumido en esta introducción están detallados en [2] y [3]. El algoritmo es el siguiente:

FM-1.Fijar $\varepsilon_1, \varepsilon_2 > 0$ constantes pequeñas. Inicializar $V = \varnothing \subset \mathbf{R}^F$. Calcular $r$ utilizando (7).

FM-2. Para $k = 1$ hasta $n$.

a. $v_{(0)} = x_k, t = 0, E_{(0)} = 1$

b. **While** $E_{(t)} \geq \varepsilon_1$

Sumar1 á $t$. Calcular $\mu_{xv}$ usando (8) y $v_{(t-1)}$. Calcular $v_{(t)}$ usando (6), y

$$E_{(t)} = \sum_{\alpha=1}^{F} \left(v_{(t)}^{\alpha} - v_{(t-1)}^{\alpha}\right)$$

c. **If** $\sum_{\alpha=1}^{F} \left(v^{\alpha} - w^{\alpha}\right) > \varepsilon_2, \forall w \in V,$

**then** $V \equiv V + \{v\}$

El algoritmo Fuzzy Minimals encuentra, mediante iteraciones de Picard, el mínimo de la función $J_{(v)}(Ec)$ más cercano a cada patrón, identificando los mínimos diferentes. Este algoritmo, además de que no necesita conocer a priori el número de clusters, presenta la ventaja de que se puede ejecutar en paralelo. En el algoritmo se han incluído dos constantes que controlan su flujo. La primera, $\varepsilon_1$, controla el grado de error que cometemos al calcular los mínimos. La segunda, $\varepsilon_2$, sirve para diferenciar los posibles mínimos, que serán los prototipos de los grupos de la partición difusa.

Una vez determinados los prototipos necesitamos conocer las probabilidades de pertenencia. Usualmente se asigna cada patrón al grupo con el que tenga mayor probabilidad de pertenencia.

## 2. DENSIDAD Y PROBABILIDAD.

Cuando utilizamos la métrica derivada de la matriz de covarianzas, $\sqrt{g_{(x)}}$ es un factor que afecta al elemento de volumen del espacio de las características en $x$, de tal manera que dicho elemento de volumen será $\sqrt{g_{(x)}} d\Omega$, siendo $d\Omega$ el elemento de volumen Euclídeo. Haciendo $\rho_{xm} = \mu_{xm} \sqrt{g_{(x)}}$, (1) se convierte en:

$J_{(m)}(Mh) = \sum_{x \in X} \rho_{xm} d_{xm}^2$, que tiene la forma de un momento de inercia por unidad de volumen, siendo $\rho_{xm}$ la densidad en $x$, lo que viene sugerido por el hecho de que $\sqrt{g_{(x)}} d\Omega$ es el elemento de volumen en $x$. Cuando $x$ coincide con la media $\rho_{mm} = \sqrt{g_{(m)}}$, y además, teniendo en cuenta (3), $\rho_{xm} = \mu_{xm}^{3/2} \sqrt{g_{(m)}}$, que quiere decir que $\mu_{xm}^{3/2} = \rho_{xm} / \rho_{mm}$ es la densidad en $x$ relativa a la densidad en la media.

Si en vez de utilizar la distancia de Mahalanobis hubiésemos utilizado la distancia Euclídea, entonces $\sqrt{g_{(x)}} = 1$, y $\rho_{xv} = \mu_{xv}$, y en consecuencia $\mu_{xv}$ es también la densidad en $x$ relativa a la densidad en $v$, pero suponiendo que la densidad en $v$ es la unidad, lo que justifica el valor del segundo miembro en (7). Si conociésemos la densidad en $v$ bastaría con multiplicar por $\mu_{xv}$ para determinar la densidad en $x$. Veamos como podemos calcular esa densidad:

Rodeemos cada patrón $x$ por un volumen $\Omega_x$, de tal manera que dentro de cada volumen $\Omega_x$ exista un solo patrón, y el volumen total $\Omega$ de la muestra sea $\Omega = \sum_{x \in X} \Omega_x$. La densidad en cada patrón será: $\rho_x = 1/\Omega_x$. Puesto que $\mu_{xv} = \mu_{vx}$, la densidad en $v$ también será proporcional a $\mu_{xv}$. En consecuencia: $\rho_v = \mu_{xv} / \Omega_x$. Como esto se cumple para cualquier $x$, también se cumplirá que: $\rho_v = \sum_{x \in X} \mu_{xv} \Big/ \sum_{x \in X} \Omega_x = \sum_{x \in X} \mu_{xv} / \Omega$. Es decir, la densidad en cualquier punto $v$ del dominio de discurso, utilizando distancia Euclídea, se puede calcular suponiendo que la masa total en $v$ es la suma de los grados de pertenencia de todos los patrones respecto de $v$. Esto quiere decir que la "masa" de un patrón situado en $x$ "visto" desde $v$ es $\mu_{xv}$, y por consiguiente (5) representa también un momento de inercia. En (7) hemos utilizado $\sum_{x \in X} \mu_{xy}$ para calcular la densidad en $y \in X$. Hemos supuesto en (7) que el volumen asociado a esa "masa" está determinado por la matriz de covarianzas en $y$. También podríamos haber asociado el mismo volumen a todos los patrones de la muestra, en este caso el determinado por la matriz de covarianzas en la media, sin embargo medidas experimentales nos han llevado a optar por la primera opción.

Las probabilidades de pertenencia de cada patrón, una vez determinados los prototipos, serán proporcionales a las densidades en cada prototipo y a las funciones de pertenencia respecto de cada prototipo. Suponiendo que se han detectado $c$ prototipos $\{v_1, v_2, \ldots, v_c\}$, serán:

$$u_{xk}(FM) = \frac{\mu_{xk}(FM) \sqrt{g_{(k)}} \sum_{y \in X} \mu_{yk}(FM)}{\sum_{j=1}^{c} \mu_{xj}(FM) \sqrt{g_{(j)}} \sum_{y \in X} \mu_{yj}(FM)} \qquad (9)$$

donde $\mu_{xk}(FM)$ está dada por (8) sustituyendo $v$ por $v_k$, y $\sqrt{g_{(k)}}$ es el determinante de la matriz de covarianzas de la muestra en el prototipo $k$. Puede ocurrir que un patrón tenga un grado de pertenencia máximo en un grupo y que tenga una probabilidad máxima en otro grupo.

El uso de la distancia Euclídea implica que el espacio de las características es homogéneo e isótropo [1]. Si suponemos que existe más de un grupo para deducir los prototipos, estamos rompiendo la homogeneidad e isotropía de dicho espacio. En consecuencia, es preferible utilizar el (a/FM) cuando asumamos que la muestra puede ser tratada con la distancia Euclídea. Con la distancia de Mahalanobis utilizaremos el (a/FCM) ([2] y [4]).

## 3. DESFUZZIFICACION.

En general, se asume que un patrón pertenece al grupo con el que tienen una mayor probabilidad de pertenencia, pero la probabilidad de pertenencia resultante es respecto de un grupo de la partición difusa. Existe una doble incertidumbre: una es debida al grado de pertenencia al grupo representado por cada prototipo, y la otra es la incertidumbre de los prototipos que están determinados en función de los grados de pertenencia. Esta doble incertidumbre se debería de evitar. Por otra parte, si la partición fuese hard diríamos que un patrón pertenece o no pertenece a un grupo ( probabilidad de pertenencia 0 ó 1), pero esta es una solución extrema que no encaja con la intuición. Coloquialmente decimos que, en una partición hard, un patrón pertenece a un grupo y que además tiene una probabilidad de pertenencia tanto al grupo al que pertenece como a los otros grupos no necesariamente 0 ó 1, aunque se espera que la probabilidad mayor se corresponda con el grupo al que pertenece. El problema es expresarlo matemáticamente.

Como se ha visto, una cosa es la métrica y otra bien distinta es el grado o la probabilidad de pertenencia.

Trasladándolo a términos hard, una cosa es qué patrones participan en la formación del prototipo y en la matriz de covarianzas de un grupo, y otra distinta es qué probabilidad de pertenencia tiene cada patrón en cada grupo. Dicho de otra manera, una partición hard es aquélla en la que sabemos qué patrones forman los grupos que son los que determinan la métrica de cada grupo. Esto no es incompatible con el grado o la probabilidad de pertenencia sino complementario. En consecuencia, una $c$-partición hard de $X$ está determinada por una matriz $\eta_{xk} = \{0,1\},\ x \in X, 1 \le k \le c,\ \sum_{x \in X} \eta_{xk} > 1$. La media y la matriz de covarianzas en la media de cada grupo son:

$$m_k = \frac{\sum_{x \in X} \eta_{xk} x}{\sum_{x \in X} \eta_{xk}} \qquad (10a)$$

$$g_k^{\alpha\beta}(h) = \frac{\sum_{x \in X} \eta_{xk} \left(x^{\alpha} - m_k^{\alpha}\right)\left(x^{\beta} - m_k^{\beta}\right)}{\sum_{x \in X} \eta_{xk}} \qquad (10b)$$

$\alpha$ y $\beta$ variando para todas las características de la muestra. Las expresiones anteriores determinan la métrica de cada grupo. En (10b) el subíndice $k$ está sin paréntesis para indicar que es la matriz de covarianzas del grupo $k$, mientras que con paréntesis indicamos que es la matriz de covarianzas de la muestra, o su determinante (sin superíndices), en el prototipo $k$.

Cuando obtenemos la partición difusa mediante el (a/FM), esperamos obtener una buena partición. Las probabilidades de pertenencia dadas por (9) son densidades relativas que sumadas dan la unidad. Parece natural definir la partición hard deducida de la partición difusa por aquélla formada por los patrones con mayor probabilidad, es decir:

$$\eta_{xk} = \begin{cases} 1 \Rightarrow u_{xk} = max\{u_{xj} | 1 \le j \le c\} \\ 0 \Rightarrow u_{xk} \ne max\{u_{xj} | 1 \le j \le c\} \end{cases} \qquad (11)$$

La expresión anterior es una condición hard que define la partición hard deducida de la partición difusa obtenida. Se podría definir una condición difusa para tener en cuenta aquéllos patrones que tienen una probabilidad de pertenencia a un grupo no muy diferente de la probabilidad de pertenencia máxima, pero en este trabajo utilizaremos (11). La métrica de cada grupo estará determinada por las expresiones (10), y las probabilidades de pertenencia a los grupos de la partición hard se deducen de dichas métricas como veremos a continuación.

Anteriormente hemos visto que el grado de pertenencia de un patrón a un prototipo representaba la densidad relativa del patrón respecto del prototipo. Vamos a utilizar este significado del grado de pertenencia para calcular las densidades respecto de cada prototipo y en consecuencia las probabilidades de pertenencia en la partición hard. En cada grupo podemos realizar una normalización de la distancia Euclídea similar a la realizada cuando se trataba toda la muestra, imponiendo la condición de que la densidad en la media de cada grupo sea la unidad, de la siguiente forma:

$$\frac{1}{r_k^F} \sqrt{g_k(h)} \sum_{x \in X} \mu_{xk}^{(h)} \eta_{xk} = 1 \qquad (12a)$$

$$\mu_{xk}^{(h)} = \frac{1}{1 + r_k^2 d_{xk}^2} \qquad (12b)$$

La expresión (12a) calcula, para cada grupo, el valor del factor por el que debemos multiplicar a la distancia Euclídea para que la densidad en la media de cada grupo sea la unidad, y la expresión (12b) calcula el grado de pertenencia del patrón $x \in X$ respecto de la media del grupo $k$. Con la letra $h$ indicamos que estamos tratando la partición hard. En estas condiciones, $\mu_{xk}^{(h)}$ representa la densidad del grupo $k$ en $x \in X$. Por consiguiente, las probabilidades serán las siguientes:

$$u_{xk}^{(h)} = \frac{\mu_{xk}^{(h)}}{\sum_{j=1}^{c} \mu_{xj}^{(h)}} \qquad (13)$$

Si la matriz de covarianzas del grupo $k$ es singular haremos $u_{xk}^{(h)} = \eta_{xk}$.

Igual que en la partición difusa, diremos que en la partición hard un patrón pertenece al grupo con el que tenga una mayor probabilidad de pertenencia. Puede suceder que un patrón forme parte de un grupo, en el sentido de definir su métrica, pero que tenga una probabilidad mayor de pertenencia respecto de otro grupo, de la misma forma que un patrón en la partición difusa puede tener un grado de pertenencia mayor a un grupo y una probabilidad de pertenencia mayor a otro. La probabilidad de error cuando afirmamos que un patrón

pertenece a un grupo es $1-max\left\{u_{xk}^{(h)} | 1 \le k \le c\right\}$, y en consecuencia la probabilidad de error media será:

$$B = E\left\{1-max\left\{u_{xk}^{(h)} | 1 \le k \le c\right\}\right\} \qquad (14)$$

similar a como se calcula el error de Bayes [5], siendo $E$ la esperenza matemática.

De una manera similar a como hemos definido el error medio cometido, podemos definir una relación binaria difusa entre los grupos hard de la partición. Sean $a,b$ las etiquetas de dos de los grupos pertenecientes a la partición hard obtenida. Un patrón $x \in X$ tendrá unas probabilidades de pertenencia hard $u_{xa}^{(h)}, u_{xb}^{(h)}$ a cada uno de los dos grupos. Si $u_{xa}^{(h)}$ ó $u_{xb}^{(h)}$ es la probabilidad máxima de pertenencia, entonces $x$ pertenecerá a uno de los dos grupos. En estas condiciones, la diferencia de las dos probabilidades nos mide en $x \in X$ la probabilidad de que los dos grupos estén separados. La probabilidad de que estén unidos será: $1-\left|u_{xa}^{(h)} - u_{xb}^{(h)}\right|$. En consecuencia, la probabilidad media de que los dos grupos estén unidos será:

$$R_{ab} = E\left\{1-\left|u_{xa}^{(h)} - u_{xb}^{(h)}\right|\right\}$$
$$max\left\{u_{xa}^{(h)}, u_{xb}^{(h)}\right\} = max\left\{u_{xk}^{(h)}\right\} \qquad (15)$$

Obviamente, $R_{aa} = 1$, $R_{ab} \le 1$, $R_{ab} = R_{ba}$, es decir, la relación definida por (15) es reflexiva y simétrica. Utilizando la composición max-min determinaremos la relación de similaridad asociada y el árbol de las particiones [7]. La función de pertenencia de la unión de dos grupos dentro del árbol de las particiones es la usual, definida en [6]. Esto quiere decir que una partición del árbol de las particiones no tiene tantos prototipos como grupos de la partición, sino que un grupo de la partición tiene tantos prototipos como subgrupos hayan sido unidos para formarlo.

## 4. TEST DEL ALGORITMO FM.

Uno de los inconvenientes mayores que se encuentra cualquier proceso de investigación aplicado es poder contrastar los resultados de la investigación en el entorno de la realidad social y económica. Por otra parte, los datos del entorno económico y social de más aplicación suelen ser voluminosos y utilizan muchas variables, y la comprobación de resultados puede ser compleja. Para poder testear, hemos seleccionado un conjunto de datos correspondientes a un número razonable de entidades bien conocidas por nosotros y que su clasificación pueda ser contrastada. Los datos corresponden al activo o inversión de oficinas de la Caja de Ahorros de Murcia.

Simplificando mucho, el negocio de una oficina bancaria consiste en captar dinero de unos clientes y prestárselo a otros. Una oficina capta dinero a través de cuentas corrientes, ahorro ordinario, ahorro plazo, etc, y presta a través de préstamos hipotecarios, préstamos personales, pólizas de crédito, etc. Puede ocurrir que el mercado en el que trabaja una oficina sea captador, es decir, que sea relativamente fácil captar dinero y relativamente difícil invertirlo (prestarlo), y al contrario. Una oficina "captadora" puede prestar o una oficina "inversora" a través de un pool de fondos. A efectos de este trabajo, podemos considerar que todo lo que una oficina capta lo invierte, bien a sus clientes o bien a sus clientes y a otras oficinas, y la cantidad captada o invertida se denomina activo total de la oficina. La posición de una oficina está reflejada por los saldos de los productos de pasivo (captación) y de activo (inversión), que se expresan como tantos por ciento de los saldos medios de los productos respecto del saldo medio del activo total de la oficina. Esta forma de tratar los datos es una práctica normal bancaria.

Se han seleccionado 187 oficinas. Las variables seleccionadas son: préstamos hipotecarios, préstamos personales, pólizas de crédito, cartera comercial y pool de fondos. Las variables utilizadas representan aproximadamente un 90% de la inversión de las oficinas.

Se han detectado 9 grupos de los que 3 están formados por patrones espúreos. La figura 1 representa el árbol de las particiones deducido de la relación de similaridad que se deduce de la relación difusa (15) utilizando la composición max-min, para los 6 grupos no espúreos. A la derecha del árbol de las particiones figuran los $\alpha$-cortes unidos por líneas de puntos al árbol. Entre las líneas de puntos se han reflejado el número de grupos de cada partición y el error medio cometido al considerar que los grupos están separados (14). El error medio cometido al considerar los 9 grupos es 0.362 que es muy alto. Para 7 grupos el error es 0.19, que, teniendo en cuenta que es medio, se ha considerado como bueno.

Los resultados obtenidos son buenos y se corresponden con la realidad. El uso de la distancia Euclídea tiene la ventaja de la fácil interpretación de los grupos formados, porque se diferencian los patrones de la misma forma que intuitivamente los diferencia el gestor.

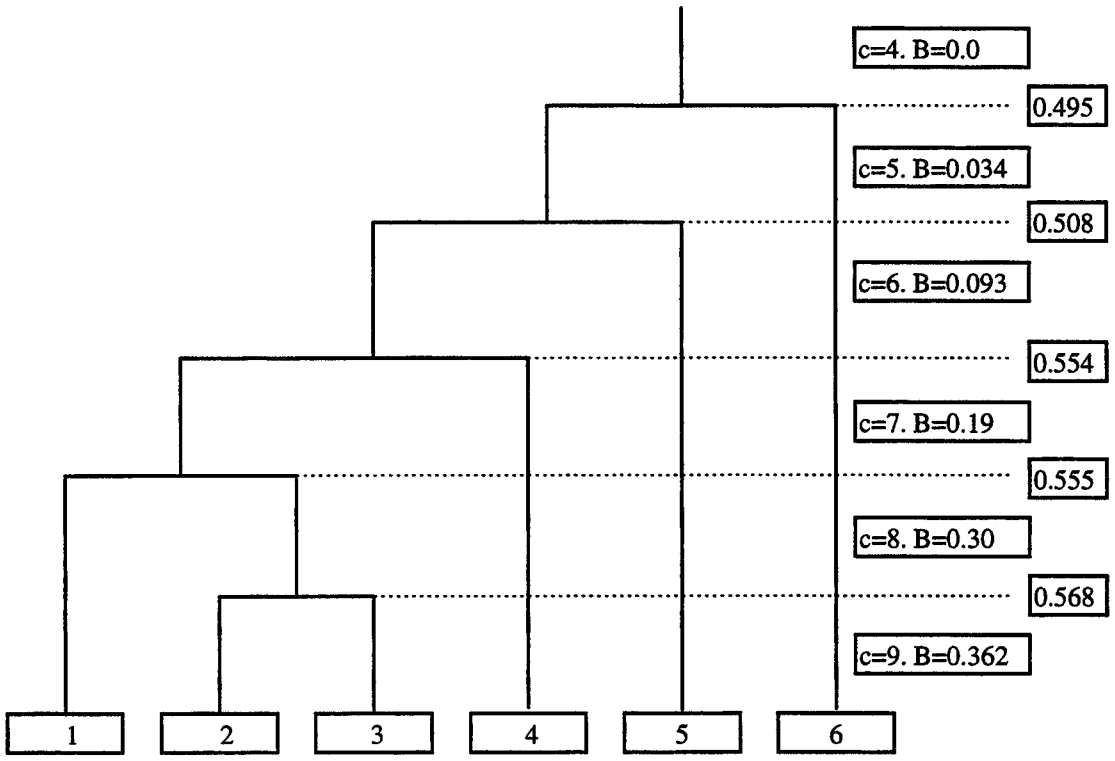



figura 1. Arbol de las particiones de los grupos no espúreos.

## 5. CONCLUSION.

El algoritmo Fuzzy Minimals se puede ejecutar en paralelo, no necesita conocer a priori el número de grupos, y detecta los patrones espúreos. Tiene el inconveniente de que utiliza sólo la distancia Euclídea, lo que implica que estamos suponiendo que el espacio de las características es homogéneo e isótropo [1]. Esta suposición está implicando a su vez que aceptamos un error por lo que se aparte el espacio de las características de la homogeneidad e isotropía, error que por otra parte es inherente a cualquier proceso de clustering. La elección de la medida de dis-similaridad es un problema importante que se aparta del objetivo de este trabajo, sin embargo parece razonable pensar que en aplicaciones de gestión y comerciales, la aproximación de homogeneidad e isotropía para el espacio de las características puede ser adecuada.

Hemos definido un nuevo concepto de partición hard y una relación de similaridad entre los grupos de la partición hard que nos permite encontrar el árbol de las particiones. También hemos definido una medida de la probabilidad media de error que cometemos cuando consideramos que los grupos están separados. El error medido nos permite seleccionar una partición con un error adecuado al problema que estemos tratando, selección que normalmente estará condicionada por el conocimiento del experto, y posiblemente por alguna otra información obtenida con técnicas diferentes a las del análisis cluster.

## REFERENCIAS.

[1] Duda P., Hart P.,"Pattern Classification and Scene Analysis",Wiley,New York(1973).

[2] Flores-Sintas A,"Clasificación Particional Difusa No Supervisada Aplicando Propiedades Geométricas Locales", Tesis Doctoral, Universidad de Murcia, (1997).

[3] Flores-Sintas A.,Cadenas J.M.,Martin F.,"A Local Geometrical Properties Application to Fuzzy Clustering", Fuzzy Sets and Systems, to be published.

[4] Flores-Sintas A.,Cadenas J.M.,Martin F.,"Membership Functions in the Fuzzy C-Means Algorithm", Fuzzy Sets and Systems, to be published.

[5] Fukunaga K.,"Introduction to statistical pattern recognition",(Academic Press,1990).

[6] Zadeh L.,"Fuzzy sets", Inform. and Control, 8-3(1965) 338-353.

[7] Zadeh L., "Similarity relations and fuzzy orderings", Inf.. Sci., 3 (1971) 177-200.

# La Envolvente Probabilística de Variables Aleatorias Difusas

**Inés Couso Blanco**
Dpto. Estadística e I.O. y D.M.
C/Calvo Sotelo s/n
33007 Oviedo (Asturias)
couso@pinon.ccu.uniovi.es

**Pedro Gil Alvarez**
Dpto. Estadística e I.O. y D.M.
C/Calvo Sotelo s/n
33007 Oviedo (Asturias)
pedro@pinon.ccu.uniovi.es

## Resumen

Considerando que un conjunto aleatorio es una observación imprecisa de una variable aleatoria, definimos la envolvente probabilística de un conjunto aleatorio como el conjunto de todas las probabilidades inducidas por las variables aleatorias compatibles con él. Este concepto es de utilidad en el modelado de experimentos aleatorios observados de forma imprecisa, siempre y cuando esta imprecisión se caracterice mediante conjuntos que contienen al resultado del experimento. El tratamiento que se hace está estrechamente ligado con los conjunto convexos de probabilidades y los intervalos probabilísticos estudiados por Campos et al.(1994), Campos y Moral (1995), Walley y Fine (1982), entre otros.

De forma más general, cuando las observaciones del experimento se caracterizan mediante conjuntos borrosos, el concepto clásico de probabilidad inducida por una variable aleatoria tiene su correspondencia en la envolvente probabilística de una variable aleatoria difusa, que será definida en este trabajo. Además de su definición, se expondrán algunas propiedades. En particular, comprobaremos que no se trata de un funcional aditivo y presentaremos algunas conclusiones al respecto.

## 1 Introducción. Concepto de envolvente probabilística de conjuntos aleatorios

Cuando se realiza un experimento aleatorio, la salida que se obtiene puede ser modelada mediante el concepto formal de variable aleatoria. Esa variable aleatoria induce una distribución de probabilidad en el espacio de las salidas posibles.

Ahora bien, en muchas mediciones físicas, no podemos observar de manera exacta, para cada $\omega \in \Omega$, la salida $X(\omega) \in \Omega'$, sino que únicamente podemos localizarla en un subconjunto de $\Omega'$, $\Gamma(\omega)$. Es decir, disponemos de una aplicación multivaluada, $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$, de manera que el valor exacto de la salida, $X(\omega)$ pertenece al conjunto $\Gamma(\omega)$, para cada elemento $\omega \in \Omega$.

En este caso, no podemos determinar, para cada conjunto medible, $A \in \mathcal{A}'$, el valor de la probabilidad inducida por $X$ para $A$, $P_X(A)$, sino que solamente podemos conocer un subconjunto de $[0,1]$ al cual pertenece. Este conjunto se denominará envolvente probabilística de $\Gamma$ y lo denotaremos por $P_\Gamma(A)$.

### 1.1 Conceptos preliminares

La condición de medibilidad de una aplicación multivaluada que consideraremos a lo largo de esta trabajo será la siguiente:

**Definición 1.1 (Nguyen, 1978)** *Sean $(\Omega, \mathcal{A})$ y $(\Omega', \mathcal{A}')$ dos espacios medibles. Diremos que una aplicación multivaluada, $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$, es fuertemente medible si se verifica, para todo subconjunto medible de $\Omega'$, $A \in \mathcal{A}'$, que el conjunto:*

$$A^* = \Gamma^*(A) = \{\omega \in \Omega \,|\, \Gamma(\omega) \cap A \neq \emptyset\} \in \mathcal{A}.$$

A lo largo del trabajo, utilizaremos indistintamente los términos conjunto aleatorio y aplicación fuertemente medible.

### 1.2 Definición y resultados

**Definición 1.2** *Sea $(\Omega, \mathcal{A}, P)$ un espacio probabilístico. Sea $(\Omega', \mathcal{A}')$ un espacio medible y sea $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$ un conjunto aleatorio. Llamamos envolvente probabilística de $\Gamma$ a la aplicación:*

$P_\Gamma : \mathcal{A}' \longrightarrow \mathcal{P}[0,1]$ *dada por:*

$$P_\Gamma(A) = \{P_U(A) \,|\, U \in C(\Gamma)\},$$

*donde:*

$$C(F) = \{U : \Omega \longrightarrow \Omega' \text{ medible } U(\omega) \in \Gamma(\omega) \ c.s.(P)\}.$$

Obsérvese que, en la definición, para cada aplicación medible, $U$, perteneciente a $C(\Gamma)$ no se exige que todos los puntos $U(\omega)$ estén incluidos en los conjuntos $\Gamma(\omega)$, sino solamente casi todos. De esta forma, la envolvente probabilística de un conjunto aleatorio coincide con la envolvente probabilística de dicho conjunto aleatorio considerado como aplicación fuertemente medible en el espacio completado.

**Ejemplo** Sea $\Omega = \{1,2\}$, $\mathcal{A} = \mathcal{P}(\Omega)$, $P(\{i\}) = 1/2$ $i = 1,2$.

$\Omega' = \{1,2,3\}$, $\mathcal{A}' = \mathcal{P}(\Omega')$, $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$.

$\Gamma(1) := \{1,2\}$ $\Gamma(2) := \{2,3\}$. Entonces:

$P_\Gamma(\{1\}) = \{0, 1/2\}$

$P_\Gamma(\{2\}) = \{0, 1/2, 1\}$

$P_\Gamma(\{3\}) = \{0, 1/2\}$

$P_\Gamma(\{1,2\}) = \{1/2, 1\}$

$P_\Gamma(\{2,3\}) = \{1/2, 1\}$

$P_\Gamma(\{1,3\}) = \{0, 1/2, 1\}$

$P_\Gamma(\emptyset) = \{0\}$

$P_\Gamma(\Omega') = \{1\}$

### 1.3 Propiedades de la envolvente probabilística de los conjuntos aleatorios

Dada una aplicación fuertemente medible, $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$, y dado un conjunto medible, $A \in \mathcal{A}'$ el conjunto $P_\Gamma(A)$ está acotado inferior y superiormente de la manera que detallamos en el siguiente resultado debido a Matheron (véase [4]):

**Teorema 1.1** *Sea $(\Omega, \mathcal{A}, P)$ un espacio probabilístico, sea $(\Omega', \mathcal{A}')$ un espacio medible y sea $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$ un conjunto aleatorio. Sea $P_\Gamma : \mathcal{A}' \longrightarrow \mathcal{P}[0,1]$ la envolvente probabilística de $\Gamma$. Sea $A \in \mathcal{A}'$. Entonces el conjunto $P_\Gamma(A)$ está acotado superior e inferiormente por las cantidades:*

$$P^*(A) = P(A^*), \text{ donde } A^* = \{\omega \in \Omega \,|\, \Gamma(\omega) \cap A \neq \emptyset\}$$

$$P_*(A) = P(A_*), \text{ donde } A_* = \{\omega \in \Omega \,|\, \Gamma(\omega) \subseteq A\}.$$

*Las aplicaciones $P^* : \mathcal{A}' \longrightarrow [0,1]$, $P_* : \mathcal{A}' \longrightarrow [0,1]$, son capacidades de Choquet conjugadas de orden $\infty$, alternada y monótona respectivamente.*

Estudiemos ahora la estructura del conjunto $P_\Gamma(A)$. Se demuestra en [3] que, dados un espacio de probabilidad, $(\Omega, \mathcal{A}, P)$ , un espacio medible, $(\Omega', \mathcal{A}')$ y un conjunto aleatorio, $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$, la envolvente probabilística de un conjunto medible $A \in \mathcal{A}'$, $P_\Gamma(A)$ está incluida en el subconjunto de $[0,1]$:

$$\{P(C) \,|\, C \in \mathcal{A} \quad A_* \subseteq C \subseteq A^*\}$$

También se prueba que, en el caso de que se alcancen las cotas, $P^*(A)$ y $P_*(A)$, se da la igualdad entre los dos conjuntos ($P_\Gamma(A) = \{P(C) \,|\, C \in \mathcal{A} \quad A_* \subseteq C \subseteq A^*\}$).

Esto nos lleva a pensar que el conjunto $P_\Gamma(A)$ no ha de ser necesariamente convexo. Por ejemplo, si el espacio inicial es discreto y $P_\Gamma(A)$ no es un conjunto unipuntual, no se verifica la propiedad de convexidad. Esto ocurre precisamente en el ejemplo dado al comienzo de esta sección.

A continuación damos una condición suficiente para que se cumpla la propiedad de convexidad.

**Proposición 1.2** *Sea $(\Omega, \mathcal{A}, P)$ un espacio probabilístico no atómico. Sea $(\Omega', \mathcal{A}')$ un espacio medible. Sea $\Gamma : \Omega \longrightarrow \mathcal{P}(\Omega')$ un conjunto aleatorio. Entonces, para todo $A \in \mathcal{A}'$, el conjunto $P_\Gamma(A)$ es convexo.*

El resultado anterior se encuentra demostrado en [3].

Como se citó al comienzo de este trabajo, la envolvente probabilística de conjuntos aleatorios está relacionada con los conjuntos de probabilidades estudiados por Campos et al. (1994), Campos y Moral (1995), Walley y Fine (1982), entre otros.

Sin embargo, en nuestro caso, no es necesaria la condición de convexidad y el conjunto universal, $\Omega'$, no es necesariamente finito, si bien se da el requerimiento de que el conjunto de probabilidades esté acotado por capacidades de Choquet de orden $\infty$.

## 2 Envolvente probabilística de una variable aleatoria difusa

En la sección anterior hemos modelado la imprecisión en las mediciones de la salida de un experimento aleatorio por medio de conjuntos aleatorios, y mencionamos que para un valor $r \in \Omega'$, $r \notin \Gamma(\omega)$ significaba que la variable aleatoria desconocida no tomaba el valor $r$ en $\omega$. Ahora iremos un paso más allá, y modelaremos esta imprecisión por medio de conjuntos difusos. Una variable aleatoria difusa

$$\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\Omega')$$

define una función $\mu_{\widetilde{X}(\omega)}(r)$, que asocia a cada elemento de $\Omega'$ un valor entre 0 y 1. Este valor puede interpretarse como el grado de confianza que tenemos en que la variable aleatoria desconocida tome el valor $r$.

### 2.1 Conceptos y resultados preliminares

La envolvente probabilística de una variable aleatoria difusa, que definiremos en la sección 3 de este trabajo, no verifica, en general, todas las propiedades de los conjuntos difusos, sino que cumple unas condiciones menos restrictivas que caracterizan a los llamados conjuntos graduales [9].

A continuación se muestra la definición de conjunto gradual, así como un resultado que relaciona los conceptos de conjunto gradual y de conjunto difuso.

**Definición 2.1 (Herencia,1995)** *Dado un conjunto arbitrario, $X$, llamaremos subconjunto (o parte) gradual de $X$ (o simplemente conjunto gradual) a cualquier aplicación decreciente de $\mathbb{I}$ en $\mathcal{P}(X)$. Es decir, cualquier $\psi : \mathbb{I} = [0,1] \longrightarrow \mathcal{P}(X)$ que cumpla*

$$\forall \alpha, \beta \in \mathbb{I}, [\alpha < \beta \Rightarrow \psi(\alpha) \supseteq \psi(\beta)].$$

*Denotaremos por $\mathcal{G}(X)$ al conjunto de partes graduales de $X$. Diremos que $\psi \in \mathcal{G}(X)$ es crisp cuando toma un valor constante $A \in \mathcal{P}(X)$. En este caso el conjunto gradual será denotado por $\psi = (A)_{\mathbb{I}}$ o incluso por $\psi = A$. Esta identificación permite considerar que $\mathcal{P}(X) \subseteq \mathcal{G}(X)$.*

**Teorema 2.1 (Herencia,1995)** *Para cualquier conjunto, $X$ y cualquier aplicación, $\psi : \mathbb{I} \longrightarrow \mathcal{P}(X)$ se cumple:*

1. *La condición necesaria y suficiente para que $\psi$ sea un conjunto gradual es la existencia de un único conjunto difuso $\mu$ que verifique:*
   $\mu^{(\alpha)} \subseteq \psi(\alpha) \subseteq \mu^{[\alpha]}$, $\forall \alpha \in \mathbb{I}$.
2. *Dicho conjunto está determinado, entonces por:*
   $\mu(x) = \sup\{\alpha \in \mathbb{I} \mid x \in \psi(\alpha)\}$, $\forall x \in X$ *y sus $\alpha-$cortes son ($\forall \alpha \in \mathbb{I}$):*

$$\mu^{[\alpha]} = \cap\{\psi(\beta) \mid \beta \in [0, \alpha)\}$$

$$\mu^{(\alpha)} = \cup\{\psi(\beta) \mid \beta \in (\alpha, 1]\}$$

**Definición 2.2 (Herencia, 1995)** *Consideremos la aplicación $\Theta : \mathcal{G}(X) \longrightarrow \widetilde{\mathcal{P}}(X)$, $\Theta(\psi) = \mu_\psi$, siendo:*

$$\mu_\psi(x) := \sup\{\alpha \in \mathbb{I} \mid x \in \psi(\alpha)\}.$$

*Denominaremos $\mu_\psi$ al conjunto difuso asociado al conjunto gradual $\psi$. Diremos que dos conjuntos graduales, $\psi$ y $\Phi$ son equivalentes si tienen el mismo conjunto difuso asociado ($\mu_\psi = \mu_\Phi$).*

Utilizaremos la condición de medibilidad fuerte de Nguyen para definir una variable aleatoria difusa, como se muestra a continuación.

**Definición 2.3** *Sean $(\Omega, \mathcal{A})$ y $(\Omega', \mathcal{A}')$ dos espacios medibles. Sea $\widetilde{\mathcal{P}}(\Omega')$ el conjunto de las partes difusas de $\Omega'$. Diremos que una aplicación, $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\Omega')$ es una variable aleatoria difusa si, para cada $\alpha \in \mathbb{I}$, la aplicación multivaluada $X_\alpha(\omega) := [X(\omega)]^{[\alpha]}$ $\forall \omega \in \Omega$ es un conjunto aleatorio (aplicación fuertemente medible).*

### 2.2 Definición y propiedades

**Definición 2.4** *Sea $(\Omega, \mathcal{A}, P)$ un espacio probabilístico. Sea $(\Omega', \mathcal{A}')$ un espacio medible. Sea $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\Omega')$ una variable aleatoria difusa. Llamamos envolvente probabilística de $\widetilde{X}$ a la aplicación $P_{\widetilde{X}} : \mathcal{A}' \longrightarrow \mathcal{G}[0,1]$ dada por:*

$$P_{\widetilde{X}}(A) := P_{\widetilde{X}_\alpha}(A) \quad \forall A \in \mathcal{A}'.$$

Es fácil comprobar que se trata de una buena definición, es decir que la imagen por $P_{\widetilde{X}}$ de un conjunto $A$ es única y pertenece a $\mathcal{G}[0,1]$.

Dado que para $A \in \mathcal{A}'$, $P_{\widetilde{X}}(A)$ es un subconjunto gradual de $[0,1]$, podemos buscar la expresión del conjunto difuso asociado. Dicha expresión se muestra en los siguientes resultados.

**Proposición 2.2** *Sea $(\Omega, \mathcal{A}, P)$ un espacio probabilístico completo. Sea $(\Omega', \mathcal{A}')$ un espacio medible. Sea $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\Omega')$ una variable aleatoria difusa. Sea $A \in \mathcal{A}'$ un conjunto arbitrario. Entonces el conjunto difuso asociado al conjunto gradual $P_{\widetilde{X}}(A)$ viene dado por la función de pertenencia:*

$$\mu_{\widetilde{P}_{\widetilde{X}}(A)}(p) =$$

$$\sup\{\mu_{\widetilde{X}}(U) \mid U : \Omega \longrightarrow \Omega' \text{ v.a. } P_U(A) = p\}$$

*donde $\mu_{\widetilde{X}}(U) = \inf_{\omega \in \Omega} \mu_{\widetilde{X}(\omega)}(U(\omega))$ y se puede interpretar como el grado de confianza de que $U$ es la verdadera variable aleatoria que rige el experimento.*

**Demostración:**

Vamos a demostrar que, dado $A \in \mathcal{A}'$, se verifica, para todo $\alpha \in [0,1]$ que:

$$[\widetilde{P}_{\widetilde{X}}(A)]^{(\alpha)} \subseteq P_{\widetilde{X}}(A)(\alpha) \subseteq [\widetilde{P}_{\widetilde{X}}(A)]^{[\alpha]}.$$

1. $[\widetilde{P}_{\widetilde{X}}(A)] \subseteq P_{\widetilde{X}}(A)(\alpha)$

   $[\widetilde{P}_{\widetilde{X}}(A)]^{(\alpha)} = \{r \in [0,1] \mid \sup A_r > \alpha\}$ donde:
   $A_r = \{\mu_{\widetilde{X}}(U) \mid U : \Omega \longrightarrow \Omega' \text{ v.a. } P_U(A) = r\}$. Sea $r \in [0,1]$ tal que $\sup A_r > \alpha$. Entonces existe algún $a_r \in A_r$ tal que $a_r > \alpha$. Por tanto, existe una variable aleatoria, $U : \Omega \longrightarrow \Omega'$, tal que $P_U(A) = r$ y de manera que $\mu_{\widetilde{X}}(U) = \inf_{\omega \in \Omega} \mu_{\widetilde{X}(\omega)}(U(\omega)) > \alpha$.

   Así que existe una variable aleatoria, $U : \Omega \longrightarrow \Omega'$, tal que $P_U(A) = r$ y $U(\omega) \in [\widetilde{X}(\omega)]^{(\alpha)} \subseteq [\widetilde{X}(\omega)]^{[\alpha]}$ $\forall \omega \in \Omega$. Por tanto, $r \in [P_{\widetilde{X}}(A)](\alpha)$.

2. $P_{\widetilde{X}}(A)(\alpha) \subseteq [\widetilde{P}_{\widetilde{X}}(A)]^{[\alpha]}$.

   $[\widetilde{P}_{\widetilde{X}}(A)]^{[\alpha]} = \{r \in [0,1] \mid \sup A_r \geq \alpha\}$, donde $A_r$ es el conjunto definido anteriormente.

   Sea $r \in P_{\widetilde{X}}(A)(\alpha)$. Entonces existe una variable aleatoria $U : \Omega \longrightarrow \Omega'$ tal que $P_U(A) = r$ y $U(\omega) \in [\widetilde{X}(\omega)]^{[\alpha]}$ para todo $\omega \in \Omega$. Así que la variable aleatoria, $U : \Omega \longrightarrow \Omega'$ verifica que $P_U(A) = r$ y $\mu_{\widetilde{X}}(U) \geq \alpha$. Por tanto, $\sup A_r \geq \alpha$ y $r \in [\widetilde{P}_{\widetilde{X}}(A)]^{[\alpha]}$.

**Proposición 2.3** *Sea $(\Omega, \mathcal{A}, P)$ un espacio probabilístico. Sea $(\Omega', \mathcal{A}')$ un espacio medible y sea $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\Omega')$ una variable aleatoria difusa. Sea $A \in \mathcal{A}'$ un conjunto arbitrario.*

*El conjunto difuso asociado al conjunto gradual $P_{\widetilde{X}}(A)$ viene dado por la función de pertenencia:*

$\mu_{\widetilde{P}_{\widetilde{X}}(A)}(p) = \sup\{\mu_{\widetilde{X}}(U) \mid U : \Omega \longrightarrow \Omega'$
$\exists\, V = U \text{ c.s.}(P) \text{ v.a. } P_V(A) = p\}$.

**Demostración:**

Como $[\widetilde{P}_{\widetilde{X}}(A)](\alpha) = \{\overline{P}[U^{-1}(A)] \mid U \text{ es } \overline{\mathcal{A}} - \mathcal{A}' \text{ medible}, U(\omega) \in \widetilde{X}_\alpha(\omega)\ \forall \omega \in \Omega\} =$
$\{P_V(A) \mid \exists\, U = V \text{ c.s.}(P)\ U(\omega) \in \widetilde{X}_\alpha(\omega)\ \forall \omega \in \Omega\}$

repitiendo la demostración de la proposición anterior, obtenemos el resultado.

A continuación exponemos algunas de las propiedades que verifica la envolvente de una variable aleatoria difusa.

**Proposición 2.4** *Sea $(\Omega, \mathcal{A}, P)$ un espacio de probabilidad. Sea $(\Omega', \mathcal{A}')$ un espacio medible y sea $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\Omega')$ una variable aleatoria difusa. Si existe algún $A_0 \in \mathcal{A}'$ tal que $\widetilde{P}_{\widetilde{X}}(A_0)$ es un conjunto difuso normal, entonces:*

1. $\widetilde{P}_{\widetilde{X}}(\emptyset) = \{0\} \in \mathcal{P}([0,1])$
2. $\widetilde{P}_{\widetilde{X}}(\Omega) = \{1\} \in \mathcal{P}([0,1])$
3. *Sean dos conjuntos $A, B \in \mathcal{A}'$ $A \subseteq B$. Entonces:*
   - *Si $p_1 \in \widetilde{P}_{\widetilde{X}}(A)(\alpha)$ entonces existe $p_2 \in \widetilde{P}_{\widetilde{X}}(B)(\alpha)$ tal que $p_2 \geq p_1$*
   - *Si $p_1 \in \widetilde{P}_{\widetilde{X}}(B)(\alpha)$ entonces existe $p_2 \in \widetilde{P}_{\widetilde{X}}(A)(\alpha)$ tal que $p_2 \leq p_1$*
4. *Sea $\{A_i\}_{i \in N} \subseteq \mathcal{A}'$ una sucesión de conjuntos disjuntos dos a dos. Entonces se verifica que:*

$$\widetilde{P}_{\widetilde{X}}(\cup_{i=1}^{\infty} A_i) \subseteq \widetilde{\sum_{i=1}^{\infty}} \widetilde{P}_{\widetilde{X}}(A_i).$$

La envolvente probabilística de una variable aleatoria difusa no es, en general aditiva.

En el resultado siguiente presentamos algunas condiciones equivalentes a la condición de aditividad de $\widetilde{\mathcal{P}}_{\widetilde{X}}$:

**Proposición 2.5** *Sea $(\Omega, \mathcal{A}, P)$ un espacio de probabilidad. Sea $(\Omega', \mathcal{A}')$ un espacio medible. Sea $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\Omega')$ una variable aleatoria difusa tal que $P_{\widetilde{X}}(A)(1) \neq \emptyset$ $\forall A \in \mathcal{A}'$.*

*Entonces las condiciones siguientes son equivalentes:*

1. $\#P_{\widetilde{X}}(A)(\alpha) = 1$ *para todo* $\alpha \in (0,1)$

2. $\widetilde{P}_{\widetilde{X}}(A)$ *es un conjunto nítido unipuntual para todo conjunto* $A \in \mathcal{A}'$.

3. $P_{\widetilde{X}}$ *es* $\sigma$- *aditiva*

4. $P_{\widetilde{X}}$ *es aditiva*

5. $\widetilde{P}_{\widetilde{X}}$ *es* $\sigma$- *aditiva*

6. $\widetilde{P}_{\widetilde{X}}$ *es aditiva*

Se puede comprobar que si, para cada elemento $\omega \in \Omega$, $\widetilde{X}(\omega)$ es un conjunto nítido unipuntual (es decir, $\widetilde{X}$ representa una variable aleatoria clásica) la envolvente probabilística de $\widetilde{X}$ es $\sigma$- aditiva y representa la probabilidad inducida por la variable aleatoria clásica.

Nos podemos preguntar si se verifica el resultado recíproco, es decir, si $\widetilde{P}_{\widetilde{X}}$ es aditiva (o, equivalentemente, $\sigma-$ aditiva) la variable aleatoria difusa $\widetilde{X}$ ¿ es una variable aleatoria clásica?.

En los resultados que exponemos a continuación se da respuesta parcial a la pregunta anterior,

**Proposición 2.6** *Sea* $(\Omega, \mathcal{A}, P)$ *un espacio de probabilidad. Sea* $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\mathbb{R})$ *una variable aleatoria difusa (medible Borel). Supongamos que* $\widetilde{X}_\alpha(\omega)$ *es un conjunto cerrado para todo* $\alpha \in [0,1]$ *y para todo* $\omega \in \Omega$. *Las condiciones siguientes son equivalentes:*

1. $\widetilde{\mathcal{P}}_{\widetilde{X}}(A)$ *es un conjunto clásico unipuntual, para cada* $A \in \beta_{\mathbb{R}}$

2. $\{\omega \in \Omega \mid \widetilde{X}(\omega) \notin U(\mathbb{R})\}$ *es un conjunto nulo.*

   *Donde* $U(\mathbb{R}) = \{\{x\} \mid x \in \mathbb{R}\}$ *es la subclase de* $\mathcal{P}(\mathbb{R})$ *de los conjuntos unipuntuales de* $\mathbb{R}$.

**Proposición 2.7** *Sea* $(\Omega, \mathcal{A}, P)$ *un espacio de probabilidad. Sea* $\widetilde{X} : \Omega \longrightarrow \widetilde{\mathcal{P}}(\mathbb{R})$ *una variable aleatoria difusa (medible Borel). Supongamos que* $\widetilde{X}_\alpha(\omega)$ *es un conjunto convexo para todo* $\alpha \in [0,1]$ *y para todo* $\omega \in \Omega$. *Las condiciones siguientes son equivalentes:*

1. $\widetilde{\mathcal{P}}_{\widetilde{X}}(A)$ *es un conjunto clásico unipuntual, para cada* $A \in \beta_{\mathbb{R}}$

2. $\{\omega \in \Omega \mid \widetilde{X}(\omega) \notin U(\mathbb{R})\}$ *es un conjunto nulo, donde* $U(\mathbb{R}) = \{\{x\} \mid x \in \mathbb{R}\}$ *es la subclase de* $\mathcal{P}(\mathbb{R})$ *de los conjuntos unipuntuales de* $\mathbb{R}$.

## 3 Conclusiones

Como se ha visto, la envolvente probabilística de una variable aleatoria difusa generaliza el concepto clásico de probabilidad inducida por una variable aleatoria. Al contrario que esta última, no es, en general, aditiva, y solamente lo es en el caso trivial en que representa la probabilidad inducida por una variable aleatoria clásica.

Las envolventes probabilísticas de conjuntos aleatorios y variables aleatorias difusas son aplicables al tratamiento estadístico de datos imprecisos. A este respecto, se desea reseñar que este estudio ha sido utilizado como base formal de trabajos aplicados en el campo del aprendizaje de máquinas y regresión a partir de datos medidos con imprecisión [13].

## Referencias


[1] L.M. de Campos, J. Huete y S. Moral (1994) Probability intervals: A tool for uncertain reasoning, *Int.Journal of Unceirtainty, Fuzziness and Knowledge-Based Systems* **2** 167-196.

[2] L.M. de Campos y S. Moral (1995) *Independence Concepts for Convex Sets of Probabilities.* Informe Técnico. Universidad de Granada.

[3] I.Couso (1997) *La envolvente probabilística. Definición y propiedades.* Trabajo de Investigación. Departamento de Estadística. Universidad de Oviedo.

[4] N.A.C. Cressie (1995) *Statistics for Spatial Data.* John Willey and Sons.

[5] L. DeRobertis y J.A. Hartigan (1981) Bayesian Inference Using Intervals of Measures, *The Annals of Statistics* **9** 235-244.

[6] A.P. Dempster (1967) Upper and Lower Probabilities Induced by a Multi-valued Mapping, *Ann. Math. Statistics* **38** 325-339.

[7] I.R. Goodman y H.T. Nguyen (1985) *Uncertainty Models for Knowledge- Based Systems*, Elsevier Science Publishers.

[8] M. Grabbish, H.T. Nguyen y E.A. Walker (1995) *Fundamentals of Uncertainty Calculi with Applications to Fuzzy Inference*, Kluwer Academic Publishers.

[9] J.A. Herencia (1995) *Origen y Uso de los conjuntos Graduales.* Tesis Doctoral. Departamento de Matemáticas. Universidad de Córdoba.

[10] C.J. Himmelberg (1975) Mesurable relations. *Fund. Math.* **87** 53-72

[11] H.T. Nguyen (1978) On Random Sets and Belief Functions. *J.Math. Analysis and Applications* **65** 531-542

[12] M.L. Puri y Ralescu (1986) Fuzzy Random Variables, *J. Math. Anal. Appl.* **114** 409-422.

[13] L. Sánchez, I. Couso (1997) Modelado a partir de ejemplos imprecisos mediante algoritmos GA-P. ESTYLF 97. Tarragona.

[14] P. Walley y T.L. Fine (1982)Towards a Frequentist Theory on Upper and Lower Probability , *The Annals of Statistics* **10** 741-761.

[15] L.A.Wasserman (1990) Prior Envelopes Based on Belief Functions, *The Annals of Statistics* **18** 454-464.

# NUEVOS ASPECTOS DE EXTRACCION DE REGLAS DIFUSAS USANDO REDES NEURONALES. *

**José M. Benítez** **Armando Blanco**
Dept. Ciencias Computación e I. A.
E. T. S. Ingeniería Informática
Universidad de Granada
Avda. Andalucía, 38.
18071 - GRANADA.
jmbs@decsai.ugr.es

**Miguel Delgado** **Ignacio Requena**
Dept. Ciencias Computación e I. A.
E. T. S. Ingeniería Informática
Universidad de Granada
Avda. Andalucía, 38.
18071 - GRANADA.
requena@decsai.ugr.es

## Resumen

En trabajos previos, hemos presentado dos metodologías para obtener reglas difusas que describan el comportamiento de un sistema, usando Redes Neuronales Artificiales (RNA) con el algoritmo *Backpropagation*, a partir de un conjunto de ejemplos del sistema. En este trabajo presentamos algunas modificaciones que permiten obtener mejores resultados, mediante una adaptación o refinamiento de las etiquetas de las variables en cada regla, o la extracción de reglas locales mediante RNA distribuidas. Se incluye también una aplicación interesante, como es la asignación de una semántica a las clases obtenidas en un proceso de clasificación sin clases previas.

**Palabras clave:** Redes Neuronales Artificiales. Aprendizaje. Reglas Difusas. Semántica en procesos de Clasificación.

## 1 INTRODUCCIÓN.

Un problema importante y con muchas aplicaciones en Inteligencia Artificial es la identificación y reproducción de sistemas. Para ello se han utilizado diversos métodos, como las RNA o la obtención de reglas que describan el sistema en estudio. En muchos problemas reales, el uso de reglas difusas es más apropiado, debido a las características del sistema, o porque la información presente es imprecisa o vaga. En trabajos previos hemos presentado dos procedimientos de obtención de reglas difusas para identificar sistemas, usando RNA.

El primero ([2]) se basa en un proceso de selección de reglas, a partir del conjunto de todas las reglas posibles, asignando previamente a cada variable un conjunto de etiquetas. Entrenamos una RNA con los ejemplos disponibles del sistema (reservando un 30% para test), y formamos todas las reglas posibles con todas las variables, teniendo en cuenta el conjunto de etiquetas de cada variable. Comprobamos la adaptación de las reglas a la RNA entrenada, y rechazamos las reglas que no se adaptan bien. Sobre las reglas inicialmente válidas, aplicamos un algoritmo tipo greedy que va seleccionando la mejor regla (de las que quedan), de acuerdo con un coeficiente de cobertura sobre los ejemplos, y la incluimos en el conjunto definitivo de reglas. El proceso finaliza cuando la cobertura del conjunto de reglas es suficiente.

Hemos aplicado el procedimiento al problema de frenado de un coche, descrito en [13]. Hemos obtenido conjuntos con igual o menor número de reglas y con un comportamiento similar a los obtenidos en [13],

El segundo se basa en un proceso constructivo, que permite obtener las variables relevantes para cada salida en el sistema, y no necesita un conjunto previo de etiquetas para cada variables. Se describe con más detalle en la sección siguiente.

Cuando el número de ejemplos disponibles para la identificación del sistema es pequeño, el comportamiento de este método no es suficientemente bueno. Por ello presentamos ahora unas modificaciones que mejoran claramente la identificación de los sistemas.

También presentamos una metodología distribuida (local) para la obtención de las reglas difusas, que mejora los resultados obtenidos con este último procedimiento.

Este trabajo ha sido financiado en parte por los proyectos TIC96-1393-C06-04 de la CICYT, Madrid y SEDI de Caja Asturias, Oviedo.

## 2 METODOLOGÍA CONSTRUCTIVA PARA OBTENER REGLAS DIFUSAS

Este método se basa en parte en el procedimiento indicado en [12], pero considerando una más adecuada representación para variables contínuas, usando reglas difusas en vez de reglas de producción. Las etiquetas de las variables no son previamente asignadas, sino construidas a posteriori de la identificación de las variables que intervienen en cada regla. Consideramos los datos del sistema como números reales normalizados en el intervalo [0,1] y las reglas usan valores difusos (números trapezoidales) [1] para las variables.

Los pasos del procedimiento, descrito en [3], los resumimos en:

a) Obtención de un conjunto de pre-reglas

1. Entrenamos una RNA (con backpropagation), con h neuronas en la única capa oculta, pero añadiendo las salidas como entradas adicionales.
2. Para cada entrada $i$ y salida $j$ (todas en la entrada de la RNA), calculamos

$$S_{ij} = \sum_{k=1}^{h} (w_{ik} - w_{jk})^2$$

3. Cambiamos las entradas originales de los datos por su complemento a uno, y con estas entradas (y las salidas sin cambiar), entrenamos mediante aprendizaje hebbiano otra RNA sin capas ocultas. Los pesos obtenidos son ahora $V_{ij}$.
4. La correlación entre las entradas y salidas se mide con

$$prod_{ij} = S_{ij} \cdot V_{ij}$$

   Para cada salida $j$, ordenamos en forma creciente estos productos.
5. Buscamos un punto de corte adecuado para los productos ordenados, y consideramos las entradas (variables) $i$ que quedan por debajo del punto de corte como las relevantes para la salida $j$. Obtenemos de esta forma, la estructura de las reglas correspondientes a la salida $j$, que llamamos pre-regla, y que expresamos como

$$P_{R_i} : X_p^i, X_q^i, X_r^i \rightarrow Y_j^i$$

b) Obtención de las reglas (etiquetas de las variables)

1. En los ejemplos de entrenamiento, consideramos para cada pre-regla solamente los valores de las variables identificadas en la pre-regla. Hacemos ahora una clasificación (clustering) difusa, rechazando los ejemplos que no clasifican bien. Cada clase (cluster) obtenida dará lugar a una regla efectiva.
2. Para cada variable en la pre-regla, ordenamos los datos de los ejemplos en cada clase, considerándolos como una distribución estadística.
3. Construimos las etiquetas de las variables (número trapezoidal) incluidas en la regla, de la siguiente y sencilla forma:
   El soporte del número trapezoidal viene dado por los valores máximo y mínimo. La moda viene dada por los cuartiles 1 y 3 (la mediana para números triangulares) de la distribución.

Este procedimiento es válido también cuando los datos del sistema no son números reales, sino que vienen representados por valores difusos (ver [3]).

Hemos aplicado este método al problema bien conocido de las plantas IRIS. Consideramos 100 ejemplos (aleatorios) como conjunto de entrenamiento, y los otros 50 como conjunto de test. En la primera etapa hemos obtenido las siguientes pre-reglas:

1. *long-pétalo* $\rightarrow$ *setosa*
2. *anchura-sépalo*, *long-pétalo*, *anchura-pétalo* $\rightarrow$ *versicolor*
3. *long-pétalo*, *anchura-pétalo* $\rightarrow$ *virginica*

Para la obtención de las reglas, usamos el método de clustering de Chiu ([6]), que no necesita el número de clases y sus centros previamente. Las reglas finalmente obtenidas han sido 4 (1 para los casos setosa y versicolor, y 2 para el caso virginica), con un porcentaje de aciertos, tanto en el conjunto de entrenamiento como en el de test, del 96 %.

Hemos probado los 150 ejemplos en el entrenamiento, obteniendo 5 reglas y el 96% de aciertos. Los resultados son buenos en general, clasificando siempre correctamente el tipo *setosa*.

Pensando que los resultados se pueden mejorar, modificando la forma de construir las etiquetas, y teniendo en cuenta que en otros problemas, cuando hay pocos ejemplos disponibles, los resultados no son tan buenos, proponemos la siguiente modificación, que permite obtener reglas refinadas más eficaces.

[1] Representamos los números difusos trapezoidales con 4 parámetros $(a, b, c, d)$ donde $[a, d]$ es el soporte, y $[b, c]$ es la moda del número difuso

## 2.1 REFINAR LAS REGLAS VIA LA CONSTRUCCIÓN DE LAS ETIQUETAS

Cuando el número de ejemplos que se incluyen en cada clase en el proceso de clustering anterior, es pequ´ no, la construcción de las etiquetas de las variables en cada regla no se adapta lo suficiente a la realidad del problema, lo que conduce a un número excesivo de errores.

Para evitar esto, tenemos en cuenta el grado de pertenencia de cada ejemplo a las clases obtenidas, y construimos la lista ordenada de valores para cada variable, considerando que cada ejemplo en el cluster tiene una frecuencia proporcional a su grado de pertenencia. Esto conlleva, que a la hora de determinar los cuartiles, los elementos con mayor grado de pertenencia tienen mayor peso, lo que refleja mejor la realidad del problema.

Además, como se utiliza una muestra de los ejemplos para la construcción de las etiquetas, la construcción del soporte del número trapezoidal mediante los valores máximos y mínimos en el cluster puede estar un poco comprimida respecto de la realidad. Por ello, modificamos los valores extremos del intervalo soporte, en el sentido de ampliarlo en un porcentaje de los valores máximos y mínimos en el cluster (entre el 5% y el 10%).

Esta modificación en la construcción de las etiquetas, que se comporta como un refinamiento de las reglas obtenidas con el proceso de la sección anterior, ha mejorado los resultados de forma importante, sobre todo en problemas con un número pequeño de ejemplos.

En concreto, en el caso del problema de las plantas IRIS, señalado en la sección anterior, se mejoran los resultados, llegando al 99% de éxitos en el conjunto de entrenamiento, y al 98% en el conjunto de test. Si tenemos en cuenta que la selección de ejemplos de entrenamiento y test se hace de forma aleatoria, y que esto puede influir (levemente) en el entrenamiento de la RNA, diversas ejecuciones del proceso completo, pueden llevar a resultados ligeramente diferentes. Sobre 10 ejecuciones realizadas, el porcentaje de errores no supera nunca el 5%, y en dos de ellas hemos obtenido un aprendizaje total, con 100% de éxitos tanto en entrenamiento como en test.

También hemos aplicado este procedimiento a un problema de clasificación de clientes de agencias de viaje. Los datos han sido proporcionados por el Centro CETT de Barcelona, y reflejan 100 ejemplos, con datos de 8 variables para cada ejemplo (entradas) y que se clasifican en 5 clases diferentes (variables de salida).

En el proceso, hemos considerado 70 ejemplos para el entrenamiento y 30 para el test. Al aplicar el proceso tal como se formuló originalmente, el procentaje de errores se situaba entre el 15 y el 20 %. Al aplicar la modificación indicada en este apartado, el aprendizaje fue casi completo (sólo 1 ejemplo, que además estaba en el conjunto de entrenamiento, no fue clasificado correctamente), con un conjunto de 5 reglas en las que intervienen todas las variables.

## 2.2 UNA APLICACIÓN INTERESANTE

En los procesos de clasificación sin clases previas, un aspecto interesante es el de asignar una semántica concreta a las clases obtenidas. En la mayor parte de los ejemplos reflejados en la literatura sobre el tema, este aspecto no es de una relevancia especial, porque lo que importa fundamentalmente es el número de clases obtenidas y sus centros respectivos.

Sin embargo, en aplicaciones reales, este aspecto es crucial, ya que para los expertos que han de manejar las clases obtenidas, es fundamental ponerles un "nombre" a cada clase, que refleje adecuadamente la pertenencia de los ejemplos a cada clase.

En el desarrollo del proyecto de investigación SEDI, financiado por Caja Asturias, hemos aplicado el procedimiento desarrollado en la sección anterior, a la asignación de una semántica a las clases obtenidas en un proceso de segmentación de clientes sin tener clases previamente establecidas. El proceso que inicialmente está siendo satisfactorio, lo hemos realizado de la forma siguiente:

1. Con los datos disponibles realizamos una segmentación sin clases previas, mediante el método de Chiu ([6]). Esto es necesario, ya que el entrenamiento de la RNA que usamos en nuestro procedimiento es supervisado.
2. Con la clasificación obtenida, aplicamos el proceso constructivo a los mismos datos disponibles. Obtenemos un conjunto de reglas difusas.
3. Hacemos una interpretación de las reglas difusas obtenidas, poniéndolas en un lenguaje comprensible por los expertos correspondientes. Esto significa asignarle una semántica inicial a las clases obtenidas.
4. Discutimos con los expertos la bondad de esta semántica, en base a la segmentación obtenida y a la asignación de la semántica.
5. Con las aportaciones del paso anterior, modificamos adecuadamente los parámetros del proceso, repitiendo los pasos del 1 al 4.

6. Establecemos la semántica definitiva, siempre en diálogo con los expertos.

# 3 UN PROCEDIMIENTO DISTRIBUIDO PARA LA EXTRACCIÓN DE REGLAS DIFUSAS

Pensamos que se pueden obtener mejores resultados haciendo una construcción local de las reglas. Se identifican primero las zonas más importantes en el comportamiento del sistema, mediante un clustering previo, y entonces se obtienen reglas difusas para cada zona, como si se trataran de sistemas diferentes. Sería, de alguna forma, un proceso distribuido.

Podemos resumir el proceso en los siguientes pasos:

1. *Identificar las clases de comportamiento importantes en el sistema.* Esto se hace con un procedimiento de clustering difuso (sin clases previas), sobre las variables de entrada.
2. *Obtener subconjuntos de ejemplos.* Fijado un grado de pertenencia $\alpha$, se obtiene un subconjunto de ejemplos para cada cluster (los que tienen un grado de pertenencia la cluster mayor o igual que $\alpha$.
3. *Extracción local de las reglas.* Aplicamos a cada subconjunto de ejemplos el anterior método de obtención de reglas. Obtenemos un conjunto de reglas para cada cluster.
4. *Construcción de la base de reglas difusas.* Se obtiene por la unión de los conjuntos locales de reglas obtenidas en el paso anterior.

Debemos de precisar algunos aspectos:

En el paso 1, usamos un proceso de clasificación sin clases previas. Esto se puede realizar mediante un proceso de clustering jerárquico, o con una RNA tipo Kohonen o mediante el método de Chiu ([6], para identificar el número de clases y sus centros, seguido de cualquier procedimiento de clustering difuso. Nosotros hemos usado el método de Chiu, seguido del método de las Fuzzy C-Means.

En el paso 2, sólo se consideran en cada cluster obtenido, los ejemplos con un grado de pertenencia por encima de un grado $\alpha$, elegido adecuadamente.

Una ventaja de este proceso es que las reglas obtenidas localmente, en general contienen un menor número de variables. Como desventaja debemos citar que en algunos casos se obtiene un mayor número de reglas definitivas.

Como test para esta metodología hemos considerado los bien conocidos problemas PIMA e IRIS.

En el caso PIMA, los 512 ejemplos disponibles (compuestos de 8 entradas y 2 salidas), se repartieron aleatoriamente en un conjunto de entrenamiento con 400 ejemplos y otro de test con 112 ejemplos.

Sobre los datos de entrenamiento hemos realizado un clustering difuso, en la forma indicada más arriba, fijando un corte $\alpha = 0.5$, obteniendo dos clases con 209 y 191 ejemplos respectivamente.

A ambos subconjuntos le hemos aplicado el método descrito en la sección 2, obteniendo 2 y 3 reglas definitivas respectivamente. El conjunto de las 5 reglas ha sido probado con los 112 ejemplos de test, con el 76% de éxitos, lo que está a la altura de los mejores resultados obtenidos con otras técnicas. Estos resultados mejoran un poco los resultados obtenidos con nuestro proceso de obtención de reglas global (74%), y aunque no mejora los resultados obtenidos con otras técnicas, sí reduce el número de reglas necesario, 5 en nuestro caso.

Sin embargo, en el caso IRIS, los resultados no han mejorado. Esto ha sido debido al pequeño número de ejemplos seleccionados en cada uno de las 4 clases obtenidas en el paso 1 (36, 18, 16 y 36, respectivamente), cosa que no pasaba con el problema PIMA. Posteriormente hemos utilizado el procedimiento de construcción de las etiquetas refinado, descrito en la sección 2.1. Los resultados obtenidos ahora alcanzan el 98'7% de éxitos, con el mismo número de reglas que en el proceso de obtención de reglas globalmente.

# 4 CONCLUSIONES Y COMENTARIOS FINALES.

Hemos presentado un procedimiento para obtener reglas difusas, determinando en primer lugar las variables relevantes para cada salida del sistema. En segundo lugar se construyen las etiquetas de cada variable en cada regla difusa, ya que no se necesitan un conjunto previo de etiquetas para cada variable. Los resultados son buenos en general, pero decaen si el número de ejemplos disponibles es pequeño.

Presentamos por ello un refinamiento en el proceso de construcción de las etiquetas, usando la información disponible, de los grados de pertenencia de los ejemplos en cada clase de los segmentos obtenidos en el proceso

de clasificación para obtener las reglas definitivas, lo que permite sensibles mejoras.

Sealamos una importante aplicación al problema de asignar semántica a las clases obtenidas en un proceso de segmentación sin clases previas.

También hemos presentado un procedimiento distribuido para la obtención de las reglas difusas, que mejora los resultados en muchos casos.

Finalmente, pensamos que, usando metodologías basadas en RNA, los resultados se pueden mejorar, atendiendo fundamentalmente al punto de la construcción de las etiquetas difusas. Las ideas que pretendemos utilizar van en dos direcciones. Por un lado, diseñar un algoritmo que nos permita obtener puntos idóneos, alternativos a los cuartiles, para obtener la moda de los números difusos. Por otro lado, mejorar la construcción de las etiquetas utilizando el conocimiento de los ejemplos negativos a cada regla, aspecto éste que no consideramos actualmente.

Adicionalmente, pretendemos mejorar el método, construyendo un procedimiento automático para seleccionar el punto de corte indicado en el paso 5 de la obtención de las pre-reglas.

## Referencias


[1] BENÍTEZ, J.M., BLANCO, A., AND REQUENA, I. *Un procedimiemto empírico con Redes Neuronales para obtener reglas difusas.* IV Congreso de la Asociación Española de Tecnologías y Lógica Fuzzy. Blanes, Septiembre, (1994).

[2] BENÍTEZ, J.M., BLANCO, A., AND REQUENA, I. *An Empirical Procedure to Obtain Fuzzy Rules using Neural Networks.* 6th IFSA Congress. Brasil, July, (1995).

[3] BENÍTEZ, J.M., BLANCO, A., DELGADO, M. AND REQUENA, I. *Neural Methods for Obtaining Fuzzy Rules.* Mathware, vol III, n. 3, pp 371-382. (1996)

[4] CASTRO, J.L. *Fuzzy Logic Controllers are Universal Approximators.* IEEE Trans. on Syst. Man and Cybern. **25**(4), April. (1995).

[5] CAMPOS, L.M. AND GONZÁLEZ, A. *A Subjective Approach for Ranking Fuzzy Numbers.* Fuzzy Sets and Systems, **29**, 145–153. (1989).

[6] CHIU, S.L.*A cluster estimation method with extension to Fuzzy Model Identification.* IEEE Intern. Congress on Fuzzy Systems. pp.1240–1245. (1994).

[7] GONZÁLEZ, A. *A study of the ranking function approach through mean values.* Fuzzy Sets and Systems, **35**, 29–41. (1990).

[8] LEE, C.C. *Fuzzy Logic in Control Systems: Fuzzy Logic Controller.* IEEE Trans. on Systems, Man, and Cybernetics. **20**(2), Part I, 404–419, Part II, 419–435. (1990).

[9] REQUENA, I. Tesis Doctoral: *Redes Neuronales en Problemas de Decisión con Ambiente Difuso.* Depto. Ciencias de la Computación e I.A. Universidad de Granada. (1992).

[10] REQUENA, I., DELGADO, M., AND VERDEGAY, J.L. *Automatic Ranking of Fuzzy Numbers with the Criteria of a Decision-maker Learnt by an Artificial Neural Network.* Fuzzy Sets and Systems, **64**, 1–19. (1994).

[11] REQUENA, I., BLANCO, A., DELGADO, M., AND VERDEGAY, J.L. *A Decision Personal Index to Fuzzy Numbers based on Artificial Neural Networks.* Fuzzy Sets and Systems, 73, pp 185-199. (1995)

[12] SESTITO S. AND DILLON, T. *Knowledge Acquisition of Conjunctive Rules Using Multilayered Neural Networks.* Inter. J. Intelligent Systems, 8, 779–805. (1993).

[13] WOLF, T. *Das Fuzzy-Mobil.* mc 3/91, pp. 50–63. (1991).

# Aprendizaje a partir de Información con Ruido en los Modelos Difuso-Neuronales FasArt y FasBack

**Jose Manuel Cano Izquierdo**
Escuela Técnica Superior de Ingenieros Industriales
Universidad de Murcia
Paseo Alfonso XIII 48,
30203, Cartagena (Murcia)
joscan@upc.um.es

**Yannis Dimitriadis**
Escuela Técnica Superior de Ingenieros de Telecomunicación
Universidad de Valladolid
Cll / Real de Burgos sn ,
47011, Valladolid
yannis@tel.uva.es

**Juán López Coronado**
Escuela Técnica Superior de Ingenieros Industriales
Universidad de Valladolid
Paseo de Cauce sn,
47011, Valladolid
coronado@dali.eis.uva.es

## Resumen

Los modelos Difuso-Neuronales se constituyen como puentes entre los Sistemas Lógico Difusos y las Arquitecturas Neuronales. La Teoría de la Resonancia Adaptativa (ART) contribuye a este campo con diferentes modelos. Basándose en la teoría ART, pero intentando una mejor integración de los conceptos de conjuntos difusos, se han propuesto los modelos FasArt y FasBack. Un problema importante, a la hora de hacer el aprendizaje, es la presencia de ruido en los datos. Nos proponemos evaluar el efecto de este problema en los modelos FasArt y FasBack comparando con otras arquitecturas difuso-neuronales.

**Palabras Clave:** Modelos Neuro-Difusos, Identificación, Teoría de la Resonancia Adaptativa, Aprendizaje con Ruido, Fuzzy ART, Fuzzy ARTMAP, FasArt, FasBack

## 1 Introducción

Los sistemas Lógico Difusos permiten una representación del conocimiento próxima a la descripción lingüística del mismo. Esto hace que sus mecanismos de funcionamiento sean transparentes e identificables. No obstante, para su definición, necesitan de un proceso de extracción del conocimiento para poder plasmarlo en forma de un conjunto de reglas que no es siempre un proceso fácil e inmediato. Las Redes Neuronales permiten construir sistemas con capacidad de aprendendizaje a partir de ejemplos numéricos. La búsqueda de relaciones entre ambos sistemas ha permitido la propuesta de modelos difuso-neuronales que pretenden conjugar las principales características de las dos áreas. La información en este tipo de sistemas se puede describir en forma simbólica, como un conjunto de reglas asociadas a unos determinados conjuntos, a diferencia de los modelos neuronales donde no es posible encontrar una descripción sencilla de sus dinámicas. Al mismo tiempo la característica de autoorganización de las redes neuronales permite que los modelos difuso-neuronales se puedan construir partiendo de datos numéricos mediante un proceso de aprendizaje.

La Teoría de la Resonancia Adaptativa ha aportado diferentes modelos de arquitecturas neuronales aplicables a problemas de clasificación, ya sea de forma no supervisada; ART 1 [6] para vectores binarios y ART 2 [5] para vectores analógicos, como problemas supervisados mediante la arquitectura ARTMAP [8]. La introducción de los conceptos de la teoría de conjuntos difusos, desarrollada a partir de las propuestas de Zadeh [17], llevó a la propuesta de los modelos Fuzzy ART [9] y Fuzzy ARTMAP [7] que podemos considerar como versiones difusas de ART 1 y ARTMAP respectivamente. Estos modelos se han venido aplicando, básicamente, a problemas de clasificación. Para problemas de identificación de funciones se han propuesto otra serie de modelos, basados en mayor o menor medida en las arquitecturas ART, entre los que podríamos mencionar las redes MIN-MAX [14], los modelos ART-EMAP [16], RFALCON [12], PROBART [13], FasArt [3] y FasBack [4].

En el presente trabajo veremos una somera descripción de las arquitecturas Fuzzy ART y Fuzzy ARTMAP así como de los modelos PROBART, FasArt y FasBack, describiendo sus principales características para centrarnos en el problema del aprendizaje cuando los patrones, disponibles para ello, presentan un cierto nivel de ruido. Haremos un estudio comparativo del comportamiento de estos modelos ante este problema y veremos como FasArt y FasBack presentan buenas características para abordar este problema.

## 2 La arquitectura Fuzzy ART

La arquitectura Fuzzy ART incorpora ideas propias de la teoría de conjuntos difusos en la arquitectura ART 1. Para ello recurre a reemplazar los operadores unión e intersección entre conjuntos por sus homónimos referidos a conjuntos difusos.. Este cambio permite a Fuzzy ART pasar, de operar sólo con entradas binarias, como hacía ART 1, a poder procesar datos de tipo analógico, en particular vectores de entrada cuyos componentes esten comprendidos en el intervalo $[0, 1]$.

Fuzzy ART mantiene las características propias de las arquitecturas ART: Comparación entre nuevas entradas y prototipos almacenados, estado de resonancia que asegura la estabilidad, búsqueda paralela y Autoorganización.

## 3 Aprendizaje Supervisado: Fuzzy ARTMAP

Como ya se ha hecho referencia anteriormente, ARTMAP es la arquitectura de la teoría ART que fue propuesta para efectuar aprendizaje de tipo supervisado. Fuzzy ARTMAP es la versión difusa de ARTMAP y está compuesta por dos módulos de tipo Fuzzy ART y una memoria asociativa, denominada mapa Inter-ART. El mecanismo de funcionamiento de Fuzzy ARTMAP es análogo al de ARTMAP En la fase de aprendizaje se presentan dos vectores que deben ser asociados, en ARTa se presenta la entrada y en ARTb la salida asociada a dicha entrada. Estos vectores activan sendas unidades del nivel F2 de ARTa y ARTb que son relacionadas mediante el mapa Inter ART. Mediante el mecanismo denominado Reset Inter ART se vigila que la generación de nuevas relaciones en el mapa Inter ART no lleve a una contradicción con las ya almacenadas. En la fase de predicción se introduce un único vector en ARTa que, a través del mapa Inter ART, activará una unidad en ARTb, cuyos pesos corresponderán a la salida predicha por la red.

## 4 El Modelo FasArt

Como hemos visto en el modelo Fuzzy ART se justifica la inclusión de conjuntos difusos mediante la analogía entre operaciones lógicas entre conjuntos con elementos binarios y operaciones logico-difusas entre conjuntos con elementos en el intervalo $[0, 1]$. No obstante, en los trabajos mencionados no queda clara la construcción de los conjuntos difusos (entrada y pesos). Un conjunto difuso se caracteriza mediante la definición de una función de pertenencia que lo defina. En Fuzzy ART no se da ninguna definición de las funciones de pertenencia de los conjuntos difusos asociados a las categorías.

En FasArt [3] se propone una nueva función de activación de las unidades ante una entrada dada. Esta nueva función representa, a su vez, una función de pertenencia a los conjuntos difusos definidos por los pesos de las unidades. Establecemos por tanto una dualidad función de pertenencia/activación. La expresión de esta nueva función viene dada por la expresión: $\eta_{R_k} = \prod_{i=1}^{M} \eta_{ki}(I_i)$ donde la función $\eta_{ki}$ presenta la siguiente expresión:

$$\eta_{ki}(x_i) = \begin{cases} \max\left(0, \frac{\gamma(I_i - w_{ki})+1}{\gamma(c_{ki} - w_{ki})+1}\right) & \text{si } x \leq c_{ki} \\ \max\left(0, \frac{\gamma(1-I_i - w^c_{ki})+1}{\gamma(1-c_{ki} - w^c_{ki})+1}\right) & \text{si } x > c_{ki} \end{cases}$$

El parámetro $\gamma$ determina el caracter más o menos difuso del sistema. Un valor de $\gamma \to 0$ hace que los conjuntos tengan un caracter más difuso, mientras que un valor $\gamma \to \infty$ produce un sistema con conjustos con un mayor caracter "*crisp*".

La mencionada dualidad función de pertenencia/activación nos permite encontrar una interpretación del funcionamiento de FasArt utilizando reglas, cuyos antecedentes y consecuentes son conjuntos difusos, del tipo: SI $I$ ES $R_j$ ENTONCES $Y$ ES $R_k$, donde el vector $\vec{I}$ se corresponde con el vector de entrada y $R_k$ representa los conjuntos difusos asociados a los pesos de las unidades del nivel F2.

Esta interpretación de FasArt como un sistema lógico difuso nos permite utilizar un método de defuzzificación para calcular la salida. Utilizando el método de defuzzificación basado en la media de los centros de los conjuntos difusos [15].

$$y_m(\vec{I}) = \frac{\sum_{l=1}^{N^b} \sum_{k=1}^{N^a} c^b_{lm} w^{ab}_{ki} \prod_{i=1}^{M^a} \eta_{R_{ki}}(I_i)}{\sum_{l=1}^{N^b} \sum_{k=1}^{N^a} w^{ab}_{ki} \prod_{i=1}^{M^a} \eta_{R_{ki}}(I_i)}$$

Esta expresión se corresponde con la salida de un sistema lógico difuso que utiliza inferencia mediante producto, fuzzificación mediante punto único y el mencionado método de defuzzificación basado en la media de los centros [15]. Utilizando los resultados sobre aproximación de funciones mediante sistemas difusos [10], [11], podemos concluir que la arquitectura FasArt es capaz de aproximar uniformente cualquier función $f$ continua definida sobre un compacto $U$, $f : U \subseteq \Re^M \to V \subseteq \Re$.

FasArt se ha utilizado en problemas de identificación de sistemas no lineales [3], [1] y para la construcción, de forma automática, de controladores difusos en procesos bioquímicos, en particular para la producción de penicilina [2]

## 5 El Modelo FasBack

El modelo FasBack pretende minimizar el número de unidades y por tanto la complejidad de la red. Para ello introduce los conceptos de aprendizaje mediante minimización del error de predicción en el modelo FasArt. Nos planteamos un sistema de aprendizaje que nos asegure la minimización matemática de los errores de predicción: $\vec{e_i} = (\vec{y_i} - \vec{d_i})$, donde $y_i$ representa la salida del sistema y $d_i$ la salida deseada.

Para cumplir este objetivo nos planteamos la minimización de un índice que tiene por expresión: $I = \frac{1}{2}\sum \vec{e_i}^2 = 1/2\sum(\vec{y_i} - \vec{e_i})^2$.

Nuestro objetivo se centrará en la elección de los parámetros adecuados del sistema que hagan que este índice sea mínimo. Este es un problema de optimización que se puede abordar con cualquiera de los métodos propuestos en la bbliografía para resolver problemas de optimización de indices cuadráticos para sistemas no lineales. Nosotros vamos a utilizar el método de gradiente descendente que se basa en la modificación de los parámetros en la dirección que marca la derivada del índice de error con respecto al vector de parámetros [15]. Este método se plasma en una ecuación de aprendizaje del tipo:

$$\vec{p}(k) = \vec{(k-1)} - \alpha(\vec{y} - \vec{d}) \left.\frac{\partial \vec{y}}{\partial \vec{p}}\right|_{k-1}$$

donde $p(k)$ representa el vector de parámetros respecto al que se está optimizando en el instante de tiempo $k$, $0 \leq \alpha \leq 1$ un escalar que representa la ratio de aprendizaje y $e_p$ el error de predicción.

Partimos de un sistema FasArt y nos planteamos la minimización del error de predicción. Para ello, en primer lugar, nos debemos plantear los parámetros que podemos modificar en FasArt durante la fase de aprendizaje. La adicción de nuevos elementos al vector de parámetros supone un aumento de los grados de libertad del sistema lo que debiera incidir en un menor error de predicción. En particular en FasBack se ha considerado como vector de parámetros en vector: $\vec{p} = (c^b_{jm}, w^{*ab}_i, c^a_{in})$. Las expresiones para calcular las leyes de aprendizaje de los mismos pueden verse recogidas en [4].

## 6 Datos de Aprendizaje con Ruido

Un problema habitual que se nos presenta cuando trabajamos con datos experimentales es la presencia de un cierto nivel de ruido en los mismos. Este problema tiene diferentes causas que van desde las perturbaciones hasta el rango de precisión de los intrumentos de medida. En una frase con pretensiones filosóficas podríamos decir que la imprecisión es el tributo que debemos pagar cuando nos alejamos del mundo de las ideas y nos aproximamos al mundo de la realidad. Los sistemas basados en conjuntos difusos se presentan como candidatos idóneos para trabajar con este tipo de información ya que establecen fronteras borrosas entre categorías o valores. El concepto de número borroso resulta de gran interés a la hora de reprensentar valores que llevan asociado un determinado grado de incertidumbre.

El modelo PROBART [13] se ha propuesto como alternativa para trabajar en este tipo de condiciones de ruido en los datos de aprendizaje. Por ello vamos a utilizarlo como punto de comparación con FasArt y FasBack en problemas de identificación de funciones con ruido en los patrones de entrenamiento.El modelo PROBART sigue básicamente la misma arquitectura y el mismo algoritmo de funcionamiento que el modelo Fuzzy ARTMAP. Su principal diferencia se centra en la ley de aprendiazaje para los pesos del mapa Inter ART. En PROBART se pretende que en estos pesos se vaya reflejando el número de veces que una determinada asociación se ha activado debido a un patrón de aprendizaje.

El test planteado en [13] consiste en identificar la función: $f(x) = 1/20\left(\sum_{n=1}^{n=7} \sin(10nx) + 10\right)$

El rango de test para $f : \Re \rightarrow \Re$ es $f(x) \in [0.2295, 0.7705]$ para un dominio de entrada $x \in [0, 1]$. Para generar los patrones con ruido se añade al patrón original un ruido de tipo Gaussino de forma que: $y_p = f(x_p) + 0.002\epsilon_p$ donde $\epsilon_p$ es un ruido Gausino $\epsilon_p \sim \mathcal{N}(0, 1)$

Siguiendo la metodología descrita en [13], en una primera fase de la experimentación se procedió a generar 1000 patrones de aprendizaje sin ruido eligiendo aleatoriamente los puntos $x_p$. En la fase de aprendizaje con FasArt y con FasBack se utilizaron los parámetros: $\rho_a = 0.5$, $\rho_b = 0.9$, $\gamma_a = \gamma_b = 50$. Para la fase de test se escogieron otros 1000 patrones de forma aleatoria. Utilizando los resultados recogidos en [13] para el mismo experimento y en las mismas condiciones utilizando Fuzzy ARTMAP y PROBART con unos valores de parámetros: $\alpha = 0.001$, $\rho_a = 0.99$, $\rho_b = 0.99$ y $\rho_{ab} = 0.9$, hemos elaborado la tabla 1, donde Na y Nb representan el número de nodos que se han necesitado en ARTa y ARTb respectivamente. En la tabla se aprecia claramente la mayor capacidad de FasArt y de FasBack a la hora de identificar la función, con respecto a Fuzzy ARTMAP se consiguen mejores niveles de error cuadrático con una complejidad de red (número de nodos) muy inferior. Con respecto a PROBART, y con una complejidad de red similar, se consiguen resultados de error claramente mejores.

Tabla 1: Comparación de resultados con aprendizaje utilizando para el aprendizaje datos sin ruido

| Modelo | Na | Nb | RMS |
|---|---|---|---|
| ARTMAP | 312 | 53 | 0.0074 |
| PROBART | 110 | 53 | 0.0169 |
| FasArt | 130 | 29 | 0.0066 |
| FasBack | 133 | 29 | 0.0035 |

Como segundo experimento se plantea utilizar como vectores de aprendizaje datos con ruido. Para ello generamos un nuevo conjunto de aprendizaje de 1000 vectores utilizando la ecuación con ruido añadido que vimos anteriormente. En la figura 1 podemos ver una representación de los nuevos datos de aprendizaje. Una vez efectuado el aprendizaje realizamos el test utilizando un nuevo conjunto formado por 1000 vectores escogidos aleatoriamente y que se suponen libres de ruido.

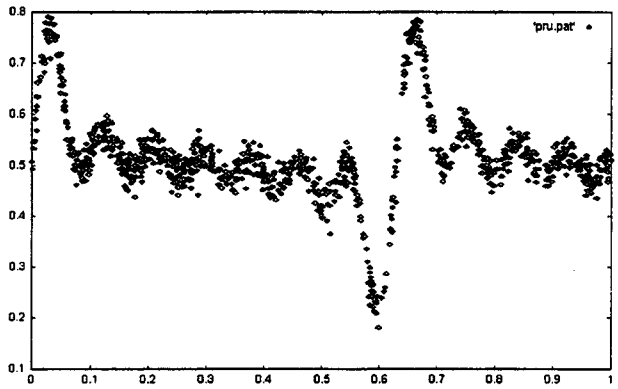



Figura 1: Patrones de aprendizaje con ruido.

En las figuras 2 y 3 puede verse el resultado del test, así como el error de predicción, cuando se utiliza el modelo FasArt y FasBack respectivamente, utilizando los mismos valores de los par metros $\rho_a$, $\rho_b$, $\gamma_a$, $\gamma_b$ que en el caso de aprendizaje con señal sin ruido. Se aprecia que, aunque la calidad de los datos de aprendizaje era deficiente tal y como veíamos en la figura 1, FasArt y FasBack son capaces de representar, con bastante precisión, la función que estamos identificando. Esta propiedad resulta de gran interés si consideramos que, para posibles aplicaciones a sistemas reales de FasArt y FasBack, los datos provinientes de sensores suelen presentar niveles de ruido apreciables. Un sistema que presente insensibilidad ante el ruido evita que se tenga que plantear la utilización de etapas de preprocesado de datos cuya definición no es siempre sencilla e inmediata.

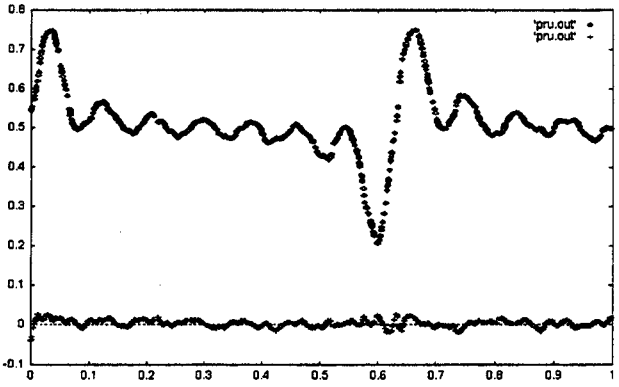



Figura 2: Resultado del test y error de predicción utilizando FasArt. Aprendizaje usando datos con ruido.

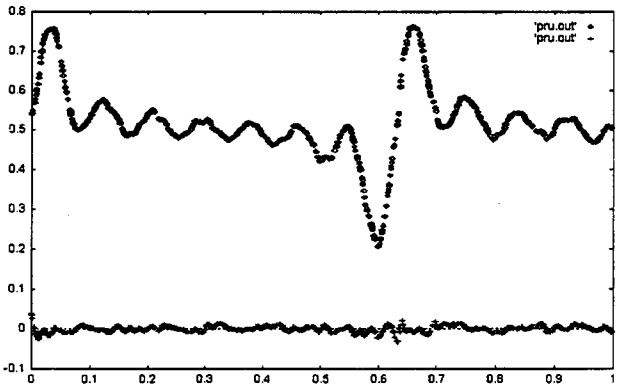



Figura 3: Resultado del test y error de predicción utilizando FasBack. Aprendizaje usando datos con ruido.

Para comparar el funcionamiento de FasArt y FasBack con el de Fuzzy ARTMAP y PROBART se han recogido nuevamente los resultados expuestos en [13]. Para el mismo experimento que hemos relatado y manteniendo los parámetros de $\alpha$, $\rho_a$, $\rho_b$, $\rho_{ab}$ que se habían fijado en el caso de aprendizaje con patrones libres de ruido, la tabla 2 recoge los resultados obtenidos con los diferentes modelos que hemos comentado. Los datos de error que vienen acompañados con la leyenda (A) se refieren a valores sobre el conjunto de aprendizaje, los que vienen acompañados por (CA) se refieren al caso en el que se ha utilizado como conjunto de test el mismo conjunto de aprendizaje pero con el valor de la función libre de ruido, se han evaluado los mismos puntos $x$ que para el conjunto de aprendizaje pero utilizando la fórmula de la función libre de ruido, estos datos reflejarán la capacidad de los diferentes modelos para "filtrar" el ruido de los datos de aprendizaje. La leyenda (T) se refiere al conjunto de test, los valores a los que acompaña esta etiqueta medirán, por tanto, la capacidad de generalización de la red a partir de los datos aprendidos.

Tabla 2: Comparación de resultados con aprendizaje utilizando para el aprendizaje datos con ruido

| Modelo | Na | Nb | RMS (A) | RMS (CA) | RMS (T) |
|---|---|---|---|---|---|
| ARTMAP | 806 | 61 | 0.0137 | 0.0302 | 0.0302 |
| PROBART | 112 | 61 | 0.0322 | 0.0189 | 0.0202 |
| FasArt | 275 | 30 | 0.0206 | 0.0099 | 0.0097 |
| FasBack | 284 | 30 | 0.0198 | 0.0074 | 0.0078 |

Analizando los datos contenidos en la tabla 2 podemos ver como la presencia de ruido en el aprendizaje hace que aumente en Fuzzy ARTMAP de una forma considerable en número de nodos en ARTa. Esto es debido a que se dispara sistemáticamente el mecanismo de Reset Inter-ART de una forma incorrecta, ya que dos puntos que debieran estar en una misma categoría pasan a estar en distintas categorías debido al ruido. Este tipo de efectos explica el considerable aumento del número de nodos en ARTa que se produce al utilizar el modelo Fuzzy ARTMAP, de 312 nodos en el caso de aprendizaje sin ruido se pasa a 806 en el caso de aprendizaje con ruido, mientras que los nodos en ARTb permanecen practicamente constantes: 53 vs. 61. El pequeño incremento se explica por el aumento en el rango de variación de la variable supervisada debido a la adición del ruido. El modelo PROBART no tiene el problema de proliferación de categorías debido al efecto de Reset Inter-ART ya que, por definición, carece de dicho mecanismo. Vemos que el número de nodos en ARTa practicamente no varía, 110 para los patrones sin ruido y 112 para los patrones con ruido. El comportamiento de ARTb es semejante al del modelo Fuzzy ARTMAP, ya que en ambos casos el aprendizaje es autoorganizativo. Con respecto a FasArt y FasBack su comportamiento está más próximo al de Fuzzy ARTMAP que al de PROBART ya que en ambos modelos se mantiene el mecanismo de Reset Inter-ART y seguimiento del emparejamiento. No obstante el crecimento del número de nodos en ARTb es mucho menor debido a que estamos trabajando con valores del parámetro de vigilancia menores ($\rho_a = 0.99 = \rho_b$ para Fuzzy ARTMAP mientras que $\rho_a = 0.5$, $\rho_b = 0.9$ para FasArt y FasBack).

Con respecto a la variación de las características de funcionamiento, cuando hacemos el aprendizaje usando datos con ruido, podemos analizarlo si nos fijamos en los diferentes valores de los indices de error reflejados en las tablas 1, 2. Vemos que en el caso de Fuzzy ARTMAP el indice de error, RMS, de los datos de test ha empeorado ostensiblemente (de 0.0074 ha pasado a 0.0302) aunque el número de categorías generada ha aumentado, tambien vemos que el error

Tabla 3: Resultados con PROBART. Utilizando 10000 puntos de aprendizaje.

| Modelo | Na | Nb | RMS (A) | RMS (CA) | RMS (T) |
|---|---|---|---|---|---|
| PROBART | 608 | 341 | 0.0276 | 0.0079 | 0.0084 |

sobre los datos de aprendizaje (0.0137) resulta menor que el error sobre los datos de test ya sea en los puntos de aprendizaje sin ruido (RMS(CA)=0.0302) o en los puntos de test (RMS(T)=0.0302). Podemos decir que se ha producido un "sobreaprendizaje", Fuzzy ARTMAP ha aprendido demasiado "fielmente" de los datos de aprendizaje y no ha sido capaz de filtrar el ruido que éstos contenian y por lo tanto tampoco es capaz de generalizar sobre nuevos puntos.

En el caso de PROBART, aunque sube el error en el caso de aprendizaje con patrones con ruido (pasamos de 0.0169 a 0.0202), el aumento es de menor orden y observamos cómo el error es mayor en el conjunto de aprendizaje (RMS(A)=0.0322) que en los conjuntos de test, el que utiliza los puntos de aprendizaje sin ruido (RMS(CA)=0.0189) y el de nuevos puntos (CA(T)=0.0202. Esto nos indica que PROBART es capaz de filtrar el ruido del conjunto de aprendizaje y, aunque no sigue de forma exhaustiva los datos del mismo, puede aprender mejor la señal básica que los había generado. FasArt y FasBack presentan un comportamiento análogo manteniendo buenas características de error en los conjuntos de test, aunque en ellos se produzca un aumento en las categorías generadas en ARTb.

Para mejorar los indices de error utilizando PROBART en [13] se propone un nuevo experimento que consiste en aumentar el número de vectores de aprendizaje, para ello se toman 10000 vectores de aprendizaje y se aumentan los valores de los parámetros de vigilancia a: $\rho_a = 0.998 = \rho_b$. En la tabla 3 se recogen los resultados que se obtienen en este caso. Como se puede comprobar se produce un aumento considerable en el número de categorías formadas, tanto en ARTa como en ARTb, siendo muy superior a las formadas utilizando FasArt o FasBack. El indice de error que se consigue es comparable con el obtenido utilizando FasBack cuando ha sido entrenado sólo con el conjunto de aprendizaje inicial de 1000 patrones.

Uno de los objetivos principales que motivaron el diseño de FasBack era la reducción del número de reglas necesarias, esto se conseguía gracias a que el aprendizaje guiado por el error permitía trabajar con parámetros de vigilancia menores y, por tanto, con un número menor de categorías. Con el ánimo de comprobar

Tabla 4: Resultados con FasBack. Utilizando 1000 puntos de aprendizaje.

| Modelo | Na | Nb | RMS (A) | RMS (CA) | RMS (T) |
|---|---|---|---|---|---|
| FasBack | 88 | 15 | 0.0209 | 0.0096 | 0.0098 |

si esta propiedad se sigue manteniendo cuando existe ruido en los patrones de aprendizaje planteamos un nuevo experimento utilizando el conjunto de 1000 vectores de aprendizaje y con los parámetros: $\rho_a = 0.5$, $\rho_b = 0.9$, $\gamma_a = \gamma_b = 50$. En la tabla 4 podemos ver los resultados obtenidos. Se ha producido una reducción importante en el número de categorías, y por lo tanto de la complejidad de la red, pero los valores de error se mantinen en el mismo orden, incluso que el mejor resultado referenciado para PROBART con un menor número menor de vectores de aprendizaje.

## 7 Conclusiones

Como conclusión del trabajo podemos recalcar el hecho de que FasArt y FasBack son capaces de trabajar con patrones de aprendizaje que contengan ruido ya que, gracias a su representación de las categorías en forma de conjuntos difusos, filtran el ruido de los patrones quedándose con la información esencial que contienen. En el test propuesto en [13] hemos comprobado que proporcionan mejores indices de funcionamiento que una arquitectura diseñada especificamente para intentar resolver este tipo de problemas.

## Referencias


[1] J. Cano, Y. Dimitriadis, M. Arauzo, F. Abajo, and J. Coronado. Fuzzy adaptive system ART-based: Theory and application to identification of biochemical systems. In *Proceedings of CESA'96*, Lille, France, 1996.

[2] J. Cano, Y. Dimitriadis, M. Arauzo, F. Abajo, and J. Coronado. A neuro-fuzzy architecture for automatic development of fuzzy controllers. In *Proceedings of CESA'96*, Lille, France, 1996.

[3] J. Cano, Y. Dimitriadis, M. Arauzo, and Coronado J. FasArt: A new neuro-fuzzy architecture for incremental learning in systems identification. In *Proceedings of IFAC'96*, San Francisco, USA, 1996.

[4] J. Cano, Y. Dimitriadis, and J. Coronado. FasBack: Matching-error based learning for automatic generation of fuzzy logic systems. In *Proceedings of FUZZ-IEEE'97*, Barcelona, Spain, 1997.

[5] G. Carpenter and S. Grossberg. ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26(23):4919–4930, December 1987.

[6] G. Carpenter and S. Grossberg. Self-organizing neural network architectures for real-time adaptive pattern recognition. In *An Introduction to Neural and Electronic Networks*, pages 455–478. Springer-Verlang, 1989.

[7] G. Carpenter, S. Grossberg, N. Markuzon, J. Reynolds, and D. Rosen. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, 3(4):698–713, September 1992.

[8] G. Carpenter, S. Grossberg, and J. Reynolds. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4(5):565–588, 1991.

[9] G. Carpenter, S. Grossberg, and D. Rosen. Fuzzy ART: Fast estable learning and categorization of analog patterns by a adaptive resonance system. *Neural Networks*, 4:759–771, 1991.

[10] J. Castro. Fuzzy logic cantrollers are universal approximators. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(4):629–635, April 1995.

[11] B. Kosko. Fuzzy function approximation. In *IJCNN'92*, Baltimore, June 1992.

[12] C. Lin and C. Lin. Reinforcement learning for an ART-based fuzzy adaptive learning control network. *IEEE Transactions on Neural Networks*, 7(3):709–730, May 1996.

[13] S. Marriott and R. Harrison. A modified fuzzy ARTMAP architecture for the approximation of noisy mappings. *Neural Networks*, 8(4):619–641, 1995.

[14] P. Simpson. Fuzzy MIN-MAX neural network for function approximation. In *ICNN'93*, San Francisco, 1993.

[15] L. Wang. *Adaptive Fuzzy Systems and Control.* Prentice Hall, 1994.

[16] J. Williamson. Gaussian ARTMAP: A neural network for fast incremental learning of noisy multidimensional maps. *Neural Networks*, 9(5):881–897, 1996.

[17] L. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, June 1965.

# ENTRENAMIENTO DE REDES DE BASE RADIAL REALIMENTADAS CON FILTROS FIR

**Luciano Boquete Rafael Barea Ricardo García Manuel Mazo**
Departamento de Electrónica.
Universidad de Alcalá. 28801.Madrid
E-mail: boquete@depeca.alcala.es

## Resumen

Se describe un nuevo modelo de red neuronal de base radial, en donde cada una de las neuronas está realimentada por un filtro FIR, siendo por tanto una red recurrente. El proceso de entrenamiento de la red incluye el ajuste de las conexiones sinápticas entre las neuronas y la salida de la red, y también el ajuste de los coeficientes de cada uno de los filtros. En comunicaciones anteriores se han obtenido las fórmulas para el entrenamiento de la red utilizando un algoritmo de descenso por el gradiente, y en la presente, se hace un estudio utilizando un algoritmo estocástico: ALOPEX. Ambas técnicas de ajuste son muy diferentes, pero cada una de ellas tiene una serie de ventajas que pueden hacer interesante su utilización en determinadas circunstancias.

**Palabras Clave:** Red de base radial, FIR/IIR, identificación de sistemas.

## 1 INTRODUCCIÓN

Una de las posibilidades para identificar el comportamiento de un sistema consiste en utilizar redes neuronales [1][6]. Si el sistema físico a identificar tiene memoria (la salida en un momento dado depende de entradas y salidas anteriores) es preciso utilizar una red recurrente, para minimizar el error de identificación. Existen diversos modelos de redes recurrentes, generalmente multicapa: Jordan, Elman, entradas retardadas en el tiempo, con memoria asociada a la neurona [7], etc. Uno de los problemas de la utilización de redes recurrentes es que el tiempo de entrenamiento puede llegar a ser muy elevado. Por este motivo, una posibilidad consiste en utilizar redes de base radial (RBF), que están formadas por una única capa intermedia. Con el fin de que una RBF tengan en cuenta estados anteriores, se pueden realizar diversas modificaciones a su arquitectura: introducir entradas de instantes anteriores, modelo de Chng[4], con conexiones sinápticas mediante filtros FIR/IIR[5], etc.

En este artículo se propone una nueva definición de neurona recurrente en donde la salida de la neurona (normalmente una función gaussiana) se realimenta hacia su propia entrada, sumándose a la entrada de la red neuronal. Esta realimentación se realiza mediante un filtro FIR, para que sea posible memorizar estados anteriores. Utilizando un número suficiente de unidades de estas características, es posible construir una red neuronal capaz de identificar sistemas dinámicos con retardos. En [2] se han obtenido las expresiones para el ajuste de las conexiones sinápticas y de los coeficientes de todos los filtros mediante un algoritmo de descenso por el gradiente, comprobando su funcionamiento a la hora de identificar diferentes plantas. Una de las ventajas de utilizar un algoritmo de descenso por el gradiente, es la rapidez con que se consigue disminuir el error entre la salida de la planta y la red neuronal.

Esta comunicación se ha estructurado en los siguientes apartados: en 2 se describe la arquitectura de una red de neuronas realimentadas con un filtro FIR; el apartado 3 está dedicado a la exposición del algoritmo ALOPEX; en el apartado 4 se presentan diferentes simulaciones realizadas y por último, en el apartado 5 se comentan las principales conclusiones del presente trabajo.

## 2 RED NEURONAL

En la figura 1 se muestra la arquitectura de una red neuronal formada por M neuronas realimentadas con filtros FIR. La salida de la red está compuesta por la suma de las salidas de todas las neuronas, ponderadas cada una de ellas con un peso $w_i$.

$$y_N(k) = \sum_{i=1}^{M} w_i \cdot g_i(k) \qquad (1)$$

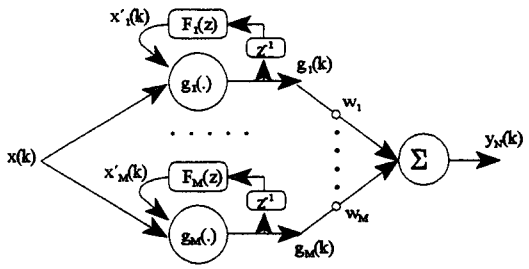



Figura 1: Modelo de red neuronal.

A su vez, la salida de cada neurona en el instante k viene dada por:

$$g_i(k) = e^{-\frac{[\,x(k) + x_i'(k) - C_i\,]^2}{\sigma^2}}; \qquad i = 1,...M; \tag{2}$$

$x_i'$ es la salida del filtro que conecta la salida de cada neurona con su propia entrada. Su expresión es:

$$x_i'(k) = \sum_{j=1}^{N} a_{ij} \cdot g_i(k-j) \qquad i = 1,...M; \tag{3}$$

Por simplicidad, se supone que los filtros de las diferentes neuronas tienen el mismo número de retardos (N).

Si se utiliza un algoritmo de descenso por el gradiente para el entrenamiento de esta red neuronal, normalmente la función de error que se pretende minimizar es cuadrática:

$$E_C = \frac{1}{2}.(y_d - y_N)^2 \tag{4}$$

siendo $y_d$ la salida deseada para la red. Con el algoritmo ALOPEX, es posible utilizar otras funciones de error, como por ejemplo:

$$E_R = \frac{1}{2}.\sqrt{(y_d - y_N)^2} \tag{5}$$

$$E_E = y_d.\log(\frac{y_d}{y_N}) \tag{6}$$

Dependiendo de la planta que se pretende identificar, puede funcionar mejor un tipo de error u otro.

Con respecto a los centros de las funciones gaussianas, éstos se distribuyen uniformemente en todo el margen de los valores de entrada, desde el valor máximo de x(k) ( $M_{x(k)}$ ) hasta su valor mínimo: $m_{x(k)}$. El parámetro σ toma el mismo valor para todas las neuronas y se calcula por la expresión:

$$\sigma = \frac{M_{x(k)} - m_{x(k)}}{2M} \tag{7}$$

## 3 ALGORITMO ALOPEX

El algoritmo ALOPEX es un proceso de optimización, que ha sido utilizado con éxito en diferentes aplicaciones: eliminación de ruido en tiempo real, reconocimiento de patrones, sistemas de control adaptativo [9], etc. Es especialmente útil en aquellos casos que se quiera minimizar ( o maximizar ) una función de coste que depende de un número muy elevado de variables: $\theta_1...\theta_R$. Si R es muy elevado, se necesita realizar una búsqueda exhaustiva hasta encontrar una solución subóptima. Este algoritmo actualiza los parámetros simultáneamente, utilizando la correlación entre las variaciones en los parámetros $\Delta\theta_i$ y la variación en la función objetivo ΔE .

Las principales ventajas con respecto a un algoritmo de descenso por el gradiente son:

- Se puede utilizar cualquier función de error. Así se puede elegir aquella que tenga menos mínimos locales.

- Soporta menos carga computacional que otros algoritmos.

- Su implementación hardware es inmediata.

- Es válido para identificar sistemas con memoria.

- Se puede utilizar cualquier función de activación en las neuronas.

- La función de error a minimizar no tiene por qué estar definida a la salida de la red neuronal.

Se considera una red neuronal con parámetros de ajuste $\theta_i$ (en nuestro caso los coeficientes de los filtros y las conexiones sinápticas). En cada iteración del entrenamiento se produce el siguiente ajuste:

$$\theta_i(k+1) = \theta_i(k) + \delta(k+1); \tag{8}$$

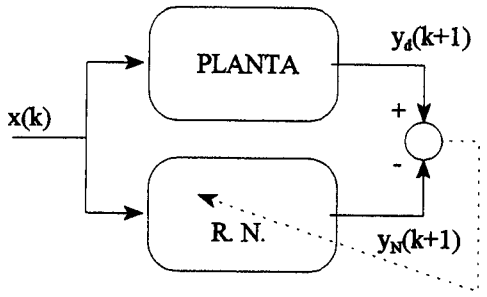



Figura 2: Identificación paralelo-paralelo

Siendo δ(k+1) un número de valor pequeño, que toma el valor +δ con probabilidad P(k+1) o el valor -δ con probabilidad [1 - P(k+1)]. A su vez, la probabilidad viene dada por:

$$P_i(k+1) = \frac{1}{1 + e^{\frac{-\Delta_i(k+1)}{T}}}; \qquad (9)$$

$\Delta_i$ es la correlación entre la variación en el parámetro $\theta_i$ y la función objetivo [E(k+1)], y T es la temperatura que determina el grado de aleatoriedad del algoritmo.

$$\Delta_i(k+1) = [\theta_i(k)-\theta_i(k-1)].[E(k)-E(k-1)] \qquad (10)$$

Se define la función objetivo como:

$$E(k+1) = [y_d(k) - y_N(k)]^2 \qquad (11)$$

El parámetro T es un valor positivo, que para valores no nulos, consigue que el error E(k+1) disminuya.

En un principio, T toma valores elevados ( el espacio de búsqueda de una solución es muy amplio ), y en sucesivos ciclos de aprendizaje, se hace igual al valor medio de $\Delta_i$ . Así se reduce el espacio de búsqueda cuando el proceso está cerca de encontrar un mínimo.

En el caso presentado, se introduce una entrada a la red neuronal, y se propaga la señal hasta obtener el valor de salida; con este dato y la salida deseada se calcula el error en el instante k y se aplica el procedimiento anterior a todas las conexiones sinápticas ( $w_i$ ) y a los coeficientes de los diferentes filtros $a_{ij}$.

# 4 SIMULACIONES

Con el fin de comprobar el funcionamiento, tanto del modelo de red neuronal propuesto como de su entrenamiento por medio del algoritmo ALOPEX, se han utilizado los modelos descritos en [8]:

$$y_d(k+1) = \frac{y_d(k).y_d(k-1).[y_d(k) + 2,5].[y_d(k) - 1]}{1 + y_d^2(k) + y_d^2(k-1)} + x(k) \qquad (12)$$

$$y_d(k+1)=\frac{y_d(k).[y_d(k)+0,3]}{1+y_d^2(k)}+x(k).[x(k)+0,8].[x(k)-0,5] \qquad (13)$$

Para ambos casos, se ha utilizado la siguiente señal de entrada:

$$\begin{aligned} x(k) &= \sin(2.\pi.k/250); \qquad 0 \le k \le 500; \\ x(k) &= 0.8.\sin(2.\pi.k/250)+ 0.2.\sin(2.\pi.k/25); \qquad k > 500; \end{aligned} \qquad (14)$$

Dos son las principales diferencias entre ambos trabajos: en [8] se definen cuatro modelos diferentes de plantas, y para cada uno de esos modelos, se utiliza un esquema diferente de identificación, por ejemplo, paralelo-paralelo, paralelo-serie, en otro caso se realimentan las salidas de la red hacia la entrada o se realimenta la salida de la planta hacia la entrada de la red. El problema que tiene este método es que se debe conocer el tipo de planta que se pretende identificar y a continuación utilizar el esquema apropiado, que puede llegar a ser relativamente complicado. En el modelo neuronal propuesto en este trabajo, siempre se utiliza un esquema de identificación paralelo-paralelo ( Fig. 2 ) , con independencia del tipo de planta, por tanto, no es preciso tener ningún conocimiento previo de la misma.

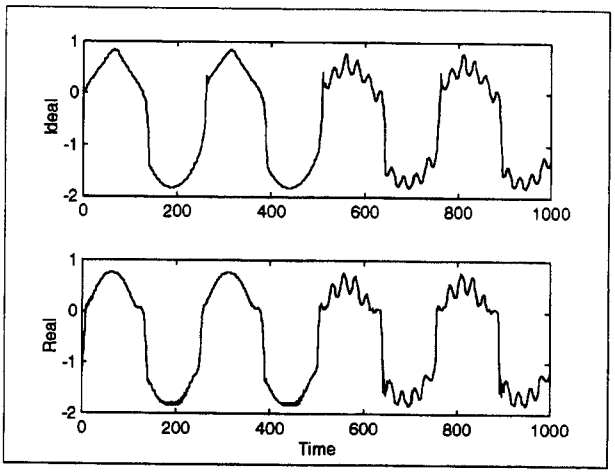



Figura 3: Identificación de la ecuación 12

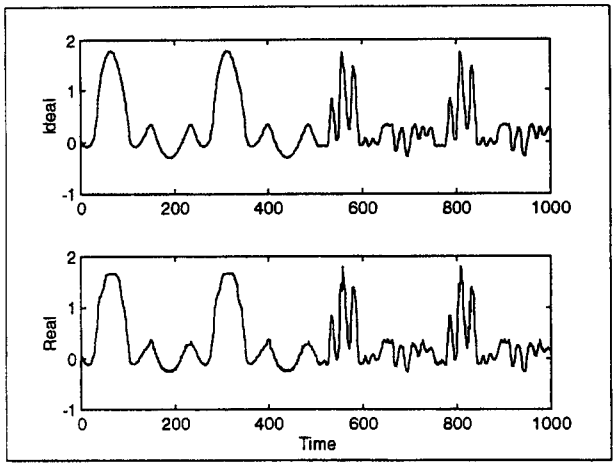



Figura 4: Identificación de la ecuación 13.

Como puede observarse ( Figuras 3 y 4), se consigue un alto grado de coincidencia entre la señal real y la señal ideal. Los ejemplos mostrados corresponden con la salida de la red neuronal en el ciclo 5.000 de entrenamiento, con lo que se puede indicar que este método no puede ser utilizado para entrenamiento "on-line".

En las figuras indicadas, se ha utilizado una red neuronal con 5 neuronas y el valor del parámetro $\delta$ es de 0,001. A efectos de comparación, en la figura 5 se presenta el resultado de la identificación del sistema con ecuación 12 después de 10.000 ciclos de entrenamiento, utilizando el mismo conjunto de parámetros.

## 5 CONCLUSIONES

Se ha implementado un nuevo modelo de red neuronal recurrente, y se ha estudiado el comportamiento del algoritmo ALOPEX en el ajuste de sus parámetros. Este

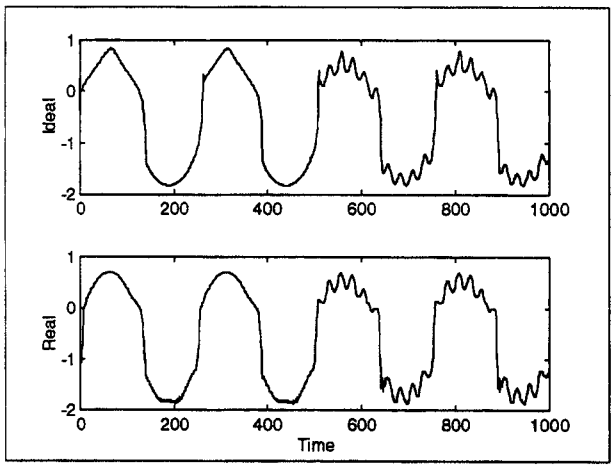



Figura 5: Identificación de la ec. 12. ( 10.000 ciclos)

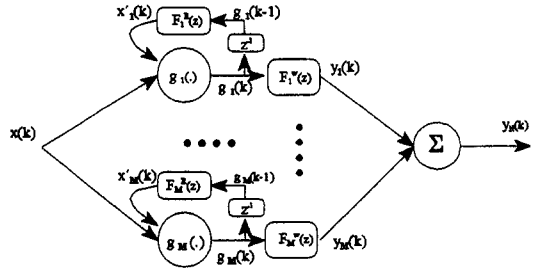



Figura 6: Red con filtros FIR como conexiones sinápticas

algoritmo destaca por su facilidad de implementación (se aplica indistintamente a las conexiones sinápticas y a los coeficientes de los filtros), el reducido número de operaciones a realizar en cada ciclo de trabajo y que no es necesario propagar errores hacia atrás.

La principal ventaja de la arquitectura propuesta como identificador de sistemas consiste en que siempre se utiliza el mismo esquema de identificación, "paralelo-paralelo", sin necesidad de utilizar retardos para las entradas o salidas de la planta.

Como inconveniente se puede comentar la necesidad de realizar un número elevado de ciclos de entrenamiento hasta conseguir errores pequeños en la identificación.

Otras líneas de trabajo están encaminadas al entrenamiento de redes de las mismas características, pero utilizando como conexiones sinápticas otros filtros FIR (Fig. 6). De igual forma que el caso presentado, el ajuste se puede realizar mediante una técnica de descenso por el gradiente, pero puede ser interesante también la utilización de un algoritmo estocástico.

Una aplicación inmediata de las ideas expuestas es la implementación de un esquema de control directo, mediante una red neuronal que actúe como controlador (Fig. 7). Puesto que no es necesario que la función de error se defina a la salida de la red neuronal, se evita el proceso de propagar los

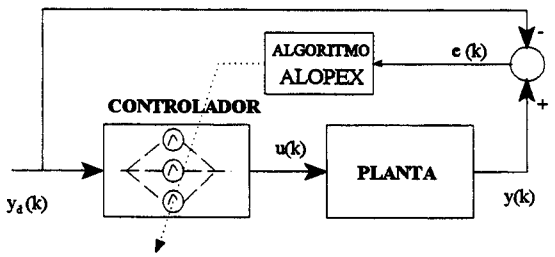



Figura 7: Ajuste de un controlador mediante el algoritmo Alopex.

errores desde la salida de la planta hacia su entrada [3].

**Agradecimientos**

Los autores desean expresar su gratitud a la Comisión Interministerial de Ciencia y Tecnología (CICYT), por la ayuda recibida en los proyectos TAP94-0656-C03-01 y TER96-1957-C03-01.

**Referencias**

[1] A.G. Barto (1989). Connectionist Learning for Control: An Overview. *COINS Technical Report* 89-89.

[2] L. Boquete, R. Barea, R. García, M. Mazo, J. A. Bernad (1997). Identification of Systems Using Radial Basis Networks Feedbacked with FIR filters. *5 Fuzzy Days in Dortmund. Lectures Notes in Computer Science. Springer Verlag.*

[3] L. Boquete, R. Barea, R. García, M. Mazo, L M. Bergasa (1997). Utilización de un Algoritmo Estocástico para el Control Neuronal Adaptativo de un Sistema no Lineal. *SAAEI'97. Seminario Anual de Automática, Electrónica Industrial e Instrumentación.* 17-19 Sep. Valencia.

[4] E. S. Chng, S. Chen and B. Mulgrew (1994). Improving the Radial Basis Function Networks for Homogeneous Nonstationary Time Series Prediction. *European Signal Processing Conference VII 94 Edinburgh,* Pág. 1819-1822

[5] I. B. Ciocoiu (1996). Radial Basis Function Networks with FIR/IIR Synapses. *Neural Processing Letters* Vol. 3, Pág. 17-22.

[6] K.S. Narendra, K. Parthasarathy (1991). Identification and Control of Dynamic Systems Using Neural Networks. *IEEE Transactions on Neural Networks,* Vol. 1, N° 1, 4-27

[7] P.S. Sastry, G. Santharam, K.P. Unnikrishnan (1994). Memory Neuron Networks for Identification and Control of Dynamical Systems. *IEEE Transactions on Neural Networks,* Vol. 5, N° 2 (1994)

[8] J. Theocharis, G. Vachtsevanos (1996). Adaptive Fuzzy Neural Networks as Identifiers of Discrete-Time Nonlinear Dynamic Systems. *Journal of Intelligent and Robotic Systems* Vol. 17 Pág. 119-168.

[9] K. P. Venugopal (1993). Learning in Connectionist Networks Using the Alopex Algorithm. *Tesis Doctoral.* Florida Atlanta University.

# Predicción de demanda por técnicas Neuro-Fuzzy. Basada en estructuras jerárquicas ARS (Autonomous Response System)

**I.Escoda**

Laboratori d'Enginyeria i Materials Electrònics
Universitat de Barcelona
escoda@iris1.fae.ub.es

**A.Sanz**

Departamento de Electrónica
Unversidad de Zaragoza
asmolina@mcps.unizar.es

**A.Herms**

Laboratori d'Enginyeria i Materials Electrònics
Uiversitat de Barcelona
herms@iris1.fae.ub.es

## Resumen.

En este trabajo se presenta una herramienta para el análisis y predicción del comportamiento de sistema complejos utilizando metodologías de soft computing. Se propone una arquitectura para construir sistemas de predicción sirviéndonod del modelo ARS (Autonomous Response System), de bloques ANS (Artificial Neural set), interconectados por objetos llamados Vínculos. Esta herramienta permite desarrollar diversos modelos predictivos de una manera sencilla, con el consiguiente análisis y explotación.

**Palabras claves**: Neuro-Fuzzy, ANS, ARS, predicción, genéticos, demanda.

## 1 Introducción.

En los problemas de predicción de comportamiento se pueden utilizar dos posibles métodos, la modelación externa y la interna. En la primera aproximación los datos históricos se consideran una serie numérica con ciertas periocidades y dependencias ocultas y se pretende obtener un aproximador funcional que permita predecir la serie de datos con independencia de la estructura interna del sistema que la genera. Este método es simple y permite una primera aproximación al problema. Per en sistemas complejos, e incluso en los sencillos, esta aproximación puede ser insuficiente. Urge entonces, disponer de una aplicación donde podamos plantear y testear los modelos basados en las estructuras internas de los sistemas. para obtener una buena descripción del sistema pues

Para los sistemas de los que se posee información sobre su estructura interna, en el métofdo de moedelización interna se describe el sistema como formado por diversos elementos que interaccionan entre si y con el exterior. El modelo completo se basa en la descripción de estos elementos, su comportamiento interno, las interacciones entre estos y la ponderación de estas interacciones. Utilizando esta aproximación jerárquica se puede convertir un sistema complejo en un conjunto de sistemas mas simples, pudiendo describir de forma mas sencilla estos.

La base de este trabajo se fundamenta en la utilización de las nuevas teorías y tecnologías que se conocen por soft computing [2-3] para la caracterización de sistemas basados en ARS y que son empleados en predicción de la demanda de productos por medio de una aproximación jerárquica y estructurada.

## 2 Descripción de ARS.

El modelo ARS resulta adecuado para la descripción de estos sistemas y de su estructura interna. Este paradigma fue propuesto en [11] y está inspirado en los trabajos del biólogo Richard Goldsmidt en 1992. Una descripción de este modelo y de las herramientas que lo utilizan puede encontrase en [12-13].El modelo ARS esta basado en una estructura jerárquica de objetos interconectados. Un conjunto de ANS (Artificial Neuronal Sets) unidos al entorno por medio de enlaces llamados Vinculos. En [12-13] se muestra un desarrollo del modelo de ANS basada una

implementación en librerías de clases C++.

ARS esta formado por ANS (Artificial Neuronal Systems). Los ANS pueden ser de tipo ANN (Artificial Neuronal Nets) o FNN (Fuzzy Neuronal Nets) [10-14]. Los ANS enlazan internamente y con el entorno exterior (WORLD) por medio del uso de Vínculos que son objetos para el intercambio de datos.

Una estructura ARS puede contener además bloques funcionales (FUN), véase figura. 1, cuyo objetivo es el de preparar los valores que serán las entradas a los ANS (FEN y ANN). También puede contener el bloque Comp que agrupa varios ANS del mismo o distinto tipo. Este tipo de bloques Comp permiten definir jerarquias de objetos.

El bloque funcional (FUN) esta compuesto por dos buffers. Uno es capaz de guardar los valores históricos de las variables comprendidos entre los periodos t y t-k, siendo el valor k configurable independientemente para cada buffer, K=1,2,...,n. El otro de los buffers guarda los valores que se utilizan para realizar análisis de tendencias.

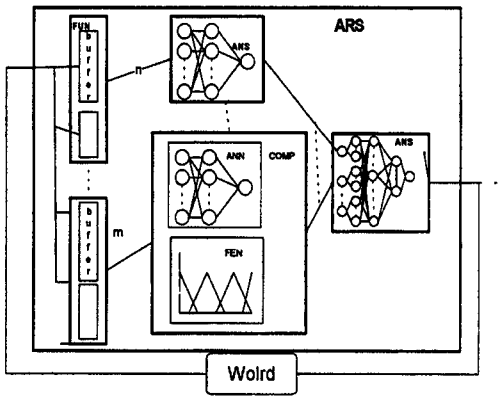



Figura 1. Estructura ARS

## 3 Arquitectura del sistema.

La estructura de la aplicación objeto de este trabajo se muestra en la fig.2. Esta estructura está formada por: bloques funcionales que se agrupan en el entorno predictivo y el entorno operativo.

El entorno predictivo lo forman la base de datos dinámica y un predictor. El objetivo del predictor es realizar predicciones del comportamiento del sistema utilizando datos históricos. Los componentes que los forman son:

1. GUI (Grafics User Interface) es unprograma desarrollado en Visual Basic, cuya misión es presentar un interface de comunicación entre el usuario y los diferentes entornos funcionales.

2. ARSAPI.DLL (ARS Aplication Interface) es el interface de comunicación con las librerías de software de ARS. Estas son librerías de enlace dinámico (DLL) de 32 bits desarrolladas en C++. Se ha obtado por un desarrollo de programación orientada a objetos, tomando como base los objetos de la librería de clases Microsoft Foundation Classes V 4.0.

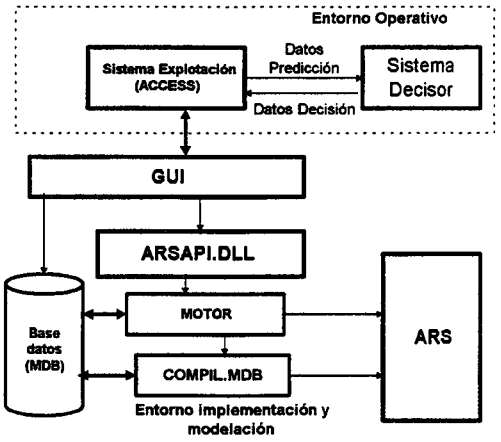



Fig.ura 2. Sistema de predicción basado en ARS

3. El bloque ARS, realiza la simulación de los m,odelos ARS que se le solicitan desde ARSAPI. Esta realizado en C++ utilizando como base la librería de Microsoft Foundation Classes V 4.0.

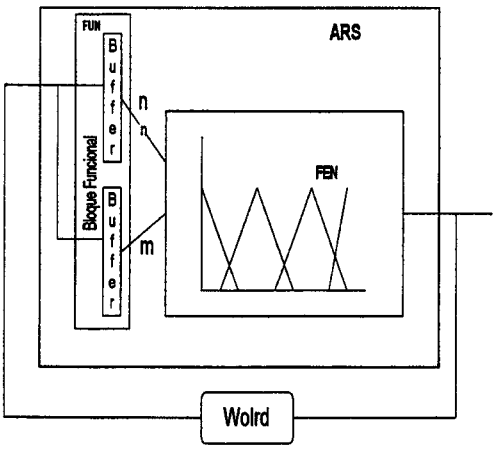



Figura 3. Estructura ARS

La base de datos ( MDB ) almacena las diferentes estructuras de datos y realiza una doble función; Almacena los modelos predictivos que se van desarrollando y los datos históricos que se utilizan para la predicción.

En la gráfica de la figura 3 puede verse la estructura del predictor ARS utilizada. Esta formada por un FUN, que realiza el preprocesamiento, y un FEN que realiza la pridicción. El bloque funcional (FUN) esta compuesto por dos buffers. Uno es capaz de guardar los valores históricos de las variables comprendidos entre los periodos t y t-k, siendo el valor k configurable independientemente para cada buffer, K=1,2,...,n. El otro buffers guarda los valores de los m últimas diferencias. El FEN (Fuzzy Equivalent Net) utiliza funciones de pertenecía de tipo triangular, reglas del tipo Sugeno y defucificación por método del centro de masa (COM).

Los parámetros que se emplean para caracterizar el FEN son:

Adjetivos_in (términos fuzzy) para cada una de las entradas al FEN.
Adjetivos_out, para las salidas del FEN.
Número de reglas para los FEN, por defecto 25.

El Sistema de Explotación basado en Access de Microsoft pretende ofrecer un entorno de comunicación y gestión del usuario con la base de datos que permita obtener información elaborada utilizando los datos históricos y las predicciones realizadas por los predictores desarrollados.

## 4 Desarrollo de modelos predictivos.

La creación del sistema predictivo se realiza en tres fases: fase de configuración de los modelos de ARS, fase de entreno o evolución y selección del modelo.

```
// pszSQL representa la consulta de la estructura
de datos Historicos.MDB que

//selecciona las entradas y salidas de cada
individuo.

//pszSQL ="SELECT Ve1,Vre2,Ve3,Ve4,Vs1,Vs1
FROM Historico"

// Entradas (4): Ve1,Ve2,Ve3,Ve4
```

**Fase de configuración y selección.** Por medio de GUI y utilizando la función ARS_Generar, del ARSAPI.DLL, el compilador MDB del bloque ARS, genera una población de individuos cuyos valores que toman en los periodos t, t-1, t-2,.....t-n, serán los mismos que se utilizan en la fase de predicción.

**Fase de selección y evolución.** Esta fase se realiza en dos pasos, en el primer paso se realiza un entrenamiento de la configuración de los ARS, por medio de la función ARSARS_EntrenarConfiguración, esta función permitirá el seleccionar un individuo de la población de ARS.

La estructura del algoritmo genético que se ha empleado para la evolucionar los predictores es la siguiente:

```
algoritmo _genético ( )
{
        Inicializar población;
        Evaluar población
        Ajustar función de aptitud;

        Realizar
                Selección de individuos;
                Cruzamiento y mutación;
                Evaluar población;
                Ajustar función de aptitud;
        Mientra no se alcanza el criterio de
        finalizar
                Seleccionar mejor
                individuo;
                Actualizar estructura con
                datos individuo;
                Guadar ultima generación
}
```

y los operadores ha emplear son:

Selección
Cruzamiento, tanto por uno de probabilidad de cruzamiento.
Mutación, tanto por uno de la probabilidad de mutación.

```
//Método: Algoritmos Genéticos con secuencia de operadores:

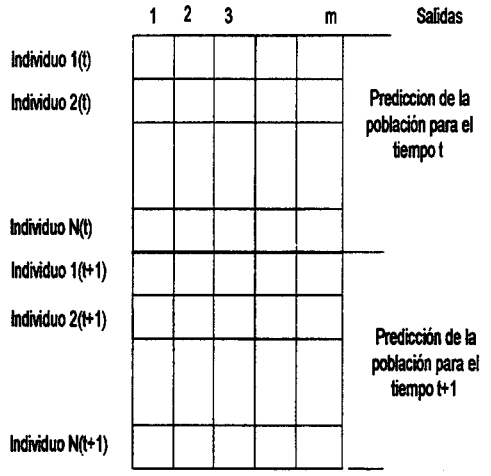
// selección, cruzamiento (tasa=75%), mutación (tasa=1%)

// pszMetodo = "GENETICOS SELECCION 0 CRUZAMIENTO 75 MUTACION 1"

// La cadena pszSQL representa la consulta SQL que selecciona las entradas y salidas deseadas
```

Finalmente se testea el modelo de predicción por medio de la
función ARS_Ejecutar, permitiendo obtener un número predicciones indicado un horizonte de predicción partiendo de periodos de la variable tiempo t que se avanza mediante la predicción.

Las predicciones se guardan en forma de tabla, donde las columnas representan las salidas de un único ARS, y las filas indican las salidas de un ARS para un tiempo de predicción t. En el caso de realizar la predicción para de una población de N individuos, la predicción para un periodo t de la población se almacenan en N filas consecutivas. Las N filas siguientes representan la predicción de la población t+1.




## 5 Descripción del sistema de demanda.

En este trabajo solamente vamos a dar una descripción del sistema de demanda en el cual será aplicado el sistema para la predicción. La realización de las predicciones será motivo de trabajos posteriores.

En el marco de la empresa la predicción o previsión de la demanda es cada vez mas compleja y menos fiable, con un aumento de la incertidumbre por factores tales como: mercado altamente cambiante, complejo y muy competitivo que operan bajo condicionantes de marcos políticos inestables, ciclos de vida de los productos masde los datos de entrada determinados, indicando el número

cortos, cambios en las tendencias de consumo cada vez mas rápidas, presión de los distribuidores, elevada rotación de las acciones promocionales, factores y condicionantes en la propia empresa. Todo ello configura un marco complejo y altamente cambiante para lo cual las técnicas de predicción o previsión de clásicas por métodos cuantitativos pudieran no poder conseguir su objetivo de obtener resultados lo mas cercanas a la realidad, por lo que la toma de decisiones no dispone de datos lo suficientemente fiables.

Un sistema de demanda se puede considerar formado por entornos funcionalmente independientes, Fabricante (FBR), Distribuidor (DIS) y Consumidor (CON), entre los cuales existen determinadas relaciones que se caracterizarán por medio de información significativa para la demanda, intercambiadas entre ellos, tales como, stocks, productos, transacciones de ventas, impulsores de demanda etc.

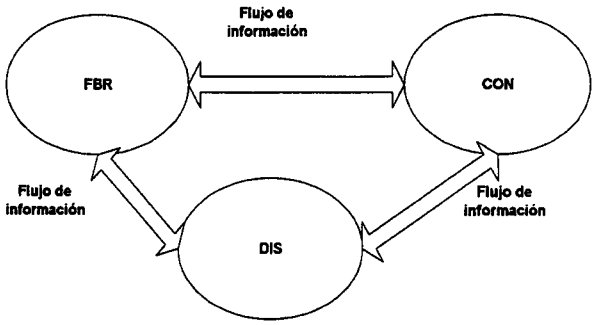



Fig.1.Sistema de demanada

Estas relaciones, Fabricante-Distribuidor (FBR-DIS), Fabricante-Consumidor (FBR-CON) y Distribuidor-Consumidor (DIS-CON) intervienen en diferentes conceptos y eventos [1], los cuales de una forma mas o menos relevante incidirán sobre la variable objeto de la predicción. Por tanto es necesario para el diseño de un modelo de predicción de la demanda el poder conocer cuales son estos eventos y conceptos y como actúan sobre la variable objeto de predicción.

En este trabajo se contemplan las dependencias entre los entornos: Fábrica-Distribuidor (FBR-DIS), y

Distribuidor-Consumidor (DIS-CON) . En el entorno (FBR-DIS) intervienen las variables que caracterizan las **ventas realizadas** por el FBR al DIS, las cuales se representan por series temporales históricas. En éste entorno se distinguen diferentes **tipos de variables significativas,** tales como: el tipo de cliente, producto y los activadores que conforman las ofertas.

Las venta **V(t)** en un periodo t, la podemos representar como una función:

$$V(t) = f(C, A, P(t))$$

donde **C** representa los clientes, **A** los artículos o productos y **P** los activadores que han intervenido en el periodo t. En la función de activación P(t) recogemos de forma cuantificada la opinión que los expertos muestran acerca de como influyen las variables subjetivas, conocidas como impulsoras de la demanda, en las ventas en un periodo V(t).

En el entorno (DIS-CON) se ha de considerar que el consumidor en el punto de venta (DIS) recibe estímulos con diversas ponderaciones por medio de impulsores de demanda los cuales forman el activador que se aplica en el entorno (FBR-DIS). El consumidor responde a estos estímulos cambiando el hábito de compra.

Las variables que representan a los impulsores de la demanda, son en su mayoría de naturaleza subjetivas y por tanto de difícil cuantificación . Así tenemos variables que corresponden al ámbito del consumidor, del producto, de la zona de hábitat, del tipo de centro de venta, de la publicidad etc. Todas estas variables se pueden agrupar según características en común, formando **grupos de impulsores de la demanda**, dado que de una forma u otra hacen que la variable objeto de la predicción (demanda) se vea activada de una forma positiva o negativa.

Si asignamos la variable (**i**) a los impulsores, en el sistema tendremos un conjunto (**I**) de impulsores que actúan de una forma general o particular sobre los consumidores. Por tanto podemos representar a los activadores como una función de los impulsores de la demanda:

$$P(t) = g(I(t))$$

Del análisis de las fuentes información existentes en las bases de datos de la empresa (clientes, productos, históricos de ventas, históricos de promociones) y de las posteriores entrevistas con los expertos, se han de determinar los datos mas relevantes así como su naturaleza (cuantificable o subjetiva). En la tabla 1, se muestran algunos de estos y la estructura de datos que los almacena.

**Tabla. 1 Estructuras de datos disponibles.**

| Datos | Naturaleza | Estructura |
|---|---|---|
| Clientes | Cuantificable | tblCliente |
| Productos | Cuantificable | tblProducto |
| Venta | Cuantificable | tblHistVenta |
| Fecha | Cuantificable | tblDias |
| Promoc | Subjetivo | tblHistPromo |
| Impulsores | Subjetivos | tblImpulsores |
| Activadores | Subjetivo | tblActivador |

A partir de las estructuras de datos citadas se pueden construir las series periódicas de ventas y de impulsores y activadores de la demanda. Las estructuras de datos se han implementado con la base de datos Acces de Microsoft, para poder ser utilizadas en un entorno de Windows NT o Windows 95.

Con el bloque modelizador se puede estudiar la demanda en diversos niveles de agregación, desde el nivel nacional hasta el nivel zona y punto de venta y realiza la predicción, utilizando modelos de representación del comportamiento temporal basados en el paradigma ARS

El proceso de desarrollo de un modelo de predicción tiene tres etapas: Una primera etapa, la de configuración, se parametrizan y determinan uno o varios modelos que nos formara una población de ARS. En una segunda etapa la de selección o evolución, se selecciona el individuo que mejor nos representa el comportamiento del sistema de demanda por medio de un algoritmo genético (AG). En la tercera se realiza el test del modelo que mejor nos represente el comportamiento de la demanda.

Este proceso se puede repetir partiendo del mejor modelo previo hasta que se dispone de un modelo adecuado para ser utilizado en la previsión. Una vez obtenidos los modelos de comportamientos de la demanda, estos son comunicados a la base de datos para ser utilizados por el sistema de explotación.

## 6. Conclusiones.

De lo expuesto se pueden deducir las conclusiones de mas o menos importancia tales como: 1) La utilización de estructuras de tipo ARS permiten el particionar un sistema complejo de una forma jerárquica, facilitando su caracterización. 2) La arquitectura del sistema propuesto permite el poder obtener modelos predictivos de una forma sencilla de sistemas donde las variables sean de naturaleza subjetiva o no. 3) La utilización de AG (Algoritmos

Genéticos) nos permiten el seleccionar de una población de individuos aquellos que mejor modelicen el sistema. 4) La utilización del sistema para la predicción de la demanda puede ser de gran utilidad dado que nos permitirá el poder contemplar variables subjetivas que con las técnicas predictivas convencionales son difíciles o imposible de contemplar.

**Referencias.**


[1] A.Aznar, F.J.Trivez .Métodos de predicción en economía I y II.Ed.Ariel, Barcelona,1993.

[2] Zadeh,L."Soft Computing and Fuzzy Logic": IEEE Software,116(6),Nov, 1994, pp,48-56.

[3] Zadeh,L., and Mo Jashhidi."Soft Computing and Fuzzy Logic:Toward High MIQ Systems", Intern,J.Intelligent Automation Soft Computing, 1(1), 1995,pp,1-30.

[4] Pulido.J.M, Predicción económica y empresarial.Ed.Pirámide. Madrid 1993.

[5] Pulido.J.M, Modelos Econométricos y predicción de series temporales.Ed.Grupo Editorial Iberoamericana, México.D.F 1989.

[6] Holt, C.C, Forecasting Seasonal and Trends by Exponencial Wighted Average. Carnegie Institute of Technology. Pisttsburgh, Pennsilvaia, 1957.

[7] Winters,P.R, Forecasting sales by Exponencial Wighted Moving Average. mangement Science.Abril 1960.

[8] G.E.P Box and G.M.Jenkins, Time Series Analysis: Forecasting and Control. Londron; Holden-Day Inc.19960.

[9] A.Sanz."Sistema para el desarrollo de Sistemas de Respuesta Autónoma (ARS)". XII Reunión Anual de la Agrupación Española de Bioingenieria; pags: 19-22;1993b.

[10] A.Sanz."An Unified Tool for fuzzy and Neural Ntwork Systems". IEEE Micro and Fuzzy Systems.

[11] A.Sanz. "Entorno para el desarrollo de ANN y circuitos Fuzzy". Tesis Doctoral, Zaragoza, Julio 1994.

[12] P.K Dash. Neural networks for time-series forcasting of electrical load. IEE Proc.Gener.Transm Distrib, vol 142, 5 Septiembre (1995), 535-544.

[13] J.J. Shann, H.C.Fu , A fuzzy neural network for rule acquiring on fuzzy control systems. Fuzzy Sets and Systems 71(1995);pp 345-357.

[14] Shin-ichi Horikawa, T.Furuhashi, Y.Uchikawa. On fuzzy modeling using fuzzy neural networks with the backpropagation algorithm. IEEE transactions on neural networks.vol.3 n.5 Septiembre 1992.

[15] Yie.Chien Cheu and Ching-Cheng Teg. A model reference control structure using a fuzzy neural Network. Fuzzy set and Systems 73(1995), 291-312.

[16] J-S Roger Jang. ANFYS.Adaptive-Network-Based Fuzzy Inferece Systems. IEEE Transaction ,Man and Cybernetics. V23.N3, May/June 1993.

[17] Goldberg D,E . "Genetic Algorithms in Search Optimization, and Machine Learning". Reading, MA, 1989. Addison-Wesley.

[18] Davis, L.,ed. "Handbook of Genetic Algorithms".New York: Van Nostrand Reinhld.

# MEDIDAS DE DESIGUALDAD DIFUSAS DE TIPO GASTWIRTH: DEFINICION Y ESTUDIO DE CONDICIONES DE EXISTENCIA

**Hortensia López-García**
Dpto. de Estadística, I.O. y D.M.,
Universidad de Oviedo,
33071 Oviedo
hortensia@etsiig.uniovi.es

**Miguel López-Díaz**
Dpto. de Estadística, I.O. y D.M.,
Universidad de Oviedo,
33071 Oviedo
mld@pinon.ccu.uniovi.es

## Resumen

En este trabajo se introduce una familia de índices de desigualdad relativa difusos para variables aleatorias difusas sobre poblaciones generales. Tras la definición de esos índices, basada en el principio de extensión de Zadeh, se examinan condiciones que garanticen su existencia y se establece una definición equivalente más operativa. El trabajo concluye con un ejemplo ilustrativo del cálculo de uno de estos índices.

**Palabras Clave:** índices de desigualdad de Gastwirth, valor esperado de una variable aleatoria difusa, variable aleatoria difusa, variable aleatoria difusa integrablemente acotada.

## 1 INTRODUCCION

En la literatura sobre la medición de la desigualdad relativa de un aspecto poblacional, se ha supuesto tradicionalmente que ese atributo provenía de un proceso cuantitativo (recuento, medida, etc.) que podía formalizarse adecuadamente mediante una variable aleatoria positiva.

Esa suposición ha limitado en cierto modo los campos de aplicación de los estudios correspondientes, siendo sin duda el económico el que más se ha beneficiado de dichos estudios. En este sentido, la mayoría de trabajos sobre la desigualdad hacen referencia expresa a variables como las rentas, o los beneficios, sobre poblaciones de familias o empresas.

El objetivo de este trabajo es ampliar el conjunto de los posibles campos de aplicación, incorporando situaciones más generales a las que pueda adaptarse el modelo de las variables aleatorias difusas. De este modo, muchas situaciones prácticas en las que interesa cuantificar la desigualdad incluyen variables de opinión, lingüísticas, etc., fácilmente asimilables con variables aleatorias difusas.

La cuantificación mencionada se llevará a cabo mediante la extensión de algunos índices clásicos (en concreto, índices que están definidos como el valor esperado de cierta función que depende de la variable aleatoria y de su valor esperado). Esta extensión se basará en los conceptos de variable aleatoria difusa (según la definición de Puri and Ralescu [9]), valor esperado de una variable aleatoria difusa integrablemente acotada (Puri and Ralescu [9]), y principio de extensión de Zadeh [10] aplicado a las operaciones algebraicas elementales (ver, por ejemplo, Kaufmann and Gupta [6], Mareš [8]).

La clase de índices de desigualdad seleccionados para la extensión se apoya en la familia establecida por Gastwirth [3] (ver también, Gastwirth *et al.* [4]), que incluye a su vez a la clase de índices aditivamente descomponibles (Bourguignon [1]), algunos de ellos a excepción de un factor adicional (el valor esperado del aspecto).

## 2 LOS INDICES DE DESIGUALDAD DIFUSOS DE TIPO GASTWIRTH: DEFINICION Y CONDICIONES DE EXISTENCIA

Para medir la desigualdad difusa correspondiente a un aspecto $\mathcal{X}$ en una población se proponen los siguientes índices:

**Definición 2.1** *Sea* $(\Omega, \mathcal{A}, P)$ *un espacio de probabilidad asociado a una población y* $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$ *una variable aleatoria difusa integrablemente acotada y positiva que representa cierto aspecto vinculado al anterior espacio probabilístico. El* $\phi$*-índice de desigualdad difuso asociado con* $\mathcal{X}$ *en la*

*población viene dado por:*

$$\widetilde{I}_{\phi}(\mathcal{X}) = \widetilde{E}\left[\phi(\mathcal{X}) \oslash \phi\left(\widetilde{E}(\mathcal{X})\right)\right] \ominus 1,$$

*siempre que dicho valor exista, con $\oslash$ y $\ominus$ operadores cociente y sustracción difusos, donde $\phi : (0,+\infty) \to (0,+\infty)$ es una función monótona, cóncava hacia arriba (en sentido estricto), y donde $\phi(\mathcal{X})$ y $\phi(\widetilde{E}(\mathcal{X}))$ son la aplicación difusa y el valor de dicha aplicación en la esperanza difusa, respectivamente, obtenidos a partir de $\mathcal{X}$ mediante el principio de extensión de Zadeh.*

En general, la existencia de $\widetilde{I}_{\phi}$ no puede asegurarse para cualquier variable o población, y las condiciones para su existencia dependen de la función $\phi$.

Los siguientes resultados servirán de apoyo para el establecimiento de una condición suficiente de existencia del $\phi$-índice de desigualdad difuso.

**Lema 2.1** *Sea $\phi : (0,+\infty) \to (0,+\infty)$ una función continua e inyectiva. Si $\widetilde{V} \in \mathcal{F}_c((0,+\infty))$ y $\phi(\widetilde{V})$ es el conjunto difuso obtenido a partir de $\widetilde{V}$ mediante el principio de extensión de Zadeh, entonces $\phi(\widetilde{V}) \in \mathcal{F}_c((0,+\infty))$.*

**Lema 2.2** *Dados un espacio de probabilidad $(\Omega, \mathcal{A}, P)$ y una variable aleatoria difusa positiva $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, si se considera una aplicación $\phi : (0,+\infty) \to (0,+\infty)$ monótona estricta y de clase $C^1$, entonces la aplicación $\phi(\mathcal{X}) : \Omega \to \mathcal{F}_c((0,+\infty))$ definida por $\phi(\mathcal{X})(\omega) = \phi(\mathcal{X}(\omega))$ para todo $\omega \in \Omega$, es una variable aleatoria difusa.*

**Lema 2.3** *Dados un espacio de probabilidad $(\Omega, \mathcal{A}, P)$ y una variable aleatoria difusa positiva $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, si se considera una aplicación $\phi : (0,+\infty) \to (0,+\infty)$ monótona estricta y de clase $C^1$, se tiene que la variable aleatoria difusa $\phi(\mathcal{X}) : \Omega \to \mathcal{F}_c((0,+\infty))$ es integrablemente acotada si, y sólo si, $\phi$ verifica que:*

*i) $\phi(\inf \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ decreciente.*

*ii) $\phi(\sup \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ creciente.*

**Lema 2.4** *Dados un espacio de probabilidad $(\Omega, \mathcal{A}, P)$ y una variable aleatoria difusa integrablemente acotada y positiva $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, si se considera una aplicación $\phi : (0,+\infty) \to (0,+\infty)$ monótona estricta y de clase $C^1$, entonces la aplicación $\mathcal{Z} : \Omega \to \mathcal{F}_c((0,+\infty))$ con $\mathcal{Z} = \phi(\mathcal{X}) \oslash \phi\left(\widetilde{E}(\mathcal{X})\right)$ es una variable aleatoria difusa.*

**Lema 2.5** *Dados un espacio de probabilidad $(\Omega, \mathcal{A}, P)$ y una variable aleatoria difusa integrablemente acotada y positiva $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, si se considera una aplicación $\phi : (0,+\infty) \to (0,+\infty)$ monótona estricta y de clase $C^1$, se tiene que la variable aleatoria difusa $\mathcal{Z} : \Omega \to \mathcal{F}_c((0,+\infty))$ con $\mathcal{Z} = \phi(\mathcal{X}) \oslash \phi\left(\widetilde{E}(\mathcal{X})\right)$ es integrablemente acotada si, y sólo si, $\phi$ verifica que:*

*i) $\phi(\inf \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ decreciente.*

*ii) $\phi(\sup \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ creciente.*

**Lema 2.6** *Dados un espacio de probabilidad $(\Omega, \mathcal{A}, P)$ y una variable aleatoria difusa integrablemente acotada y positiva $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, si se considera una aplicación $\phi : (0,+\infty) \to (0,+\infty)$ monótona estricta y de clase $C^1$, se tiene que la aplicación $\mathcal{V} : \Omega \to \mathcal{F}_c(\mathbb{R})$ con $\mathcal{V} = \left[\phi(\mathcal{X}) \oslash \phi\left(\widetilde{E}(\mathcal{X})\right)\right] \ominus 1$ es una variable aleatoria difusa. Además, es integrablemente acotada si, y sólo si, $\phi$ verifica que:*

*i) $\phi(\inf \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ decreciente.*

*ii) $\phi(\sup \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ creciente.*

**Teorema 2.7** *Dados un espacio de probabilidad $(\Omega, \mathcal{A}, P)$ y una variable aleatoria difusa integrablemente acotada y positiva $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, si se considera una aplicación $\phi : (0,+\infty) \to (0,+\infty)$ monótona estricta y de clase $C^1$, se tiene que el $\phi$-índice de desigualdad difuso dado por:*

$$\widetilde{I}_{\phi}(\mathcal{X}) = \widetilde{E}\left[\phi(\mathcal{X}) \oslash \phi\left(\widetilde{E}(\mathcal{X})\right)\right] \ominus 1,$$

*está bien definido y pertenece a $\mathcal{F}_c(\mathbb{R})$ si, y sólo si, $\phi$ verifica que:*

*i) $\phi(\inf \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ decreciente.*

*ii) $\phi(\sup \mathcal{X}_0) \in L^1(P)$, en el caso de $\phi$ creciente.*

**Corolario 2.8** *Dados un espacio de probabilidad $(\Omega, \mathcal{A}, P)$ y una variable aleatoria difusa positiva y simple $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, si se considera una aplicación $\phi : (0,+\infty) \to (0,+\infty)$ monótona estricta y de clase $C^1$, entonces el $\phi$-índice de desigualdad difuso dado por:*

$$\widetilde{I}_{\phi}(\mathcal{X}) = \widetilde{E}\left[\phi(\mathcal{X}) \oslash \phi\left(\widetilde{E}(\mathcal{X})\right)\right] \ominus 1,$$

*está bien definido y pertenece a $\mathcal{F}_c(\mathbb{R})$.*

## 3 DEFINICION ALTERNATIVA DE LOS INDICES DE GASTWIRTH

Los siguientes resultados ofrecen una versión equivalente de la Definición 2.1 (bajo condiciones de concavidad estricta) en términos de sus $\alpha$-cortes. Dichas expresiones serán muy útiles en la formalización de las propiedades de $\widetilde{I}_{\phi}$.

**Teorema 3.1** *Dados un espacio de probabilidad* $(\Omega, \mathcal{A}, P)$ *y una variable aleatoria difusa integrablemente acotada y positiva* $\mathcal{X} : \Omega \to \mathcal{F}_c((0,+\infty))$, *si se considera una aplicación* $\phi : (0,+\infty) \to (0,+\infty)$ *monótona estricta, de clase* $C^1$ *y tal que* $\widetilde{I}_\phi(\mathcal{X}) \in \mathcal{F}_c(\mathbb{R})$, *entonces el* $\phi$*-índice de desigualdad difuso asociado a* $\mathcal{X}$ *en la población es el valor en* $\mathcal{F}_c((-1,+\infty))$ *cuyo* $\alpha$*-corte, para todo* $\alpha \in [0,1]$, *viene dado por:*

$$\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha = \left[\inf\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha, \sup\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha\right],$$

*donde:*

$$\inf\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha = E\left(\min\left\{\frac{\phi(\inf \mathcal{X}_\alpha)}{\phi\left(E(\sup \mathcal{X}_\alpha)\right)}, \frac{\phi(\sup \mathcal{X}_\alpha)}{\phi\left(E(\inf \mathcal{X}_\alpha)\right)}\right\}\right) - 1,$$

$$\sup\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha = E\left(\max\left\{\frac{\phi(\inf \mathcal{X}_\alpha)}{\phi\left(E(\sup \mathcal{X}_\alpha)\right)}, \frac{\phi(\sup \mathcal{X}_\alpha)}{\phi\left(E(\inf \mathcal{X}_\alpha)\right)}\right\}\right) - 1,$$

*es decir, si* $\phi$ *es decreciente,* $\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha$ *es igual a:*

$$\left[E\left(\frac{\phi(\sup \mathcal{X}_\alpha)}{\phi\left(E(\inf \mathcal{X}_\alpha)\right)}\right) - 1, E\left(\frac{\phi(\inf \mathcal{X}_\alpha)}{\phi\left(E(\sup \mathcal{X}_\alpha)\right)}\right) - 1\right],$$

*mientras que si* $\phi$ *es creciente, es igual a:*

$$\left[E\left(\frac{\phi(\inf \mathcal{X}_\alpha)}{\phi\left(E(\sup \mathcal{X}_\alpha)\right)}\right) - 1, E\left(\frac{\phi(\sup \mathcal{X}_\alpha)}{\phi\left(E(\inf \mathcal{X}_\alpha)\right)}\right) - 1\right].$$

A continuación se presenta la particularización de este teorema para poblaciones finitas:

**Corolario 3.2** *Sea* $\mathcal{X}$ *una variable aleatoria difusa positiva definida sobre una población de* $N$ *individuos, sobre los que toma los valores difusos* $\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_N \in \mathcal{F}_c((0,+\infty))$, *distintos o no. Se considera una aplicación* $\phi : (0,+\infty) \to (0,+\infty)$ *monótona estricta y de clase* $C^1$. *Entonces el* $\phi$*-índice de desigualdad difuso asociado a* $\mathcal{X}$ *en la población es el valor en* $\mathcal{F}_c((-1,+\infty))$ *cuyos* $\alpha$*-cortes, para todo* $\alpha \in [0,1]$, *vienen dados por:*

$$\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha = \left[\inf\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha, \sup\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha\right],$$

*donde:*

$$\inf\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha = \min\left\{\frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\inf(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\sup(\tilde{x}_i)_\alpha\right)} - 1, \frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\sup(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\inf(\tilde{x}_i)_\alpha\right)} - 1\right\},$$

$$\sup\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha = \max\left\{\frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\inf(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\sup(\tilde{x}_i)_\alpha\right)} - 1, \frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\sup(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\inf(\tilde{x}_i)_\alpha\right)} - 1\right\},$$

*es decir, si* $\phi$ *es decreciente* $\left(\widetilde{I}_\phi(\mathcal{X})\right)_\alpha$ *es igual a:*

$$\left[\frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\sup(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\inf(\tilde{x}_i)_\alpha\right)} - 1, \frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\inf(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\sup(\tilde{x}_i)_\alpha\right)} - 1\right],$$

*mientras que si* $\phi$ *es creciente, es igual a:*

$$\left[\frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\inf(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\sup(\tilde{x}_i)_\alpha\right)} - 1, \frac{\frac{1}{N}\sum_{i=1}^{N}\phi(\sup(\tilde{x}_i)_\alpha)}{\phi\left(\frac{1}{N}\sum_{i=1}^{N}\inf(\tilde{x}_i)_\alpha\right)} - 1\right].$$

## 4 EJEMPLO ILUSTRATIVO

Para ilustrar el cálculo del índice hiperbólico (es decir, para $\phi(x) = 1/x$) se va a presentar un ejemplo real (López-García [7]).

Se consideran dos poblaciones que corresponden a un curso universitario de Primer Ciclo, el primero formado por 105 estudiantes de Biología y el segundo formado por 76 estudiantes de Matemáticas. En ambos cursos se imparte como materia troncal una Introducción a la Estadística.

La Universidad en la que se encuentran los dos cursos anteriores, está interesada en realizar un estudio en el que se comparen las opiniones de esos cursos respecto a la importancia y utilidad de esa materia. Con este fin se ha llevado a cabo una encuesta entre los estudiantes de los dos cursos, en la que se les pide que indiquen el "grado de relevancia de la Estadística para su formación científico-profesional". Las respuestas posibles a dicha cuestión son: $\tilde{x}_1$ = "muy bajo", $\tilde{x}_2$ = "bajo", $\tilde{x}_3$ = "mediano", $\tilde{x}_4$ = "alto", $\tilde{x}_5$ = "muy alto", $\tilde{x}_6$ = "no sabe/no contesta".

Los resultados de la encuesta son los siguientes: los estudiantes de Biología han respondido "muy bajo" 11 estudiantes, "bajo" 24, "mediano" 31, "alto" 25, "muy alto" 8, y "no sabe/no contesta" 6, mientras que los estudiantes de Matemáticas han respondido "bajo" 5, "mediano" 13, "alto" 32, "muy alto" 25 y "no sabe/no contesta" 1.

Supongamos que las respuestas anteriores se identifican con números difusos positivos cuyos soportes están contenidos en el intervalo $[1, 10]$ y que están expresados en términos de las representaciones de $S$-curvas en la Figura 1 (ver, por ejemplo, Zadeh [11], Cox [2], Jang and Gulley [5]).

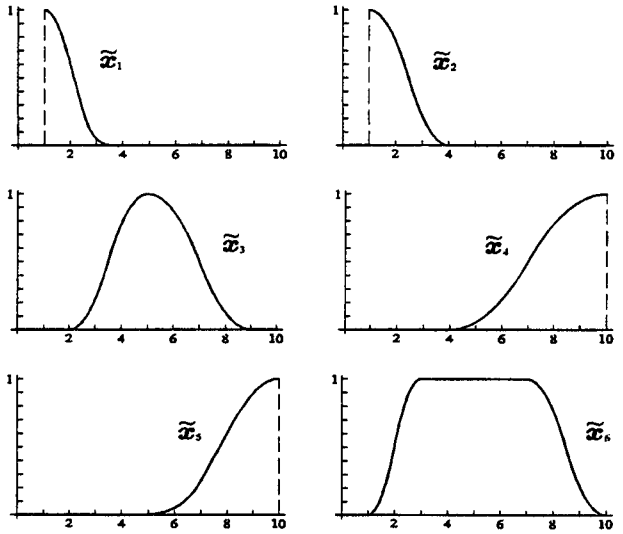



**Figura 1** Respuestas sobre el grado de relevancia de la Estadística en la formación científico-profesional: $\tilde{x}_1$="muy bajo", $\tilde{x}_2$="bajo", $\tilde{x}_3$="mediano", $\tilde{x}_4$="alto", $\tilde{x}_5$="muy alto", $\tilde{x}_6$="no sabe/no contesta"

Si $\mathcal{X}$ es la variable aleatoria difusa positiva definida sobre las dos poblaciones consideradas ($\mathcal{X}_{(1)}$ para los estudiantes de Biología y $\mathcal{X}_{(2)}$ para los estudiantes de Matemáticas) y que toma como valores distintos sobre ellas $\tilde{x}_1$, $\tilde{x}_2$, ..., $\tilde{x}_6$, entonces una medida resumen interesante en el estudio realizado por la Universidad sería la desigualdad difusa (por ejemplo, hiperbólica) asociada a $\mathcal{X}$ en ambas.

La Figura 2 presenta los valores del índice hiperbólico para la primera y la segunda población.

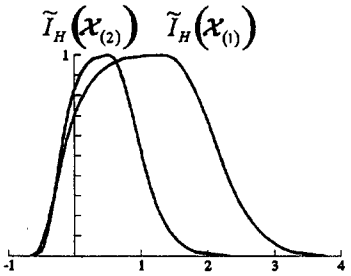



**Figura 2** Indices de desigualdad hiperbólicos para la población de estudiantes de Biología, $\tilde{I}_H\left(\mathcal{X}_{(1)}\right)$, y para los estudiantes de Matemáticas, $\tilde{I}_H\left(\mathcal{X}_{(2)}\right)$

A partir de su representación gráfica, puede concluirse que la desigualdad en la opinión acerca del "grado de relevancia de la Estadística para su formación científico-profesional" es superior (según la mayoría de los criterios de ordenación de números difusos) en la población de estudiantes de Biología que en la población de estudiantes de Matemáticas cuando se considera el índice hiperbólico.


### Agradecimientos

La investigación en este trabajo se ha financiado en parte mediante el Proyecto DGES No. PB95-1049 del Ministerio de Educación y Cultura, al que deseamos poner de manifiesto nuestro agradecimiento.


## Referencias


[1] Bourguignon, F. (1979). Decomposable income inequality measures. *Econometrica*, **47**, 901-920.

[2] Cox, E. (1994). *The Fuzzy Systems Handbook.* Academic Press, Cambridge.

[3] Gastwirth, J.L. (1975). The estimation of a family of measures of income inequality. *Journal of Econometrics*, **3**, 61-70.

[4] Gastwirth, J.L., Nayak, T.K. and Krieger, A.M. (1986). Large sample theory for the bounds on the Gini and related indices of inequality estimated from grouped data. *Journal of Business and Economic Statistics*, **4**, 269-273.

[5] Jang, J.-S.R. and Gulley, N. (1995). *Fuzzy Logic Toolbox for use with MATLAB.* The Math Works Inc., Natick-Massachussets.

[6] Kaufmann, A. and Gupta, M.M. (1991). *Introduction to Fuzzy Arithmetic.* Van Nostrand Reinhold Co., New York.

[7] López-García, H. (1997). *Cuantificación de la desigualdad asociada a conjuntos aleatorios y variables aleatorias difusas.* Tesis doctoral. Universidad de Oviedo.

[8] Mareš, M. (1994). *Computation over Fuzzy Quantities.* CRC Press, Boca Raton.

[9] Puri, M.L. and Ralescu, D.A. (1986). Fuzzy random variables. *J. Math. Anal. Appl.*, **114**, 409-422.

[10] Zadeh, L.A. (1975). The concept of a linguistic variable and its application to approximate reasoning. *Inform. Sci.*, Part 1, **8** 199-249; Part 2, **8** 301-353; Part 3, **9**, 43-80.

[11] Zadeh, L.A. (1976). A fuzzy-algorithmic approach to the definition of complex or imprecise concepts. *Int. Jour. Man-Machine Studies*, **8**, 249-291.

# MEDIDAS DE INFORMACIÓN DE UN CONJUNTO DIFUSO

**Carlo Bertoluzza and Antonella Bodini**
Dpt. Informatica e Sistemistica, Univeridad de Pavia
via Ferrata 1, 27100 Pavia, Italia
e-mail: carlo@ipv36.unipv.it

## Resumen

El objetivo de esta investigación es la construcción de una medida de información en una oportuna algebra de subconjuntos difusos a partir de una medida de información nítida. En un precedente trabajo hemos resuelto el problema en el caso de subconjuntos difusos cuyos valores de pertenencia son a lo sumo numerables. Aquí presentamos la solución en el caso general.

**Palabras Clave:** Medidas de Información, Conjunto difuso medible.

## 1 INTRODUCCIÓN

Una *medida de información* ([1]) sobre un espacio medible $(\Omega, \mathcal{A})$ es una función

$$J : \Omega \longrightarrow \mathbb{R}^+$$

con las siguientes propiedades

$$J(\Omega) = 0 \quad , \quad J(\emptyset) = +\infty \quad (a)$$

$$A \subset B \implies J(A) \geq J(B) \quad (b)$$

$$A \cap B = \emptyset \implies J(A \cup B) = J(A) \perp J(B) \quad (c)$$

donde $\perp$ es una norma triangular continua cuya expresión mas general ([2]) es

$$x \perp y = \begin{cases} f_i^{(-1)}[f_i(x) + f_i(y)] \\ \quad \text{si } x, y \in ]a_i, b_i[ \\ \\ \min(x, y) \\ \text{en caso contrario} \end{cases} \quad (1)$$

donde $\{]a_i, b_i[\}$ es una sucesión numerable de intervalos abiertos disjuntos, $f_i : ]a_i, b_i[ \rightarrow \mathbb{R}^+$ es estrictamente decreciente con $f_i(b_i) = 0$, y $f^{(-1)}(t) = f^{-1}[\min\{t,\ f(b)\}]$.

Los elementos del subconjunto $\Delta = \mathbb{R}^+ - \cup ]a_i, b_i[$ son *idempotentes* y cumplen con las propiedades

$$x \in \Delta \implies x \perp x = x \quad (2)$$

$$x \in \Delta \implies x \perp y = \min(x, y) \quad (3)$$

## 2 EL CASO DISCRETO

En un espacio $\Omega$ cosideramos los subconjuntos difusos del tipo

$$\widetilde{A} = \bigcup_{i=0}^{n} \alpha_i A_{(i)} \quad (4)$$

$$A_{(i)} = \{\omega \in \Omega \mid \widetilde{A}(\omega) = \alpha_i\} \quad (5)$$

donde $0 = \alpha_0 \leq \alpha_1 \leq \ldots \leq \alpha_n = 1$. Suponiendo que $\mathcal{I} = (\Omega,\ \mathcal{A},\ J,\ \perp)$ sea un espacio de información compositiva y que $A_{(i)} \in \mathcal{A}$, queremos asociar a $\widetilde{A}$ una medida de información $\overline{J}(\widetilde{A})$ (*información difusa*) que sea compatible con $\mathcal{I}$ en el sentido que

$$\overline{J}(\widetilde{A}) = G_n[\alpha_1, \ldots, \alpha_n, J(A_{(1)}), \ldots, J(A_{(n)})] \quad (6)$$

También pretendemos que $\overline{J}$ sea compositiva con ley $\overline{\perp}$ e indicaremos con $\overline{\Delta}$, $\overline{f}_j$, $]\overline{a}_j, \overline{b}_j[$, los elementos que determinan $\overline{\perp}$.
Importancia especial tiene la función $G(\alpha, x)$ (*función de transferencia*) que determina la información difusa de los conjuntos con un solo nivel $\widetilde{A} = \alpha A$ por medio de

$$\overline{J}(\alpha A) = G[\alpha, J(A)] \quad (7)$$

En el trabajo [2] se ha probado que

$$\alpha' < \alpha'' \Rightarrow G(\alpha', x) \geq G(\alpha'', x)$$

$$x' < x'' \Rightarrow G(\alpha, x') \leq G(\alpha, x'')$$

$$G(\alpha, x \perp y) = G(\alpha, x) \overline{\perp} G(\alpha, y)$$

$$G(\alpha, \Delta) \subset \overline{\Delta}$$

$$\overline{M} \notin \overline{\Delta} \Rightarrow \overline{M} \in ]\overline{a}, \overline{b}[ \subseteq G(\alpha, ]a, b[)$$

$$\overline{M} \in \overline{\Delta} \Longrightarrow \overline{J}(\widetilde{A}) = \overline{M} \quad (8)$$

$$G(\alpha, x) = \overline{f}^{(-1)}[k(\alpha) f(x)] \quad (9)$$

donde $\overline{M} = \inf\{\overline{J}(\alpha_i A_{(i)})\}$, $f, \overline{f}$ son las funciones que determinan $\perp$ en $]a, b[, ]\overline{a}, \overline{b}[$ respectivamente, y $k$ es una función cualquiera no decreciente. Por fin se ha probado que, cuando $\overline{M} \notin \overline{\Delta}$

$$\overline{J}(\widetilde{A}) = \overline{f}^{(-1)}[\sum_{j \in H} \overline{f} \circ G(\alpha_j, J(A_{(j)})] \quad (10)$$

donde $H = \{r \mid G(\alpha_r, J(A_{(r)}) \in ]\overline{a}, \overline{b}[\}$.

# 3 EL CASO CONTINUO

**Definición 3.1** Un subconjunto difuso $\widetilde{A}$ es *medible* si y solo si, $\forall\, \alpha, \beta \in [0, 1]$

$$A_\alpha^\beta = \{\omega \mid \alpha \leq \widetilde{A}(\omega) < \beta\} \in \mathcal{A}.$$

La familia $\widetilde{\mathcal{A}}$ de los difusos medibles es un algebra, así que $(\Omega, \widetilde{\mathcal{A}})$ es un espacio medible.

**Definición 3.2** Sean $\widetilde{A}$ un conjunto difuso, $0 \leq \alpha_1 < \ldots \alpha_n = 1$ una sucesión de valores en $[0, 1]$ y $A_i = \{\omega \mid \alpha_i \leq \widetilde{A}(\omega) < \alpha_{i+1}\}$. Llamaremos *aproximación estándar* de $\widetilde{A}$ al conjunto difuso

$$\widetilde{P}(\widetilde{A}, \alpha_1 \ldots \alpha_n) = \cup \alpha_i A_i$$

En base a los resultados del párrafo 2 la información difusa de las aproximaciones estándar queda completamente individualizada. A partir de aquí determinaremos $\overline{J}(\widetilde{A})$.

**Definición 3.3** La *medida de información difusa* de un conjunto $\widetilde{A}$ se define como

$$\overline{J}(\widetilde{A}) = \inf\{\overline{J}(\widetilde{P}) \mid \widetilde{P} \in \mathcal{P}(\widetilde{A})\} \quad (11)$$

donde $\mathcal{P}(\widetilde{A})$ es la familia de las aproximaciones estándar de $\widetilde{A}$.

**Definición 3.4** El *elemento discriminante* del subconjunto $\widetilde{A}$ es el número

$$\Gamma(\widetilde{A}) = \inf\{\overline{J}(\alpha A) \mid \alpha A \subset \widetilde{A}\} \quad (12)$$

**Teorema 3.1** La medida de información difusa de la definición 3.3 es compositiva en $\widetilde{\mathcal{A}}$.

**Teorema 3.2** Si $\Gamma(\widetilde{A}) \in \overline{\Delta}$, entonces $\overline{J}(\widetilde{A}) = \Gamma(\widetilde{A})$.

**Teorema 3.3** Si $\Gamma(\widetilde{A}) \notin \overline{\Delta}$ (es decir $\Gamma(\widetilde{A})$ pertenece a un intervalo $]c, d[$ de no-idempotentes) entonces

$$c \leq \overline{J}(\widetilde{A}) \leq \Gamma(\widetilde{A}) < d \quad (13)$$

Considerando ahora los subconjuntos nitidos $A_\alpha^{\alpha+\epsilon}$ vamos a subdividir el conjunto $[0, 1]$ de los valores de pertenencia en dos clases: en la primera pondremos los valores que no van a influir en el cómputo de $\overline{J}(\widetilde{A})$, en la segunda aquellos que son relevantes al fin del mismo cómputo. Esta subdivisión queda asociada a la familia de abiertos disjuntos $\{]a_i, b_i[ | i \in I\}$ que determina la ley de composicion nitida $\perp$. Mas precisamente: para cada $\alpha \in [0, 1]$ definimos $\xi(\alpha) = \inf\{x \in \mathbb{R}^+ \mid G(\alpha, x) \geq d\}$.

**Teorema 3.4** Existe $\alpha_0 \geq 0$ tal que $G(\alpha, 0) \geq d$ para todo $\alpha \leq \alpha_0$.

**Teorema 3.5** Para todo $\alpha \leq \alpha_0$ existe $]a, b[ \subset \Delta^c$ tal que $\xi(\alpha) = b$. Para todo $i \in I$ ponemos $T_0 = [0, \alpha_0]$, $T_i = \{\alpha \mid \xi(\alpha) = b_i\}$.

La subdivisión previamente mencionada se obtiene a partir da los subconjuntos

$$D_i = \{\alpha \in T_i \mid \lim_{\epsilon \to 0} J(A_\alpha^{\alpha+\epsilon}) \geq b_i\}$$

$$R_i = T_i - D_i$$

$$R = \cup_{i \in I} R_i \quad , \quad D = \cup_{i \in I} D_i$$

$$\Omega_{R,i} = \{\omega \in \Omega \mid \widetilde{A}(\omega) \in R_i\}$$

$$\Omega_R = \bigcup_{i \in I} \Omega_{R,i}$$

$$\Omega_{D,i} = \{\omega \in \Omega \mid \widetilde{A}(\omega) \in D_i\}$$

$$\Omega_D = \bigcup_{i \in I} \Omega_{D,i}$$

donde los subindices $R$ y $D$ significan respetivamente "relevante" y "despreciable". Los subconjuntos difusos $\widetilde{A}_*$ que vamos a definir a continuación son aquellos que nos permitiran obtener explicitamente la medida de información difusa $\overline{J}(\widetilde{A})$. Esta misma información se construye por medio de los teoremas 3.6, 3.7, 3.8.

$$\widetilde{A}_{R,i}(\omega) = \begin{cases} \widetilde{A}(\omega) & \text{si } \omega \in \Omega_{R,i} \\ 0 & \text{de otro modo} \end{cases} \tag{14}$$

$$\widetilde{A}_R(\omega) = \begin{cases} \widetilde{A}(\omega) & \text{si } \omega \in \Omega_R \\ 0 & \text{de otro modo} \end{cases} \tag{15}$$

$$\widetilde{A}_{D,i}(\omega) = \begin{cases} \widetilde{A}(\omega) & \text{si } \omega \in \Omega_{D,i} \\ 0 & \text{de otro modo} \end{cases} \tag{16}$$

$$\widetilde{A}_D(\omega) = \begin{cases} \widetilde{A}(\omega) & \text{si } \omega \in \Omega_D \\ 0 & \text{de otro modo} \end{cases} \tag{17}$$

Es inmediato reconocer que

**Teorema 3.6** Valen las siguientes propiedades:

$$\widetilde{A}_D \cap \widetilde{A}_R = \emptyset \quad \widetilde{A}_D \cup \widetilde{A}_R = \widetilde{A}$$

$\widetilde{A}_{R,i}$, $\widetilde{A}_R$, $\widetilde{A}_{D,i}$, $\widetilde{A}_D$
son medibles

$$\widetilde{P} \in \mathcal{P}(\widetilde{A}_R) \implies c \leq \overline{J}(\widetilde{P}) \leq d$$

$$\{\widetilde{A}_{R,i} \mid i \in I\} \text{ es una partición de } \widetilde{A}_R \tag{18}$$

$$\overline{J}(\widetilde{A}) = \overline{J}(\widetilde{A}_R) \tag{19}$$

Sean $\overline{f}$, $f_i$ los generadores aditivos respectivamente de $\overline{\perp}$ en $]c,d[$ y de $\perp$ en $]a_i,b_i[$. Valen los dos siguientes resultados.

**Teorema 3.7.a** Si $f_i \circ J$ es aditiva, entonces

$$\overline{J}(\widetilde{A}_{R,i}) = \overline{f}^{(-1)}[\int_{\Omega_{R,i}} k(\widetilde{A}_{R,i})\, d(f_i \circ J)]$$

**Teorema 3.7.b** Si $f_i \circ J$ es pseudo-aditiva, entonces, por definición (ver [3],[4]), existe una partición $\{\Omega_1^*, \ldots, \Omega_m^*\}$ del universo $\Omega$ con $f_i[J(\Omega_t^*)] < f_i(0)$, $\sum_t f_i[J(\Omega_t^*)] > f_i(0)$, o sea la medida de la unión de subconjuntos disjuntos es menor de la suma de las medidas. En este caso $\overline{J}(\widetilde{A}_{R,i})$ se determina como medida de la unión de los conjuntos disjuntos $\widetilde{A}_{R,i} \cap \Omega_t^*$. Mas precisamente

$$\overline{J}(\widetilde{A}_{R,i} = \overline{f}^{(-1)}[\textstyle\sum_t \int_{\Omega_{R,i} \cap \Omega_t^*} k(\widetilde{A}_{R,i})\, d(f_i \circ J)]$$

**Teorema 3.8** A partir de (18)(19) y del teorema (3.7) se obtiene inmediatamente la expresión de $\overline{J}(\widetilde{A})$. En efecto, teniendo en cuenta que los $A_{R,i}$ constituyen una partición de $A_R$ resulta

$$\overline{J}(\widetilde{A}) = \overline{J}(\widetilde{A}_R) = \overline{f}^{(-1)}[\textstyle\sum_{i \in I} \overline{f} \circ \overline{J}(\widetilde{A}_{R,i})] \tag{20}$$

**Ejemplo** Aquí $\Omega = [0,1]$, $\mathcal{A} = \mathcal{B}[0,1]$ y $\widetilde{\mathcal{A}}$ es el algebra de los dufusos cuyos $\alpha-$cortes y $\alpha-$niveles pertenecen a $\mathcal{A}$. $\widetilde{A}$ será el conjunto triangular definido por

$$\widetilde{A}(\omega) = \begin{cases} 2\omega & \text{si } \omega \in [0, 0.5] \\ 2(1-\omega) & \text{si } \omega \in [0.5, 1] \end{cases}$$

La medida de información nitida y la difusas serán archimedianas con generador aditivo $f(x) = (1+x)^{-1}$. Un posible ejemplo de este espacio es la medida de Picard $J(A) = m(A)^{-1} - 1$ donde $m(A)$ es la medida de Lebesgue; en este caso se reconoce facilmente que $J(A_\alpha) = (1-\alpha)^{-1} - 1$. Si escogemos $G(\alpha, J) = \alpha^{-2}(1+J) - 1$ como función de transferencia (lo que corresponde al escoger $k(\alpha = \alpha^2$ en la (9)), entonces se reconoce facilmente que $\Gamma(\widetilde{A}) = \inf_\alpha[\alpha^{-2}(1-\alpha)^{-1} - 1] = 7$. Considerando que $\overline{\Delta} = \{0, +\infty\}$ y por lo tanto $\Omega_R = \Omega$ y que $f \circ J$ es la medida de Lebesgue en $[0,1]$, aplicando el resultado del teorema 3.7.a, se obtiene

$$\overline{J}(\widetilde{A}) = [\int_0^1 \widetilde{A}^2(x)dx]^{-1} - 1 = 2$$

# 4 ANEXO: Esbozo de las demonstraciones

Los resultados mas importantes son los teoremas 3.2 que determina $\overline{J}(\widetilde{A})$ en el caso que $\Gamma(\widetilde{A})$ es idempotente y los teoremas 3.7.a,b que sirven cuando $\Gamma(\widetilde{A})$ no es idempotente (el resultado final, teorema 3.8 se obtiene aplicando directamente la ley de composición $\overline{\perp}$).

La prueba del teorema 3.2 es muy sencilla. En efecto la monotonia de $\overline{J}$ asegura siempre que $\Gamma(\widetilde{A}) \geq \overline{J}(\widetilde{A})$. Por otro lado comoquiera se escoga una aproximación estándar $\widetilde{P}(\widetilde{A}, \alpha_1 \ldots \alpha_n)$ tenemos $\overline{J}(\widetilde{P}) = \overline{J}(\alpha_1 A_1)\overline{\perp}\ldots\overline{\perp}\overline{J}(\alpha_n A_n)$ y $\Gamma(\widetilde{A}) \leq \overline{J}(\alpha_i A_i)$. Por consiguiente

$$\begin{aligned}\overline{J}(\widetilde{P}) &= \overline{J}(\alpha_1 A_1)\overline{\perp}\ldots\overline{\perp}\overline{J}(\alpha_n A_n)\\ &\geq \Gamma(\widetilde{A})\overline{\perp}\ldots\overline{\perp}\Gamma(\widetilde{A}) = \Gamma(\widetilde{A})\end{aligned}$$

Pasando al extremo inferior se obtiene el teorema.

La demonstración del teorema 3.7.a es mas compleja. Por primero se reconoce que contribuyen a determinar $\overline{J}(\widetilde{P})$ solo los elementos $\alpha A$ para los cuales $G[\alpha, J(A)] \in ]c, d[$. Las informaciones nítidas de estos elementos pertenecen a un cierto numero de intervalos abiertos disjuntos $]a_i, b_i[$ de no idempotentes respecto a la ley de composición nítida $\perp$; son aquellos que determinan los subconjuntos $\Omega_{R,i}$ de la parte relevante del conjunto $\widetilde{A}$ o sea las unicas partes de $\Omega$ que contribuyen a la medida de $\widetilde{A}$.

Para cada uno de los $\Omega_{R,i}$ se construye una generica aproximación estandar $\widetilde{P}_i = \cup_j \alpha_j A_{i,j}$ y su información difusa $\overline{J}(\widetilde{P}_i)$ que al final tiene la forma

$$\overline{J}(\widetilde{P}_i) = \overline{f}^{(-1)}\left[\sum k(\alpha_j) \cdot f_i \circ J(A_{i,j})\right]$$

donde $f_i \circ J$ es una medida o aditiva o bien pseudo aditiva. Pasando al extremo inferior se obtienen los teoremas 3.7.a y 3.7.b en los dos casos.

# 5 REFERENCIAS


- [1] B.Forte, J.Kampé de Feriet : *Theorie de l'information; information et probabilité*, C.R.A.S, Paris, t.265, Serie A (1965), pp.111-114.
- [2] C.Bertoluzza, A.Bodini : *Información de un conjunto difuso: el caso discreto*, Proceedings of ESTYLF'96.
- [3] S.Weber ; $\perp$-*Decomposable measures and integrals fot archimedean t-conorms*, J.Math.An.Appl, 101 (1984), pp.114-138.
- [4] D.Cariolaro : *Un'estensione del concetto di integrale sfumato rispetto a misure compositive*, Tesi di laurea, Universitá di Pavia (1995).

# FUSIÓN DE ATRIBUTOS CON TÉCNICAS FUZZY EN DATA MINING

**David Nettleton**
Departamento de Lenguajes y Sistemas Informáticos.
*Universidad Politécnica de Catalunya*
netleton@lsi.upc.es

**Karina Gibert Oliveras**
Departamento de Estadística e Investigación Operativa.
*Universidad Politécnica de Catalunya*
karina@eio.upc.es

## Resumen

Se presenta un método de fusión de atributos que parte de una matriz de datos definidos en términos difusos, solventando algunos problemas que tiene el algoritmo 'crisp', frente a datos mixtos, es decir, atributos continuos, categóricos ordinales, nominales e incluso difusos. El enfoque difuso aporta la plataforma común para tratar todas las variables de forma homogénea. Se detalla la interpretación en el contexto difuso, de datos de admisiones hospitalarias.

**Palabras Clave:** Fusión de Atributos. Matriz de Covarianzas. Atributos Continuos, Categóricos Ordinales, Nominales e Difusos.

## 1. INTRODUCCIÓN

*Data Mining* es un proceso de análisis de datos de naturaleza inter-disciplinaria, cuyo propósito es identificar y extraer conocimiento de alto valor a partir de grandes volúmenes de datos. Los conjuntos de datos pueden ser muy voluminosos, complejos (muchos atributos), de estructura no muy evidente, y pueden presentar valores 'missing', errores y 'ruido' [9]

Data Mining utiliza técnicas diversas para el análisis y proceso de datos:

- Estadística clásica (regresión lineal, correlación, etc. ..).
- Algoritmos de aprendizaje para clasificación y predicción (Inducción, Redes Neuronales, etc. ..).
- Exploración de los datos mediante herramientas de visualización gráfica y manipulación.

El artículo se estructura en 6 secciones, la primera de las cuales es una introducción al ámbito de Data Mining. En las Secciones 2 y 3 se explica el proceso de fusión de atributos complejos y su problemática, como consecuencia de lo cual surge la propuesta del enfoque difuso. En la Sección 4 se presenta un método para representar de forma *fuzzy* algunas de las variables. En la Sección 5 se presenta una aplicación real a datos de 'admisiones hospitalarias' y en la Sección 6 se termina con un resumen del enfoque.

## 2. LA FUSIÓN DE ATRIBUTOS

Los métodos de fusión de atributos fueron introducidos por Hartigan en 1975 [6]. Una de sus aplicaciones es la reducción del conjunto de atributos inicial a un espacio de dimensión 2 ó 3 que facilite, por ejemplo, la visualización de datos. Esto permite una interpretación más sencilla y clara, y es desde este punto de vista, que tienen interés en el contexto de Data Mining.

El algoritmo de fusión cumple dos objetivos. El primero es la reducción de atributos mediante su progresiva unificación; el segundo es la identificación de los factores

más significativos y los factores entre los cuales haya mayor relación.

En realidad, un proceso de fusión de atributos se puede ver como un método ascendente de clasificación (*cluster*) de variables, donde cada *cluster* representa grupos de variables que se pueden fusionar. Las covarianzas entre las variables juegan aquí un papel fundamental.

*Grosso modo*: en cada paso, se *fusiona* el par de atributos con las covarianzas más altas para formar nuevos atributos, hasta obtener el número de atributos deseado, o hasta concluir un árbol binario con las agrupaciones realizadas. Es a partir de este árbol de atributos fusionados que se podrán seleccionar distintas descripciones de los objetos a analizar; descripciones con la dimensión más conveniente en cada caso.

## 3. LA FUSIÓN EN DOMINIOS COMPLEJOS

En el análisis de dominios complejos, es habitual encontrarse con matrices de datos mixtas [4,5] en las que se mezclan, de forma natural, atributos continuos, categóricos ordinales, nominales e incluso difusos. Sin embargo, la reducción de la dimensionalidad de la matriz de datos, sigue siendo un objetivo a alcanzar.

Ya hemos dicho que el punto de partida en un proceso de fusión de atributos es el cálculo de la matriz de covarianzas entre dichos atributos. El problema que se plantea cuando se quiere analizar una matriz mixta es que la covarianza (que mide la asociación lineal entre las variables) solamente está definida para pares de variables numéricas. Es cierto que para pares de variables ordinales, por ejemplo, existen algunas medidas de asociación - como el coeficiente de correlación de Spearman o el de Kendall [7,11] - que podrían jugar un papel parecido al de las covarianzas. Incluso la distancia de $\chi^2$ [7] se podría utilizar en pares de variables categóricas. Pero en cualquier caso, lo que queda por definir es cómo construir el equivalente a la matriz de covarianzas, cuando los datos vienen descritas simultáneamente por variables de distintos tipos.

Si además se tiene en cuenta que muchas veces algunos atributos son difusos por naturaleza (ver Sección 5), el problema aumenta su complejidad.

En este trabajo abordamos la aplicación de la fusión de atributos a dominios tales que:
(a) la descripción de los objetos puede estar sujeta a incertidumbre (atributos difusos).
(b) se trata simultáneamente información de naturaleza diversa (numérica, simbólica, ...).

## 4. SOBRE LA REPRESENTACIÓN DIFUSA DE LAS VARIABLES

En efecto, en muchas aplicaciones reales, la representación difusa es la que mejor se adecúa a las variables que se quiere *medir*.

Por ejemplo, en la Sección 5 se presenta una aplicación real relacionada con la admisión de pacientes en un hospital, en la que existen diversas variables para las que una representación difusa se ajustaría bien al proceso de toma de decisión de los expertos médicos. Este es el caso de las variables *'Tipo de Paciente'* e *'Infección probable a admisión a la UCI'*, donde el valor que se asigna a cada individuo se determina a partir de un proceso intuitivo. Un caso difícil de categorizar se representaría con un grado de pertenencia a las posibles clases situado alrededor de 0.50. y correspondería a unos síntomas que no son característicos de ninguna de las categorías disponibles.

Esta es una situación relativamente frecuente en el entorno médico. En [3], se detalla un ejemplo de 107 vectores de datos con valores binarios, cada uno con 11 atributos. Los atributos son síntomas considerados clínicamente relevantes para pacientes que sufren de dolor abdominal causado por o (i) hernia de hiato o (ii) cálculo biliar, en lo cual se consigue identificar casos con grados de pertenencia ambiguos.

Teniendo en cuenta que el enfoque difuso puede aportar la plataforma común para tratar todas las variables de una forma homogénea, nuestra propuesta pasa por adaptar el algorítmo de fusión de atributos propuesto en [6] para que se pueda aplicar a matrices de variables difusas. La Sección 4 indica como se representa las variables difusas en los distintos casos. A partir de esta representación, se puede utilizar un cálculo difuso de las covarianzas [3] que permite decidir en cada paso, qué variables fusionar.

En principio, la representación que adoptamos es el uso, ya habitual, de conjuntos difusos con función de pertenencia trapeziodal. En [1] se presenta un método que automáticamente construye un sistema de 5 etiquetas linguísticas que representan los valores ordenados de la variable a partir de lo que se llama *par parmenideano*, lo cual responde a los valores antagónicos que toma basicamente la variable. Este método resulta apto en variables como 'Días de Estancia', para lo cual se pueden

definir los valores difusos MUY CORTA, CORTA , MEDIA, LARGA, MUY LARGA a partir del antagonismo básico CORTA, LARGA.

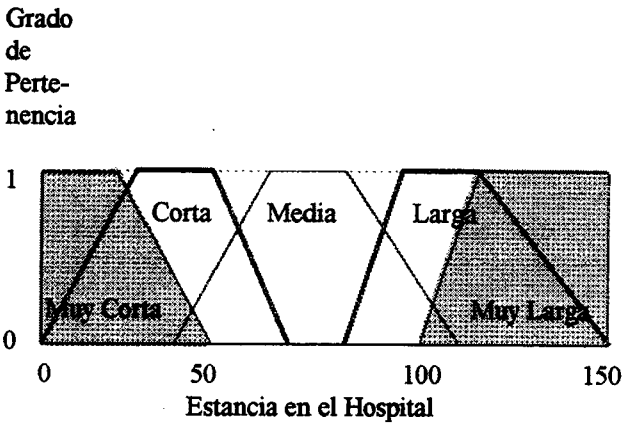



Figura 1: Representación de Variables Ordinales.

La Figura (1) muestra la representación difusa para la variable *'Días de Estancia en el Hospital'*.

## 5 APLICACIÓN: 'ADMISIONES-HOSPITALARIAS'

En [10] se analiza, utilizando diversos métodos, un conjunto de datos de 'admisiones hospitalarias' suministrado por un Hospital de Barcelona. Son datos complejos, dado el elevado número de atributos, pero la calidad, estadísticamente hablando, es buena (se dispone de una muestra bastante representativa, hay pocos valores desconocidos, etc.).

### 5.1 LOS DATOS

La matriz de datos a analizar contiene unos 1100 casos, descritos por unos 100 atributos cada uno. Los primeros 17 atributos son los signos vitales y otros datos generales del paciente. El resto de los atributos son derivados del análisis de una muestra de orina y una muestra de sangre que se toma en todos los pacientes bajo evaluación de admisión a la Unidad de Cuidados Intensivos del Hospital.

Los pacientes vienen descritos por las siguientes variables:

(1) Urea Sanguínea (mMol/l o mg/dl),
(2) Número de Sistemas de Organos fallando (N),
(3) Incremento de Creatinina > 124 Mol/l en últimas 24 horas asociado con Oliguria {si, no},
(4) Fallo Respiratorio{si, no},
(5) Fallo Neurológico {si, no},
(6) Coma o estupor profundo a 24 horas {si, no},
(7) Fallo Renal{si, no},
(8) Fallo Cardiovascular {si, no},
(9) Fallo Renal Agudo{si, no},
(10) Estado Anterior de Salud {1,2,3,4},
(11) Tipo de Paciente {Emergencia Cirugía, Cirugía Planificada, Sin Cirugía},
(12) Infección probable a admisión en la UCI {si, no},
(13) Sexo {V, M},
(14) Estancia en UCI para 24 horas o más {si, no},
(15) Estancia en días en el hospital (N),
(16) Estado vital a la salida de la UCI {muerto, vivo}.

La primera variable es de cáracter claramente numérico, mientras variables como (11) son ordinales y otras como (10) y (13) son nominales. Finalmente, variables como (15) y (16) se pueden representar en realidad como variables difusas, debido a la dificultad de consenso por parte de los expertos, y sus complejas dependencias con otras muchas variables en los distintos entornos.

### 5.2 ANÁLISIS

Mediante un estudio histórico de admisiones a UCI de este hospital, de las cuales se conoce si los pacientes fallecieron o no y, caso de no fallecer, cuál fue su proceso de recuperación, se puede extraer información útil en el apoyo a la toma de decisiones. Incluso se puede abordar la creación de modelos predictivos para el *riesgo de fallecimiento* (esto se hizo con una precisión resultante del 80%), y de los *días de estancia*.

Un segundo objetivo es la identificación de relaciones entre atributos, así como las relaciones entre las variables de entrada y el *riesgo de fallecimiento* o los *días de estancia*, que actúan como variables respuesta.

La aplicación del algorítmo de fusión de atributos a estos datos produce una secuencia de atributos fusionados. Ello permite ver qué atributos están relacionados entre sí, que es parte del segundo objetivo citado al principio de la Sección 2. Con los niveles superiores del árbol de fusión se pueden representar los datos en dimensiones reducidas a 2 ó 3 de tal forma que podemos estudiar la forma de los *clusters* con herramientas gráficas, para mejor entender su estructura.

## 6 RESUMEN

En resumen, el objetivo del trabajo es poder procesar datos que incorporan intrínsicamente grados de borrosidad, mejorando la precisión de selección de variables y su representación para este tipo de valores.

El trabajo está orientado hacia el proceso de datos clínicos y hospitalarios, y disponemos del consejo de expertos médicos para la interpretación de los resultados, manteniendo el objetivo de obtener resultados realmente útiles en el entorno real.

**Referencias**


[1] Aguilar-Martín, J. , Gibert-Oliveras, K., Sobre Variables Linguisticas Difusas, Paradigmas Parmenidianos y Logicas Multivaluadas. Primer Congreso Español sobre Tecnologías y Lógica Fuzzy. Universidad de Granada (1991) pp185-192.

[2] Béjar, J. "Adquisición de Conocimientos en Dominios Poco Estructurados". PhD Thesis. Departament of Computer languages and Systems, University Polytechnic of Catalunya, 1994.

[3] Bezdek, J.C. *"Pattern recognition with Fuzzy Objective Function Algoritms"*. Plenum Ress, 1981.

[4] Gibert, K., Cortés, U. *"Combining a Knowledge Based System and a Clustering Method for a Construction of Models in Ill-Structured Domains."*. Cap. V Selecting Models from Data. Lecture Notes in Statistics vol. 89. P Cheeseman and R.W. Oldford Eds. Springer-Verlag. New York, 1994. pp351-360.

[5] Gibert, K., Cortés, U. *"Weighing Quantitative and Qualitative Variables in Clustering Methods" (en prensa)*.. Journal Mathware and Soft Computing 4 (2), número especial de Gener 1997.

[6] Hartigan, J.A. *"Clustering Algorithms"*. Wiley, 1975.

[7] Lebart L., Morineau A., Fénelon J.P., *"Tratamiento Estadistico de Datos"*. Marcombo, 1985.

[8] Manton. K.G, Woodbury, Max. A. Statistical Applications using Fuzzy Sets. John Wiley & Sons, Inc. 1992.

[9] Nakhaeizadeh, G. *"Classification as a Subtask of Data Mining: Experiences from some Industrial Projects"*. IFCS-96. Fifth Conference of International Federation of Classification Societies, 1996.

[10] Nettleton, D.F. *"Data Mining en el Entorno Hospitalario"*. Novatica, Nº. 124, 1996.

[11] Peña, D. *"Estadistica Modelos y Métodos"*. Alianza, 1984.

[12] Zadeh, L.A. *"Outline of a New Approach to the Analysis of Complex Systems and Decision Processes"*. IEEE Trans. Syst., Man, Cybern., 1973.

# DEFINICIÓN DE PARÁMETROS ESTADÍSTICOS BIDIMENSIONALES BASADOS EN LA INTEGRAL DE CHOQUET

**Manuel Jorge Bolaños Carmona**
Dpto. Estadística e I.O.
Universidad de Granada
e-mail: mjbc@robinson.ugr.es

**Fernando Reche Lorite**
Dpto. Estadística y Matemática Aplicada
Universidad de Almería
e-mail: freche@stat.ualm.es

## Resumen

En este trabajo abordamos el problema de la evaluación estadística de información proviniente de varias fuentes en un contexto no-aditivo. Para ello presentamos algunos coeficientes que nos permiten estudiar el comportamiento de dos funciones definidas sobre referenciales finitos sobre los que hay subyacente una medida difusa.

**Palabras Clave:** Medidas difusas, Integral de Choquet, Esperanza Monótona, Esperanza Común, Esperanza Global, Coeficiente de Concordancia

## 1 INTRODUCCIÓN

Las medidas difusas (no-aditivas) han sido ampliamente estudiadas [15], así como los operadores conocidos como "integrales difusas" o "integrales no-aditivas" [3, 4, 6, 10, 11], entre las que destacan la integral de Sugeno y la de Choquet. Por contra, los trabajos destinados a su utilización para el análisis estadístico de datos son mucho más escasos [2, 10, 11], tal vez porque el paradigma de la aditividad está presente en la Estadística a través de las frecuencias y las probabilidades; sin embargo, podría ser enormemente útil el establecimiento de una metodología que permitiera tratar datos objetivos o subjetivos en el contexto no-aditivo de las medidas difusas utilizando integrales difusas en el papel de la suma o la integración habituales.

En un trabajo anterior [2], los autores abordaron la definición de las medidas paramétricas fundamentales para una muestra estadística sobre una variable, como la esperanza, la varianza o los momentos, en el contexto general de las medidas difusas.

Se prefirió utilizar la integral de Choquet frente a otras alternativas porque sus características se asemejan más o generalizan mejor las de la integración clásica.

En el presente trabajo, se aborda el problema bidimensional en un doble enfoque (dos muestras de una misma variable o de dos variables) y con un doble objetivo (estudiar la relación entre las muestras o variables y obtener una agregación o acumulación de información); dada la complejidad de las cuestiones planteadas, debe considerarse que lo que sigue es, en todo caso, una primera aproximación al problema.

En el apartado 2 se recogen las definiciones y conceptos ya conocidos y en el 3 se exponen las dificultades para generalizar los resultados clásicos de forma directa y además se indican más concretamente los objetivos de lo que sigue. La definición, propiedades y ejemplo de algunos coeficientes que permiten analizar el caso de dos funciones sobre el mismo referencial se recogen en el apartado 4, mientras que en el 5 se establece la forma de medir la composición de funciones, que se utiliza en el 6 para la definición de medidas globales, ahora en el caso general de espacios distintos, de los que el caso del apartado 4 es particular para uno sólo; así mismo, se muestran en el apartado 6 los resultados del ejemplo que se presentó anteriormente. Finalmente, el apartado 7 refleja algunas conclusiones provisionales y líneas de trabajo futuro.

## 2 CONCEPTOS Y DEFINICIONES CONOCIDAS

Sea $X = \{x_1, x_2, \ldots, x_n\}$, diremos que $g : \mathcal{A} \longrightarrow [0,1]$, siendo $\mathcal{A}$ un álgebra[1], es una ***medida difusa*** si cumple:

1. $g(\emptyset) = 0$ y $g(X) = 1$
2. Si $\forall A, B \in \mathcal{A}$ tales que $A \subset B$, entonces $g(A) \leq g(B)$

[1] habitualmente consideraremos $\mathcal{A} = \mathcal{P}(X)$

En lo sucesivo, consideraremos siempre un referencial $X$ finito.

Sea $(X, \mathcal{A}, g)$ un espacio de medida y una función $h : X \longrightarrow \mathbb{R}^+$, se define la *integral de Choquet* de $h$ con respecto a $g$ como

$$\int g(H_\alpha)d\alpha = \sum_{i=1}^{n} g(H_{\alpha_i})(\alpha_i - \alpha_{i-1}) \qquad (1)$$

donde $H_\alpha$ son los $\alpha$-cortes de $h$, es decir, $H_\alpha = \{x \in X/h(x) \geq \alpha\}$, $\alpha_i = h(x_i)$, $\alpha_0 = 0$, y podemos suponer, sin pérdida de generalidad, que $0 = \alpha_0 \leq \alpha_1 \leq \alpha_2 \leq \ldots \leq \alpha_n$, lo cual nos simplifica la notación.

La Integral de Choquet tambien es conocida con el nombre de *Esperanza Monótona* [6] de la función $h$ con respecto a la medida difusa $g$, la cual notaremos por $E_g(h)$ y cuyas propiedades son una extensión de las de la Esperanza Matemática [3, 7].

En [2] se planteó utilizar la integral de Choquet para la definición de parámetros estadísticos univariantes sobre medidas difusas. A la ampliamente conocida Esperanza Monótona citada, se añadieron los conceptos de varianza y momentos, cuando sobre el conjunto de referencia hay establecida una medida difusa. Para ello nos basamos en el concepto de Función de Probabilidad asociada a una medida difusa en la forma que se recoge a continuación:

Sea $(X, \mathcal{A}, g)$ un espacio de medida y $h$ una función de $X$ en $\mathbb{R}^+$, se define la *Función de Probabilidad asociada a la medida $g$ y a la función $h$* como el conjunto $P_{g,h} = \{p_1, p_2, \ldots, p_n\}$ donde

$$\begin{array}{l} p_i = g(H_{\alpha_i}) - g(H_{\alpha_{i+1}}) \quad si \quad i < n \\ p_n = g(H_{\alpha_n}) \end{array} \qquad (2)$$

$\alpha_i = h(x_i)$, $p_i = p(x_i)$ y $H_{\alpha_i} = \{x \in X/h(x) \geq \alpha_i\}$

Asi pues, utilizando esta función se puede comprobar de forma inmediata que $E_g(h) = E_P(h)$, y además se define una extensión de la varianza clásica como:

Sea $(X, \mathcal{A}, g)$ un espacio de medida y $h : X \longrightarrow \mathbb{R}^+$; se definen

1. La *Varianza Monótona* de $h$ con respecto a la medida difusa $g$ como

$$\mathrm{Var}_g(h) = \sum_{i=1}^{n} [h(x_i) - E_g(h)]^2 p_i \qquad (3)$$

con $p_i \in P_{g,h}$.

2. El *Momento Monótono respecto al origen* de orden $k$ de la función $h$ considerando la medida difusa $g$ de la forma

$$m_g^k(h) = E_g(h^k) = \sum_{i=1}^{n} h^k(x_i)p_i \qquad (4)$$

donde $p_i \in P_{g,h}$

3. El *Momento Monótono Central* de orden $k$ de la función $h$ con respecto a la medida difusa $g$ de la forma

$$\mu_g^k(h) = \sum_{i=1}^{n} (h(x_i) - E_g(h))^k p_i \qquad (5)$$

donde $p_i \in P_{g,h}$

Un estudio más pormenorizado de las propiedades de estos parámetros se pueden ver en [2].

# 3 EL PROBLEMA BIDIMENSIONAL

Los parámetros referidos en el apartado anterior nos permitieron generalizar las medidas de posición y de dispersión que aparecen en la Estadística Descriptiva Clásica de una variable, habitualmente definidas para probabilidades, al caso en que se presenta una medida difusa cualquiera sobre el referencial.

En este contexto cabe insistir en que los valores de la variable estudiada dados por la función $h$ están asociados a cada uno de los elementos del referencial $X$, y en que la clave de los parámetros estadísticos que nos miden esa función $h$ es la función de probabilidad asociada a $g$ y $h$.

Surge de modo inmediato el interés de acumular información procedente de diversas fuentes (varias valoraciones de expertos, resultados de distintas mediciones, etc...); en particular, sería interesante poder agregar la información proporcionada por dos funciones $h_1$ y $h_2$.

Dado que los valores de estas dos funciones están apareados de forma natural por su asociación a los elementos de $X$ $(h_{1i} = h_1(x_i), h_{2i} = h_2(x_i), \quad i = 1, \ldots, n)$, parece natural "promediar" punto a punto tales valores obteniendo una nueva función $\overline{h}$. Sin embargo, tal proceder carece de sentido en este contexto no aditivo ya que no se puede ni siquiera garantizar que $E_g(\overline{h})$ tenga un valor intermedio entre $E_g(h_1)$ y $E_g(h_2)$, salvo que $h_1(x) \leq h_2(x)$ $\forall x \in X$. Por lo tanto habrá que considerar otros mecanismos para, en algún sentido, agregar o combinar información.

Por otra parte, el citado apareamiento de los valores de la función plantea el interés de establecer parámetros que reflejen el grado de asociación que hay entre $h_1$ y $h_2$, en la línea de generalizar los resultados ya conocidos en el ambiente aditivo (en este caso $h_1$ y $h_2$ se contemplarán como valores correspondientes a dos variables).

Si generalizamos el concepto de covarianza de la forma $E_g[(h_1 - E_g(h_1))(h_2 - E_g(h_2))]$, la Función de Probabilidad asociada que se utiliza para este cálculo no tiene, en general, relación alguna con las asociadas a $h_1$ o a $h_2$ para la misma medida; en consecuencia, ni ésta, ni otra definición similar a la de la varianza (fijando una función de probabilidad determinada) tiene utilidad para medir la relación buscada entre ambas funciones, ya que a partir de ellas no podría definirse algo similar a los coeficientes de correlación habituales. Es necesario, pues, enfocar el asunto desde otra perspectiva.

# 4 ESPERANZA COMÚN Y COEFICIENTE DE CONCORDANCIA

Una primera aproximación al problema de la relación entre dos funciones definidas sobre el mismo referencial puede ser la medición de sus coincidencias relativas, lo que abordaremos mediante la construcción de un índice que exprese similaridad entre ambas funciones, para lo que usaremos la medición de su parte común en la forma que se recoge en las siguientes definiciones:

**Definición** Sea $(X, \mathcal{P}(X), g)$ un espacio de medida y $h_1$, $h_2$ dos funciones de $X$ en $\mathbb{R}^+$; se denomina *Esperanza Común* de esas dos funciones con respecto a $g$ a la esperanza monótona de la función $\min(h_1, h_2)$, esto es,

$$CE_g(h_1, h_2) = E_g[\min(h_1, h_2)] \tag{6}$$

**Definición** Sea $(X, \mathcal{P}(X), g)$ un espacio de medida y $h_1$, $h_2$ dos funciones de $X$ en $\mathbb{R}^+$; puede definirse un *Coeficiente de Concordancia* entre $h_1$ y $h_2$ de cualquiera de las formas siguientes:

$$CC1_g(h_1, h_2) = \frac{CE_g(h_1, h_2)}{\sqrt{E_g(h_1)E_g(h_2)}} \tag{7}$$

$$CC2_g(h_1, h_2) = \frac{CE_g(h_1, h_2)}{E_g(\max(h_1, h_2))} \tag{8}$$

$$CC3_g(h_1, h_2) = \frac{CE_g(h_1, h_2)}{\min(E_g(h_1), E_g(h_2))} \tag{9}$$

Estos tres coeficientes coinciden en expresar de forma relativa la Esperanza Común de las dos funciones, que nos dice hasta qué punto coinciden o concuerdan $h_1$ y $h_2$, mientras que los denominadores son posibles valores de normalización, para que el coeficiente varíe entre 0 y 1.

**Propiedades:**

Sea $(X, \mathcal{P}(X), g)$ un espacio de medida y $h_1$, $h_2$ dos funciones de $X$ en $\mathbb{R}^+$, entonces se cumplen las siguientes propiedades:

1. Si $h_1(x_i) = h_2(x_i)\ \forall x_i \in X$ entonces $E_g(h_1) = E_g(h_2) = CE_g(h_1, h_2)$, y los tres coeficientes de concordancia valen 1.

2. Si $h_1$ y $h_2$ tienen intersección vacía, es decir, $\{x \in X / h_1(x) > 0\} \cap \{x \in X / h_2(x) > 0\} = \emptyset$, entonces la esperanza común y los tres coeficientes de concordancia se anulan.

3. Si $h_1(x_i) \leq h_2(x_i)\ \forall x_i \in X$ entonces $E_g(h_1) = CE_g(h_1, h_2)$ y se tiene que:

$$CC1_g(h_1, h_2) = \sqrt{\frac{E_g(h_1)}{E_g(h_2)}} \tag{10}$$

$$CC2_g(h_1, h_2) = \frac{E_g(h_1)}{E_g(h_2)} \tag{11}$$

$$CC3_g(h_1, h_2) = 1 \tag{12}$$

4. Si $h_1(x_i) = kh_2(x_i)\ \forall x_i \in X$ y $k > 1$, entonces $E_g(h_2) = CE_g(h_1, h_2)$ y se tiene:

$$CC1_g(h_1, h_2) = \frac{1}{\sqrt{k}} \tag{13}$$

$$CC2_g(h_1, h_2) = \frac{1}{k} \tag{14}$$

$$CC3_g(h_1, h_2) = 1 \tag{15}$$

Aunque estos coeficientes no puedan considerarse como una generalización de coeficientes de correlación, sí reflejan el grado de coincidencia y, en cierto modo, el grado de asociación de las funciones implicadas. Como hemos visto, se anulan en el caso de que los dominios no nulos sean disjuntos y se hacen máximos cuando hay una coincidencia total en los valores de las funciones. La utilización en la práctica de uno u otro coeficiente dependerá de la importancia que se le quiera dar a la esperanza de cada función. Mientras *CC1* utiliza para normalizar los valores de ambas integrales, *CC3* solo considera la integral más pequeña, lo cual no permitiría distinguir situaciones distintas en la otra función y, finalmente. *CC2* no utiliza ninguna de las dos funciones originales, sino la esperanza de la función máximo. Por lo tanto, este último parece reflejar mejor el nivel de coincidencia, mientras que *CC1* parece indicar más un nivel de asociación y, formalmente, recuerda a las definiciones habituales de los coeficientes de determinación o correlación.

*Ejemplo:* El trabajo de dos alumnos va a ser globalmente evaluado en función de 4 materias *A, B, C* y *D*, para lo cual se quiere valorar especialmente el rendimiento en las dos científicas (*A,B*), las dos humanísticas (*C,D*) o en la asociación (*B,C*), que corresponde a una orientación hacia las ciencias sociales.

Así la medida $g$ que consideramos la podemos ver en la Tabla 1 y la puntuación que obtuvieron los dos alumnos en la Tabla 2. Realizando los cálculos podemos

Tabla 1: Medida difusa

| Conjuntos | Medidas |
|---|---|
| A ó B ó C ó D | 0.2 |
| AB | 0.6 |
| AC ó AD | 0.3 |
| BC | 0.5 |
| BD | 0.4 |
| CD | 0.7 |
| ABC | 0.9 |
| ABD | 0.6 |
| ACD | 0.7 |
| BCD | 0.8 |

Tabla 2: Puntuación de los alumnos

| Alumno | A | B | C | D |
|---|---|---|---|---|
| $h_1$ | 0.9 | 0.8 | 0.3 | 0.2 |
| $h_2$ | 0.2 | 0.3 | 0.8 | 0.9 |

ver que:

$E_g(h_1) = 0.61$, $\mathrm{Var}_g(h_1) = 0.0769$, $E_g(h_2) = 0.65$ y $\mathrm{Var}_g(h_2) = 0.0765$.

Por lo tanto, ambas funciones tienen un promedio y una varianza de puntuación muy similar.

La esperanza común vale $CE_g(h_1, h_2) = 0.25$ y si calculamos los coeficiente de concordancia obtenemos:

$CC1_g(h_1, h_2) = 0.397$

$CC2_g(h_1, h_2) = 0.301$

$CC3_g(h_1, h_2) = 0.410$

Los coeficientes de concordancia tienen valores bajos, lo cual es razonable, ya que ambos alumnos tienen un perfil bastante diferente, puesto que el primero es un alumno "de ciencias" y el segundo "de letras"; por lo tanto se puede entender que hay una pequeña concordancia.

## 5 MEDIDA DE LA COMPOSICIÓN DE FUNCIONES

Puesto que no es posible establecer con carácter general medidas de promedios de las funciones, trataremos de abordar la agregación de información desde el punto de vista de la composición. Asi pues, a partir de dos funciones definidas sobre dos espacios de medida obtendremos una especie de función compuesta que nos exprese su valoración conjunta. Como caso particular, cuando los dos espacios de medida coincidan podremos abordar la valoración de las dos funciones $h_1$ y $h_2$ que nos planteabamos en los apartados anteriores, y previamente a la composición de las funciones hemos de obtener una medida compuesta sobre el espacio $X \times Y$.

**Definición** Sean $(X, \mathcal{P}(X), g_X)$ e $(Y, \mathcal{P}(Y), g_Y)$ dos espacios de medida; diremos que $g^* : \mathcal{P}(X \times Y) \longrightarrow [0,1]$ es la medida compuesta de $g_X$ y $g_Y$ si

$$g^*(C) = \min(g_X(A), g_Y(B)) \tag{16}$$

donde

$C \in \mathcal{P}(X \times Y)$,

$A = \{x \in X / (x,y) \in C \text{ para algún } y\}$

$B = \{y \in Y / (x,y) \in C \text{ para algún } x\}$

($A$ y $B$ son las proyecciones de $C$ sobre $X$ e $Y$ respectivamente).

**Proposición** $(X \times Y, \mathcal{P}(X \times Y), g^*)$ es un espacio de medida.

*Dem.*

Es obvio que $g^*(X \times Y) = 1$ y $g^*(\emptyset) = 0$

Sea $C_1, C_2 \in \mathcal{P}(X \times Y)$ de tal forma que $C_1 \subset C_2$, entonces las proyecciones cumplen que $A_1 \subset A_2$ y $B_1 \subset B_2$. Como $g_X$ y $g_Y$ son medidas difusas ocurre que $g_X(A_1) \leq g_X(A_2)$ y $g_Y(B_1) \leq g_Y(B_2)$; por lo tanto, $g^*(C_1) \leq g^*(C_2)$, por lo que $g^*$ es una medida difusa.

**Definición** Sean $h_1 : X \longrightarrow \mathbb{R}^+$ y $h_2 : Y \longrightarrow \mathbb{R}^+$ dos funciones. Diremos que $h^* : X \times Y \longrightarrow \mathbb{R}^+$ es la función compuesta de $h_1$ y $h_2$ si

$$h^*(x,y) = \min(h_1(x), h_2(y)) \tag{17}$$

Hemos definido la medida $g^*$ al objeto de poder valorar las funciones compuestas. Parece claro que hay que exigir que la Esperanza Monótona quede invariante cuando componemos una función consigo misma. El siguiente resultado nos muestra que esa propiedad se verifica cuando usamos la medida y la función compuesta.

**Teorema** Sea $(X, \mathcal{P}(X), g)$ un espacio de medida, y sea $(X \times X, \mathcal{P}(X \times X), g^*)$ un espacio de medida donde $g^*$ es la medida compuesta de $g$ consigo misma.

Si $h^*(x,y) = \min(h(x), h(y))$ $x, y \in X$, entonces

$$E_{g^*}(h^*) = E_g(h) \tag{18}$$

*Dem.*

$$\begin{aligned} E_{g^*}(h^*) &= \sum_{i=1}^{n \times n} g^*(H_{\alpha_i})(\alpha_i - \alpha_{i-1}) \\ &= \sum_{i=1}^{n} \left[\min(g_X(H^X_{\alpha_i}), g_X(H^X_{\alpha_i}))\right](\alpha_i - \alpha_{i-1}) \end{aligned}$$

$$= \sum_{i=1}^{n} [g_X(H^X_{\alpha_i})] (\alpha_i - \alpha_{i-1})$$
$$= E_g(h)$$

donde

$$\begin{aligned} H_{\alpha_i} &= \{(x,y) \in X \times X / h^*(x,y) \geq \alpha_i\} \\ &= \{(x,y) \in X \times X / \min(h(x), h(y)) \geq \alpha_i\} \\ &= \{(x,y) \in X \times X / h(x) \geq \alpha_i \ y \ h(y) \geq \alpha_i\} \end{aligned}$$

por lo que la proyección $H^X_{\alpha_i}$ de $H_{\alpha_i}$ sobre $X$ es

$$\begin{aligned} H^X_{\alpha_i} &= \{x \in X / (x,y) \in H_{\alpha_i}\} \\ &= \{x \in X / h(x) \geq \alpha_i\} \end{aligned}$$

que son los $\alpha_i$-cortes de la función $h$, por lo que el resultado queda demostrado.

Este resultado nos justifica la elección del operador mínimo a la hora de hacer las composiciones, ya que de haber utilizado otra norma triangular no se verifica, en general, un resultado similar.

# 6 ESPERANZA GLOBAL Y COEFICIENTE DE CONCORDANCIA GLOBAL

A partir de las definiciones dadas para la composición de medidas y funciones, puede entenderse que la integración de Choquet de la función compuesta es una extensión del caso unidimensional, lo que nos permite definir nuevos parámetros y coeficientes, como los siguientes:

**Definición** Sea $(X \times Y, \mathcal{P}(X \times Y), g^*)$ un espacio de medida obtenido por composición de los espacios $(X, \mathcal{P}(X), g_X)$ e $(Y, \mathcal{P}(Y), g_Y)$ de la forma definida anteriormente, y sean $h_1 : X \longrightarrow \mathbb{R}^+$ y $h_2 : Y \longrightarrow \mathbb{R}^+$, cuya función compuesta notaremos por $h^* : X \times Y \longrightarrow \mathbb{R}^+$. En estas condiciones, diremos que

$$GE_g(h_1, h_2) = E_{g^*}(h^*) \tag{19}$$

es la *Esperanza Global* de las funciones $h_1$ y $h_2$ respecto de las medidas $g_X$ y $g_Y$ (o bien, respecto a la medida compuesta $g^*$).

En base a la Esperanza Global (denominada así porque refleja una valoración conjunta de las dos funciones, y no de la parte común, como en el caso unidimensional) pueden obtenerse también coeficientes que relativicen su valor respecto a la esperanza de cada una de las funciones consideradas individualmente; entre ellos, parecen tener especial interés los recogidos en la definición siguiente:

**Definición** En las condiciones de la definición anterior, decimos que:

$$CCG1_{g^*}(h_1, h_2) = \frac{GE_{g^*}(h_1, h_2)}{\sqrt{E_{g_X}(h_1) E_{g_Y}(h_2)}} \tag{20}$$

$$CCG3_{g^*}(h_1, h_2) = \frac{GE_{g^*}(h_1, h_2)}{\min(E_{g_X}(h_1), E_{g_Y}(h_2))} \tag{21}$$

son *Coeficientes de Concordancia Global* entre $h_1$ y $h_2$ respecto a las medidas $g_X$ y $g_Y$ (ó $g^*$).

La definición de estos coeficientes se puede ampliar también a la utilización en el denominador de una esperanza sobre una composición definida con el máximo (al estilo del *CC2* del apartado 4) o directamente del máximo entre las esperanzas de $h_1$ y $h_2$, pero tales posibles definiciones parecen tener menor significado numérico e interpretativo que las dadas en la definición, por otra parte suficientes para nuestro interés.

A pesar de su similitud formal, no es posible establecer un estrecho paralelismo entre las propiedades de la Esperanza Global y las de la Esperanza Común anteriormente definida, por lo que no es posible asegurar aquí las que allí se cumplían; en particular, de las cuatro propiedades recogidas en el apartado 4 para los coeficientes de concordancia, sólo la primera tiene validez para los coeficientes de concordancia global, referida al caso en que $X \equiv Y$ y $h_1(x_i) = h_2(x_i)\ \forall x_i \in X$; en este caso el $CCG_{g^*}$ es igual a 1 para cualesquiera de las definiciones dadas, como se deduce de forma inmediata del teorema probado en el apartado anterior.

*Ejemplo* (Continuación) En la Tabla 3 podemos ver la forma que tiene la función compuesta del ejemplo anterior En este caso, $GE_{g^*}(h^*) = 0.60$ y los

Tabla 3: Función compuesta de $h_1$ y $h_2$

| Asignatura | A | B | C | D |
|---|---|---|---|---|
| A | 0.2 | 0.3 | 0.8 | 0.9 |
| B | 0.2 | 0.3 | 0.8 | 0.8 |
| C | 0.2 | 0.3 | 0.3 | 0.3 |
| D | 0.2 | 0.2 | 0.2 | 0.2 |

valores de los coeficientes de concordancia global son $CCG1_{g^*}(h_1, h_2) = 0.953$ y $CCG3_{g^*}(h_1, h_2) = 0.984$.

Como se ve, la Esperanza Global alcanza en este ejemplo un valor muy próximo al de las valoraciones de cada alumno (0.60 por 0.61 y 0.65, respectivamente), indicando que, en términos generales, las puntuaciones de ambos alumnos son aceptables y su valoración conjunta también lo es, lo que da lugar a unos Coeficientes de Concordancia Global muy altos; obsérvese que la *GE* no recoge el hecho de que uno de los alumnos

"es de Ciencias" y el otro "de Humanidades", lo cual si hacía *CE*, que tenía un valor muy inferior (0.25), dando lugar a unos coeficientes de concordancia bastante bajos.

# 7 CONCLUSIONES Y LÍNEAS FUTURAS

Aunque no se haya seguido un paralelismo formal con los conceptos clásicos habituales, las valoraciones aquí definidas a partir de la integral de Choquet y los coeficientes que de ellas se obtienen pueden ocupar, al menos en parte, el hueco que parece existir en el terreno de la evaluación de la agregación y de la relación entre muestras sobre medidas difusas.

En el caso general $X \times Y$, realmente bidimensional, se pierde la asociación punto a punto que parece formar parte de la esencia de las aplicaciones hasta ahora conocidas de la integración difusa; la Esperanza Global y los coeficientes a ella asociados permiten una evaluación "conjunta" que puede compararse con la de las funciones "marginales".

En el caso particular $X \times X$ (como aquel en que $X$ e $Y$ tengan el mismo cardinal y una asociación "uno a uno" entre sus elementos), las diferencias entre las definiciones dadas están mucho más claras y permiten entender mejor el significado de las mismas. Así, mientras la Esperanza Común mide punto a punto la parte compartida por las dos funciones y los coeficientes de ella derivados pueden entenderse como respuestas a la pregunta ¿están asociadas ambas informaciones?, el concepto de Esperanza Global mide sin asociación puntual el "volumen" conjunto de ambas funciones y permite abordar la cuestión ¿cuánto vale la agregación de ambas informaciones?, lo que podemos también poner en relación con las evaluaciones individuales a través de los Coeficientes de Concordancia Global; además, la comparación de ambas esperanzas permite entender la fuerza de la relación entre las dos informaciones dadas.

La evaluación de funciones directamente definidas en espacios multidimensionales, el análisis de la estructura de los $\alpha$-cortes de las distribuciones conjuntas, la profundización en la semántica de los conceptos definidos y el estudio de su sensiblidad respecto de cambios de los valores de las funciones, son algunos de los temas de estudio que nos proponemos para un futuro próximo, y que en todo caso permacen abiertos para su análisis y proporcionarán nuevos enfoques del problema

## Referencias


[1] Bolaños, M.J, Hernández, L.D., Salmerón, A. (1996) Numerical experimentation and comparison of fuzzy integrals. *Mathware and Soft Computing* **3**, 309-319.

[2] Bolaños, M.J, Reche, F. (1996) Definición de parámetros estadísticos utilizando la Integral de Choquet. *Actas del Congreso ESTYLF96*, 47-51.

[3] Bolaños, M.J, de Campos, L.M., González, A. (1989) Convergence properties of the monotone expectation and its application to the extension of fuzzy measures. *Fuzzy Sets and Systems* **33**, 201-213.

[4] Bolaños, M.J, Lamata, M.T., Moral, S. (1987) Extensión de medidas difusas usando la esperanza monótona. *Stochastica* **XI-2,3**, 75-92.

[5] Choquet, G. (1953/54) Theory of capacities. *Ann. Ins. Fourier* (Grenoble) **5**, 131-293.

[6] de Campos, L.M, Bolaños, M.J. (1989) Representation of fuzzy measures through probabilities. *Fuzzy Sets and Systems* **31**, 23-36.

[7] de Campos, L.M, Bolaños, M.J. (1992) Characterization and comparison of Sugeno and Choquet integrals. *Fuzzy Sets and Systems* **52**, 61-67.

[8] de Campos, L.M., Lamata, M.T., Moral, S. (1990) The concept of conditional fuzzy measure *International Journal of Intelligent Systems* **5**, 237-246.

[9] Dubois, D., Prade, H. (1980) Fuzzy Sets and Systems: Theory and Application *Academic Press, Inc.*.

[10] Friedman, M., Schneider, M., Kandel,A. (1989) The use of weighted fuzzy expected value in fuzzy expert systems. *Fuzzy Sets and Systems* **31**, 37-45.

[11] Kandel, A. (1988) Theory and applications of fuzzy statistics. *In Combining fuzzy imprecision with probabilistic uncertainty in decision making* , 89-112. Lecture notes in Econom. and Math. Systems.

[12] Murofushi, T., Sugeno, M. (1989) An interpretation of fuzzy measures and the Choquet integral a fuzzy measure. *Fuzzy Sets and Systems* **29**, 201-227.

[13] Nguyen,H. (1978) On conditional possibility theory. *Fuzzy Sets and Systems* **1**, 299-310 .

[14] Suarez, F., Gil, P. (1986) Two families of fuzzy integrals. *Fuzzy Sets and Systems* **18**, 67-81.

[15] Wang, Z., Klir, G. (1992) Fuzzy Measure Theory. *Plenum Press.*

# On defuzzification with continuous WOWA operators

**Vicenç Torra[1], Lluís Godo[2]**

[1] Departament Enginyeria Informàtica (ETSE), Universitat Rovira i Virgili,
Carretera de Salou s/n, 43006 Tarragona (Spain). e-mail: vtorra@etse.urv.es
[2] Institut d'Investigació en Intel.ligència Artificial (IIIA), CSIC
Campus UAB s/n, 08193 Bellaterra (Spain). e-mail: godo@iiia.csic.es

## Abstract

Techniques of information aggregation are mainly being used in decision making or when a system needs a comprehensive representation of its application domain. Recently, aggregation operators have been also applied for defuzzification where the defuzzified value is calculated as an aggregation of all possible solutions according to their attached possibility. In this paper we propose and analyze the use of the so-called WOWA operators (and their extension to the continuous case), which generalize both Weighted Mean and OWA operators, for defuzzification.

**Keywords:** decision making, aggregation operators, fuzzy control, defuzzification methods, WOWA operators.

## 1 INTRODUCTION

Within the framework of Artificial Intelligence, techniques of information aggregation are mainly used (see [16] for more details) when a system has to make a decision or when it needs a comprehensive representation of its application domain. Typical examples of fields where these techniques are necessary are knowledge based systems [13], decision making [11], management [11], scheduling [14], vision [10], robotics [3] or economy [25]. Since aggregation functions are defined according to the objects to be synthesized, there exist several functions for each of the knowledge representation formalisms. For example, functions to combine probability distributions [7], may be different from functions to combine fuzzy sets [4, 2].

Recently [24, 22], aggregation operators have been also applied in the defuzzification process of fuzzy control systems. Actually, several defuzzification methods can be understood as a blending or mixing of different solutions: the defuzzified value is calculated as an aggregation of all possible solutions according to the possibility attached to each one.

In this paper we suggest and analyze the use of a class of aggregation operators which have been recently introduced, called WOWA operators, in the defuzzification process. WOWA operators generalize both Weighted Mean (WM) and Ordered Weighted Averaging (OWA) operators and are described in Section 2. In Section 3 we deal with the extension of WOWA operators to average continuous distributions, while in Section 4 we propose how to use such operators for defuzzifying fuzy sets. An example of their application is shown in Section 5. Finally, we end up in Section 6 with some remarks about related work and conclusions.

## 2 AVERAGING NUMERICAL VALUES WITH WOWA OPERATORS

When numerical values have to be aggregated, for instance in the case of fuzzy membership values, the basic operations which have been considered are the Weighted Mean (WM operator for short) and the Ordered Weighted Averaging operator (OWA operator for short). Given a

normalized weighting vector[1] $\mathbf{p} = [p_1,\ldots,p_n]$ of dimension n, the **Weighted Mean (WM) operator** of dimension n based on the weights **p** is the mapping $wm^{\mathbf{p}}: \mathbb{R}^n \to \mathbb{R}$ defined as

$$wm^{\mathbf{p}}(A) = \sum_{i=1}^{n} p_i \cdot a_i, \quad (1)$$

where $A = [a_1, \ldots, a_n]$ is a set of n numerical observations. Together with its general form (the quasi-arithmetic mean), the WM operators are studied in [1]. The OWA operators were introduced by Yager in [19, 20] and are defined as follows. Let **w** be a normalized weighting vector of dimension n and let A be a set of n numerical observations, $A = [a_1, \ldots, a_n]$. In such a case, a mapping $f^{\mathbf{w}}_{owa}: \mathbb{R}^n \to \mathbb{R}$ is an **Ordered Weighted Averaging (OWA) operator** of dimension n if

$$f^{\mathbf{w}}_{owa}(A) = \sum_{i=1}^{n} w_i \cdot a_{\sigma(i)} \quad (2)$$

where $\{\sigma(1),\ldots,\sigma(n)\}$ is a permutation of $\{1,..,n\}$ such that $a_{\sigma(i-1)} \geq a_{\sigma(i)}$ for all i=2, ..., n. OWA operators are analysed in [8, 6].

Both kinds of operators are to combine the values according to a single set of weights. Their difference is in the meaning they assign to these weights, and therefore how these weights are combined with the input values. See [16] for a detailed analysis of their difference. In short, on the one hand, the Weighted Mean allows the system to compute an aggregation of values for a variable coming from different sources taking into account the reliability of each information source. Alternatively, the OWA operator allows the user to weight the values supplied in relation to their relative ordering. In this way, a system can give more importance to a subset of the input values than to another subset. For example, as it is described in [21], the OWA operator can model how are determined the ratings in certain sports in an Olympic Games: five judges give their punctuation and the average of this values is performed after droping the extreme values). This corresponds to give zero importance to extreme values. Therefore, in the weighted mean, weights measure the importance of an information source with independence of the value that the source has captured. On the other hand, in the OWA operator weights measure the importance of a value (in relation to other values), independently from the information source that has captured it.

However, both types of weights can be needed in a single problem. As a matter of example [15], consider a robot with five sensors (whose reliability is known) to determine the distance to the nearest object in the direction of its movement. In this case, the data supplied by each source can be combined by means of the weighted mean. However, with independence of the information source that provides a measurement they are not equally important. The robot should give more importance to nearest objects than those that are further. For example, due to the orientation of the robot two of the readings can be very large. This situation is easily modelled by means of the OWA operator. The OWA operator uses a set of weights $\mathbf{w} = [w_1,\ldots,w_n]$ where $w_1$ is the weight corresponding to the greatest value, and $w_i$ corresponds to the i-th greatest value. In the particular case of the robot, a possible set of weights which gives more importance to nearest values could be **w** = [0, 0.1, 0.2, 0.3, 0.4]. To calculate an averaged distance and to make a decision the robot should consider both factors, the reliability of the sensors and the importance of small values. The Weighted OWA operator (WOWA operator for short) defined in [15, 16] allows the user to consider both aspects using two sets of weights. Namely, let **p** and **w** be two normalized weighting vectors of dimension n and let $A = [a_1, \ldots, a_n]$ be a set of n numerical observations. In such a case, a mapping $f^{\mathbf{w},\mathbf{p}}_{wowa}: \mathbb{R}^n \to \mathbb{R}$ is a **Weighted Ordered Weighted Averaging (WOWA) operator** of dimension n if

$$f^{\mathbf{w},\mathbf{p}}_{wowa}(A) = \sum_{i=1}^{n} \lambda_i \cdot a_{\sigma(i)} \quad (3)$$

where $\{\sigma(1),\ldots,\sigma(n)\}$ is a permutation of $\{1,..,n\}$ such that $a_{\sigma(i-1)} \leq a_{\sigma(i)}$ for all i=2, ..., n, and the weights $\lambda_i$'s are defined as

$$\lambda_i = W^*(\Sigma_{j\leq i}\, p_{\sigma(j)}) - W^*(\Sigma_{j<i}\, p_{\sigma(j)})$$

where $W^*: [0, 1] \to [0, 1]$ is a monotonic increasing function that interpolates the points $(i/n, \Sigma_{j\leq i}\, w_j)$ together with the point (0,0). $W^*$ is required to be a straight line when the points can be interpolated in this way (this is a technical condition).

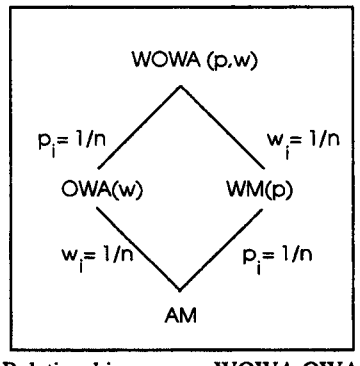



Figure 1. Relationships among WOWA,OWA and WM

Notice that WOWA operators generalize both WM and OWA operators. As it is shown in Figure 1, in the

[1]Hereafter by a *normalized weighting vector* $\mathbf{p}=[p_1 \ldots p_n]$ of dimension n we refer to a vector satisfying the following conditions: i) $p_i \in [0,1]$,and ii) $\Sigma_i\, p_i = 1$.

particular case that $\mathbf{w} = [1/n, ..., 1/n]$ we recover the WM operator and when $\mathbf{p} = [1/n, ..., 1/n]$ we recover the OWA operator. Finally if we take both $\mathbf{w} = \mathbf{p} = [1/n, ..., 1/n]$ what we get is just the arithmetic mean (AM).

To avoid constructing the functions $W^*$ by interpolation, there is an alternative definition of WOWA operators in terms of the Yager's *regular monotonically non-decreasing quantifiers* [21], that is, functions Q satisfying the same properties of $W^*$: non-decreasing, $Q(0) = 0$ and $Q(1) = 1$.

Then, if $\mathbf{p}$ is a normalized weighting vector of dimension n and Q is a regular monotonically non-decreasing quantifier, the **WOWA operator** of dimension n $\text{wowa}^{Q,\mathbf{p}}: \mathbb{R}^n \to \mathbb{R}$ induced by Q and $\mathbf{p}$ is the mapping defined as

$$\text{wowa}^{Q,\mathbf{p}}(A) = \sum_{i=1}^{n} \lambda_i \cdot a_{\sigma(i)}, \qquad (4)$$

where $\{\sigma(1),...,\sigma(n)\}$ is a permutation of $\{1,..., n\}$ such that $a_{\sigma(i-1)} \le a_{\sigma(i)}$ for all i=2, ..., n, and the weights $\lambda_i$'s are defined as

$$\lambda_i = Q(\Sigma_{j \le i} p_{\sigma(j)}) - Q(\Sigma_{j < i} p_{\sigma(j)}). \qquad (5)$$

Definition (4) is equivalent to definition (3) in the sense that:

1) given $\mathbf{p}$ and $\mathbf{w}$ in (3), Q in (4) corresponds to W*, that is $f_{\text{wowa}}^{\mathbf{w},\mathbf{p}}(A) = \text{wowa}^{W^*,\mathbf{p}}(A)$
2) given $\mathbf{p}$ and Q in (4), there exist an interpolation function such that $W^* = Q$ if we define the weights $w_i = Q(i/n) - Q((i-1)/n)$.

A final remark is that there is a direct correspondence between definition (4) of WOWA operators and the extension of OWA operators in [22] to take into account importance weights for different information sources .

## 3 CONTINUOUS WOWA OPERATORS

In [17], WOWA operators have been extended in order to average continuous distributions. Based on the probabilistic interpretation of the normalized weighting vectors $\mathbf{p}$, (5) can be expressed as

$$\lambda_i = G(b_i) - G(b_{i-1}) \qquad (7)$$

where $b_i = a_{\sigma(i)}$ and $G(b)=Q(F(b))$, F being the corresponding accumulative probability distribution F of $\mathbf{p}$. Therefore, the extension to the continous case amounts to consider a continuous random variable defined by a probability density function f instead of $\mathbf{p}$. Assuming F and Q to be absolutely continuous (and thus G too), the extension of (4) for the continuous case is:

$$\int_{-\infty}^{+\infty} g(x)xdx$$

where $g(x) = q(F(x))\cdot f(x)$ and $q(x) = dQ/dx$. The function $q: [0, 1] \to \mathbb{R}$ is called *relative importance function* because it can be used to weight elements according to their accumulative probability, and thus it allows, for instance to weight small or large values independently on the values themselves. See [17] for further details.

**Definition.** Let $f: \mathbb{R} \to \mathbb{R}^+$ be a probability density function, F being its corresponding accumulative distribution function, $F(x) = \int_{-\infty}^{x} f(y)dy$, and let q be a relative importance function. The corresponding **Continuous WOWA operator** $\text{cwowa}^{q,f}$ is the mean value with respect to the probability density $g(x) = q(F(x))\cdot f(x)$, i.e.

$$\text{cwowa}^{q,f} = \int_{-\infty}^{+\infty} q(F(x))f(x)xdx \; . \qquad (6)$$

By analogy to the discrete case, when the relative importance function gives equal importance to any value, i.e. when $q(u) = 1$ for $u \in [0, 1]$ and $q(u) = 0$ otherwise, we get from (4), as a particular case, the expression for the **Continuous Weighted Mean** $\text{cwm}^f$ with respect to the density f, i.e. the expected value of the random variable distributed acoording to f:

$$\text{cwm}^f = \int_{-\infty}^{+\infty} f(x)xdx = E(f) \qquad (7)$$

On the other hand, if in (6) we now let f correspond to the uniform probability distribution on an interval [a, b], i.e. $f(x) = 1/(a-b)$ for $x \in [a, b]$ and $f(x) = 0$ otherwise, we get the following expression for the **Continuous Ordered Weighted Average** induced by q in the interval [a, b]:

$$\text{cowa}^q([a, b]) = \frac{1}{b-a}\int_a^b q\left(\frac{x-a}{b-a}\right)xdx \qquad (8)$$

These relationships are depicted in Figure 2. Notice that again, if we take both the uniform density function $f(x) = 1/(a-b)$ for $x \in [a, b]$ and $f(x) = 0$ otherwise, and the uniform quantifier $Q(x) = x$, we come up with the mean value of the interval [a, b].

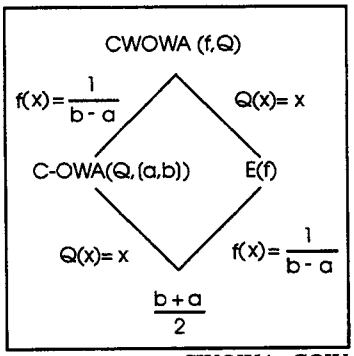



Figure 2. Relationships among CWOWA, COWA and CWM.

## 4 DEFUZZIFICATION WITH WOWA OPERATORS

Let A be a fuzzy set defined over a discrete domain $X = \{x_1, x_2, ..., x_n\}$ and let $\mu_A: X \rightarrow [0, 1]$ denote its membership function. Then if for each $x_i$ we define, by means of $\mu_A$, a weight $p_i \in [0, 1]$ such that $\sum p_i = 1$, the defuzzified value $x^*$ of A can be calculated as:

$$x^* = \Sigma_{i=1,n}\, p_i \cdot x_i$$

The definition of $p_i$ from the membership function $\mu_A$ corresponds to a selection of a defuzzification method. A simple, and very often used method for defuzzification, is the Center of Area (CoA). This corresponds to define

$$p_i = \frac{\mu_A(x_i)}{\sum_{j=1,n}\mu_A(x_j)}$$

When, instead of a discrete domain, a continuous one is considered the defuzzification can also be understood as an aggregation function: (1) is replaced in this case by the continuous weighted mean (definition (7)):

$$cwm^f = \int_{-\infty}^{+\infty} f(x)x\,dx$$

In this case, the $p_i$'s, that in the discrete case can be interpreted as probabilities, are replaced by a probability density function. Since $\mu_A(x)$ is a membership function (and thus, it is not required that its integral over the domain has to be 1 ) we need to normalise it. Therefore, the corresponding probability density is defined as:

$$f_A(x) = k\,\mu_A(x)\text{ , where } k = \left(\int_{-\infty}^{+\infty}\mu_A(x)dx\right)^{-1} \quad (9)$$

In this way, the continuous weighted mean can be applied. With this definition, the defuzzification method corresponds to the Center of Area Method

The approach of considering the defuzzification as an aggregation allows us to apply the continuous WOWA to this problem. To do this, we need to define the probability distribution and the fuzzy quantifier. The former is defined according to (9), the latter, in general can be any arbitrary quantifier. According to (9), we can rewrite the continuous WOWA, so it depends on the membership function $\mu_A$:

$$cwowa^{q,f_A} = k\int_{-\infty}^{+\infty} q\left(k\int_{-\infty}^{x}\mu_A(y)\,dy\right)\mu_A(x)\,x\,dx \quad (10)$$

This corresponds to a continiuous CoA method where each value is weighted according to each accumulated probability. In this way, the defuzzification can filter the values to consider. For example, if all probabilities are to be considered, then the relative importance function is defined as $q(x) = 1$ for all x in [0,1]. On the other hand, if extreme values that have a probability (below and above respectively) less than 0.1 are not to be considered, then we should take as relative importance function $q(x) = 1.25$ for all $x \in [0.1, 0.9]$ and $q(x)=0$ otherwise.

Note that in the definition (9) given above the expression

$$k^{-1} = \int_{-\infty}^{+\infty}\mu_A(x)dx$$

corresponds [26,9] to the cardinality Card(A) of the fuzzy set A. Analogously, we can define the cardinality of the subset $A_{\leq x} = A \cap [-\infty, x]$ of those elements of A which are smaller or equal than a certain value x as

$$Card(A_{\leq x}) =$$
$$= \int_{-\infty}^{+\infty}\min(\mu_A(y), \mu_{[-\infty, x]}(y))dy = \int_{-\infty}^{x}\mu_A(y)dy \quad (11)$$

This definition permits to see $Card(A_{\leq x} / A) = Card(A_{\leq x}) / Card(A)$ as the relative cardinality of the fuzzy subset $A_{\leq x}$ with respect to the fuzzy set A. In this way, the defuzzification method using the continuous WOWA operator and induced by a relative importance function q is defineda as:

$$DF_q^{wowa}(A) = cwowa^{q,f_A} =$$
$$= \frac{1}{Card(A)}\int_{-\infty}^{+\infty} q(Card(A_{\leq x} / A))\,\mu_A(x)\,x\,dx \quad (12)$$

This definition underlines the role of q as a filtering function defined over the relative cardinality of the fuzzy set $A_{\leq x}$. In particular, if we take $q(x) = 1$, for all $x \in [0, 1]$, we are not making any filtering and (12) yields the well known expression of the Center-of-Area method of defuzzification. The usefulness of the function q is made clear in the examples of next section.

## 5 EXAMPLE

Let us show how continuous WOWA operator can be used to defuzzify the fuzzy set A depicted in Figure 3 with the 3 different relative importance functions illustrated in Figure 4. We first consider the relative importance function $q_1$, a function that makes all the values equally important and, as already noticed in last section, (12) coincides with the CoA defuzzification method:

$$DF_{q_1}^{wowa}(A) \approx 2.70 \quad \text{(DF1)}$$

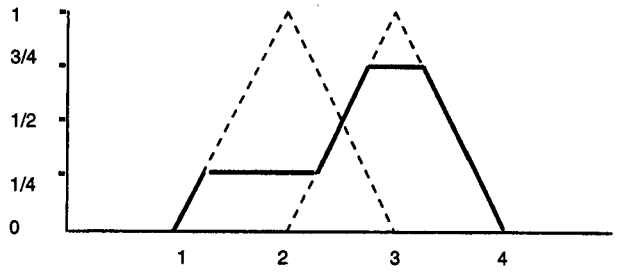


Figure 3. Membership function of fuzzy set A.

If we now defuzzify with the function $q_2$, we are discarding all those values outside the central interval with the 50% of the fuzzy cardinality of A. This would correspond to the case where we are not very confident in the extreme values and we prefer to concentrate on a central range by ignoring the 25% of the lowest values and the 25% of the highest values. In this case (12) yields the following value:

$$DF_{q_2}^{wowa}(A) \approx 2.82 \qquad \text{(DF2)}$$

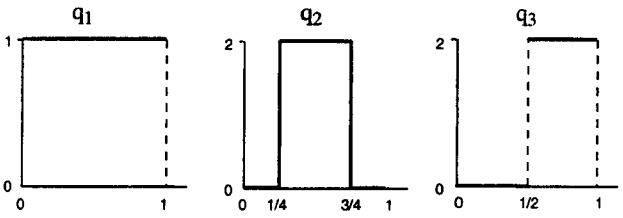


Figure 4. Three different relative importance functions.

Finally, if we consider the function $q_3$ only large values are considered to be important and we discard the 50% of the lowest values. This option would correspond to a very optimistic or very pessimistic, depending on the meaning of the variable being defuzzified, criterion. The defuzzied value is then:

$$DF_{q_3}^{wowa}(A) \approx 3.25 \qquad \text{(DF3)}$$

which is, as expected, sensibly higher than the previous ones. The three defuzzified values are shown in Figure 5.

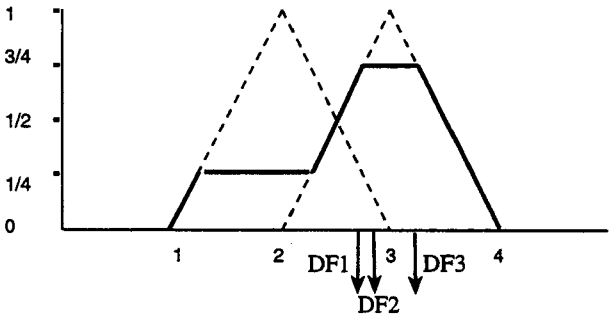


Figure 5. Defuzzified values using $q_1$, $q_2$ and $q_3$.

# 6 RELATED WORK AND CONCLUSIONS

A lot of defuzzification methods have been proposed in the literature. For instance in [5] six of the most well-known methods of defuzzification, Center of Area (CoA), Center of Sums (CoS), Center of Largest Area (CLA), First of Maxima (FoM), Midle of Maxima (MoM) and Height Method (HM), are described and analyzed. More recent references are for example [23, 22, 12, 18]. Our approach is close in spirit to Yager's approach [22], where he suggests the use of heuristic knowledge to discard or focus on some part of the fuzzy set to be defuzzified in order to overcome some typical problems (e.g. ambiguity) that use to appear in fuzzy control sytems. Yager's way to give more importance to some of the possible solutions than to another ones is not exactly comparable to ours. The weighting of values in [22] amounts to a kind of filtering of the membership function which yields another membership function over which a "classical" defuzzification method applies. Instead, our filtering is not directly over the membership function but on the function which at each point gives the accumultive relative cardinality.

In order to compare defuzzification methods, five criteria are considered in (5). Here only the first four criteria make sense:

- *Continuity*: a small change in the membership function should not result in a large change in the defuzzified value.
- *Disambiguity*: the defuzzification should never provide ambiguous results (even in the case of a symmetric bimodal membership function)
- *Plausibility*: the defuzzified value of a convex membership function should have a high degree of membership value.
- *Computational complexity*: the less computationaly complex method, the better.

The fifth criterion (weight counting) takes into account the matching degree of the input with the membership functions of the antecedents in the fuzzy control rules. This criterion does not apply in our setting, since we are only taking into account the output membership functions and not how they have been obtained. Defuzzifying with continuous WOWA operators fit for sure the *continuity* and *disambiguity* criteria, the computational complexity may vary on the chosen function q (but in any case not very low), and the plausibility criterion is not generally satisfied like the CoA method.

As for future work, we envisage an immediate extension of the method proposed in this paper by taking into account the methodolgy proposed in [23]. They consider a defuzzification process composed of two steps: applying a BADD transformation to the membership function to generate a probability distribution and then to compute the mean value of this distribution. The BADD transformation plays a role of a weighting of values according to their plausibility. Therefore it seems rather natural to extend our method by first applying a BADD transformation to take into account the plausibility importance of values and then applying a WOWA aggregation to take into account the accumulative relative cardinalities.

**Acknowledgements**

Authors are partially supported by the European Community, under the contract VIM: CHRX-CT93-0401 and by the CICYT project SMASH: TIC-96-1038-C04-01 and TIC-96-1038-C04-04.

**References**

[1] Aczél, J., On Weighted synthesis of judgements, Aequationes Math., 27 (1984) 288-307.

[2] Alsina, C., Mayor, G., Tomás, M.S., Torrens, J., A characterization of a class of aggregation functions, *Fuzzy Sets and Systems*, 53 (1993) 33-38.

[3] Amat, J., Esteva, F., López de Mántaras, R., Autonomous Navigation Troup for Cooperative Modelling of Unkown Environments, *Proceedings of the Int. Conference on Advanced Robotics* (ICAR'95), Sant Feliu de Guixols, Catalunya, Spain, (1995) 383-389.

[4] Dubois, D., Koning, J-L., Social choice axioms for fuzzy set aggregation, *Fuzzy sets and systems*, 43 (1991) 257-274.

[5] Driankov D., Hellendoorn H., Reifrank M. *An introduction to Fuzzy Control.* Springer Verlag (1993).

[6] Fodor, J., Marichal, J.-L., Roubens, M., Characterization of the Ordered Weighted Averaging Operators, *IEEE Trans. on Fuzzy Systems*, 3:2 (1995) 236-240.

[7] Genest, C., Zidek, J.V., Combining Probability Distributions: A Critique and An Annotated Bibliography, *Statistical Science*, 1:1 (1986) 114-148.

[8] Grabisch, M., Fuzzy integral in multicriteria decision making, *Fuzzy Sets and Systems*, 69 (1995) 279-298.

[9] Klir G.J., Yuan B.: Fuzzy Sets and Fuzzy Logic, Prentice Hall PTR (1995).

[10] López, B., Álvarez, S., Millán, P., Puig, D., Riaño, D., Torra, V., Multistage vision system for road lane markings and obstacle detection, Proceedings of *Euriscon '94*, Malaga (1994) 489-497.

[11] Nishiwaki, Y., Preyssl, C., Schmid, S., Decision Making under uncertainty: Application of fuzzy logic to risk assessment abd nabagement of space systems, *Proceedeing of the third International Conference on Fuzzy Logic, Neural Networks and Soft Computing*, Iizuka, Japan, (1994) 263-264.

[12] Rondeau L., Ruedas R., Levrat L., Lamotte L. A defuzzification method respecting the fuzzification. *Fuzzy sets and systems*, 86 (1997) 311-320.

[13] Sandri, S.A., Dubois, D., Kalfsbeek, H., Elicitation, Assesment, and Pooling of Expert Judgements Using Possibility Theory, *IEEE Trans. on Fuzzy Systems*, 3:3 (1995) 313-335.

[14] Slany, W., Scheduling as a fuzzy multiple criteria optimization problem, *Fuzzy Sets and Systems*, 78 (1996) 197-222.

[15] Torra V.: Weighted OWA operators for synthesis of information, Proceedings of the Fifth IEEE International Conference on Fuzzy Systems (IEEE-FUZZ'96), New Orleans, USA, 1996, pp. 966-971.

[16] Torra, V., The Weighted OWA operator, *Int. J. of Intel. Systems*, 12 (1997) 153-166.

[17] Torra V., Godo L. : Averaging continuous distributions with the WOWA operator. Proc. of the second European Workshop on Fuzzy Decision Analysis and Neural Networks (EFDAN'97), Dortmund, Germany, 1997, pp. 10-19.

[18] Valente de Oliveira J. A set-theoretical defuzzification method. *Fuzzy sets and systems*, 76 (1995) 63-71.

[19] Yager, R.R., On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Trans. on SMC*, 18 (1988) 183-190.

[20] Yager, R.R., Families of OWA operators, *Fuzzy Sets and Systems*, 59 (1993) 125-148.

[21] Yager, R.R., Quantifier Guided Aggregation Using OWA operators, *Int. J. of Intel. Systems*, 11 (1996) 49-73.

[22] Yager R.R. Knowledge-based defuzzification. *Fuzzy sets and systems*, 80 (1996) 177-185

[23] Yager, R.R., Filev, D.P. On the issue of defuzzification and selection based on a fuzzy set. *Fuzzy sets and systems*, 55 (1993) 255-271.

[24] Yager, R.R., Filev, D.P., *Essentials of fuzzy modeling and control*, John Wiley (1994).

[25] Yager, R.R., Goldstein, L.S., Mendels, EFUZMAR: An approach to aggregating market research data based on fuzzy reasoning, *Fuzzy sets and systems*, 68 (1994) 1-11.

[26] Zadeh L.A.: PRUF — A meaning representation language for natural languages. *Int. J. of Man-Machine Studies*, 10, (1978) 395-460.

# CONTROL BORROSO ADAPTATIVO DE UN CONVERTIDOR DC/DC CONMUTADO

**Pablo J. Carbonell Cortés**
Escola Politécnica Superior d'Alcoi
Dpto. DISCA, *Universitat Politècnica de València*
P. Viaducte, 1, 03801 ALCOI
pjcarbon@aii.upv.es

## Resumen

El presente trabajo propone un regulador borroso de parámetros adaptables como técnica de control de un convertidor dc/dc conmutado. Se trata de sistemas fuertemente no lineales y que se hallan sometidos a importantes variaciones respecto a las condiciones nominales por el hecho de una modificación en la carga o en la tensión de entrada. Como solución, se mostrará un algoritmo borroso basado en información lingüística, que admitirá un esquema simple de adaptación con el fin de sintonizar sus parámetros. En la propuesta aquí presentada se primará la sencillez de implementación de dicho regulador. Este control será evaluado por medio de diversas simulaciones, en las que se analizará su comportamiento en cuanto a regulación en la carga y en la línea.

**Palabras Clave**: Control neuroborroso, fuentes conmutadas, sistemas borrosos adaptativos.

## 1 INTRODUCCIÓN

Dentro del campo de la electrónica de potencia, una de los puntos de mayor importancia es la elección entre las posibles alternativas del esquema de control de dichos dispositivos. Esto es así porque se suele tratar de sistemas no lineales, con comportamientos complejos. Por ello, la solución clásica de adopción de técnicas de control lineales analógicas, generalmente diseñadas en el dominio de la frecuencia, se ha mostrado en muchos casos ineficaz y susceptible de mejoras.

Entre las diversas alternativas posibles, las técnicas de control inteligente se vislumbran como una opción interesante y que puede competir perfectamente frente a otras tecnologías [1]. En el presente trabajo, se presenta una aplicación de electrónica de potencia donde resulta adecuado emplear tecnología *fuzzy*. Se trata de controlar una fuente conmutada a través de un regulador borroso. Frente a trabajos anteriores [4], cabe destacar como principal aportación la adición a dicho regulador de un algoritmo de adaptación que permite ajustar las reglas a las condiciones de trabajo cambiantes.

## 2 CONVERTIDOR DC/DC DE TIPO *BUCK*

Una fuente conmutada es un dispositivo electrónico que se va a emplear como fuente de alimentación de corriente continua a diferentes cargas. Por tanto, las condiciones exigibles a este sistema son el suministro de una tensión continua estable frente a variaciones en la carga o en la línea. Las fuentes conmutadas se emplean cuando las cargas que se van a alimentar son de potencias medias o altas, de manera que una fuente clásica de tipo lineal resulta de difícil aplicación, ya que se precisaría de un dispositivo excesivamente pesado e incluso costoso. Frente a esta fuente, surgen las conmutadas, que tienen como principal característica su funcionamiento en modo conmutado. Es decir, existe un elemento semiconductor que actúa como interruptor y conmuta a una frecuencia fija $T_s$. La elección en todo momento del *ciclo de trabajo* $d$ se emplea para realizar el control sobre la tensión de salida, al decidir el tiempo que la tensión de línea no regulada transfiere energía hacia el convertidor.

$$d = \frac{T_{on}}{T_s} \tag{1}$$

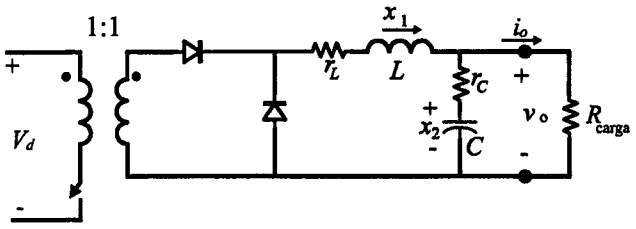



Figura 1: Estructura básica de un convertidor *buck*.

De las diversas topologías existentes [3], se centrará este estudio en el convertidor denominado de tipo *buck*. En la figura 1 puede verse la estructura básica de un convertidor de estas características. Durante el tiempo del cierre del conmutador, la tensión de entrada, procedente de un puente rectificador se encarga de cargar el condensador, mientras que durante el intervalo de apagado éste se descarga a través de la carga.

La principal dificultad que presenta controlar un sistema de estas características es el hecho de que la carga puede ser variable. Al mismo tiempo, la tensión de entrada puede estar sometida igualmente a fluctuaciones. Los modelos empleados habitualmente para analizar estos sistemas se basan en que éste va evolucionando a través de varias topologías lineales.

$$\left.\begin{aligned} \dot{x} &= A_i x + B_i v_e \\ v_o &= C_i x \end{aligned}\right\} \tag{2}$$

Las matrices de estados $A_i$, $B_i$ y $C_i$ representan cada una de estas topologías. Pueden ser variables con el tiempo, ya que dependen de la carga. Las variables de estado vienen determinadas por los elementos almacenadores de energía presentes en el sistema (bobinas y condensadores).

En general, en un dispositivo conmutado pueden distinguirse tres estados:

a) Durante el tiempo de conducción, en el que la tensión de entrada carga el condensador.
b) Durante el apagado, en el que se descarga la bobina.
c) Si la bobina llega a descargarse totalmente, el diodo entra en polarización inversa, dando lugar a una nueva topología. Es lo que se denomina *modo de conducción discontinua*.

# 3 ESQUEMA DEL CONTROL BORROSO EMPLEADO

Se pretende obtener un regulador borroso que calcule la variación en el ciclo de trabajo más adecuada para mantener la tensión de salida del convertidor estable en el valor deseado. Este controlador presentará los cuatro módulos típicos: *borrosificador*, base de reglas, motor de inferencia y *desborrosificador*, más un módulo de adaptación.

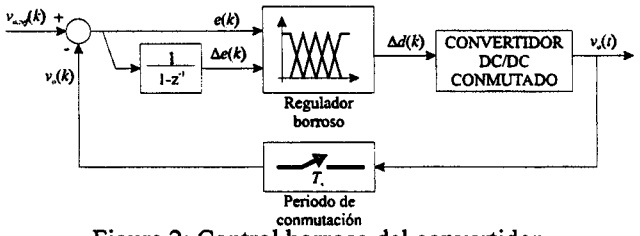



Figura 2: Control borroso del convertidor.

En la figura 2 puede verse la estructura del sistema de control empleado. Como entrada se tomará el error $e(k)$ y la variación instantánea en el error $\Delta e(k)$.

$$e(k) = v_{o,ref}(k) - v_o(k) \tag{3}$$

$$\Delta e(k) = e(k) - e(k-1) \tag{4}$$

Por su parte, la salida del regulador se tomará como la variación en el ciclo de trabajo $\Delta d(k)$ según la siguiente relación:

$$d(k) = d(k-1) + \eta \Delta d(k) \tag{5}$$

donde $\eta$ es la ganancia de salida del regulador y actúa como constante de tiempo sobre el sistema.

El espacio de entrada se divide en una primera aproximación en cinco conjuntos borrosos que corresponden a: NB (negativo grande), NS (negativo pequeño), ZE (cero), PS (positivo pequeño) y PB (positivo grande). Con la intención de disponer de un regulador que sea fácilmente implementable, las funciones de pertenencia se van a tomar de tipo triangular, tal y como se muestra en la figura 3.

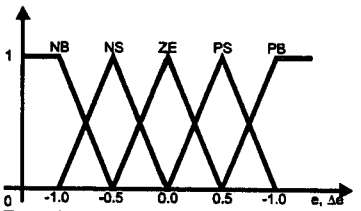


Figura 3: Funciones de pertenencia de los conjuntos borrosos de entrada.

A efectos de normalizar las variables, se definen dos factores de escala $\beta$ y $\beta_e$ que dividen cada una de las entradas y que deberán ser ajustados de forma experimental.

Las reglas se tomarán todas del tipo:

$$R_i\text{: IF } e(k) \text{ is } \Omega_{1i} \text{ and } \Delta e(k) \text{ is } \Omega_{2i}\text{, THEN } \Delta d(k) \text{ is } \Phi_i \qquad (6)$$

donde $\Omega_{ji}$ son los conjuntos de entrada y $\Phi_i$ son los conjuntos de salida. Las funciones de pertenencia de salida se considerarán de tipo *singleton*, resultando un sistema borroso asimilable a los de tipo *Sugeno*. Esta base de reglas estará compuesta por todas las posibles combinaciones entre los conjuntos de entrada.

Con el fin de lograr un regulador de gran simplicidad computacional, para la operación de *conjunción* se ha elegido la función *mínimo*, mientras que para la inferencia se elige el producto. Para *desborrosificar* la variable de salida se elige la técnica del centro de gravedad [6]. Esto lleva a que la salida sea calculada a partir de:

$$z_i = min\left\{\mu_{\Omega_{1i}}(e(k)), \mu_{\Omega_{2i}}(\Delta e(k))\right\} \qquad (7)$$

donde $\mu_{\Omega_{1i}}(e(k)), \mu_{\Omega_{2i}}(\Delta e(k))$ son las funciones de pertenencia mostradas en la figura 3 para las dos entradas. Obteniéndose como valor de salida de la base de reglas mostrada en (6):

$$\Delta d(k) = \frac{\sum_{i=1}^{N} z_i \Phi_i}{\sum_{i=1}^{N} z_i} \qquad (8)$$

Las reglas deberán partir del conocimiento *a priori* que se posee del convertidor. Estas reglas, tal y como propone [4], son heurísticas y muy simples. Se resumen en:

1. Cuando la salida del convertidor está lejos del punto de trabajo, el cambio en el ciclo de trabajo debe ser grande.
2. Si, por el contrario, la salida se está aproximando al punto de trabajo, la variación en el ciclo de trabajo deberá ser pequeña.
3. Si la salida se aproxima hacia el punto de trabajo de forma rápida, el ciclo de trabajo se debe mantener constante para evitar sobreoscilaciones.
4. Si se ha alcanzado el punto de trabajo, pero la salida continúa variando, se debe modificar ligeramente el ciclo de trabajo para impedir que la salida se aleje.
5. Si se alcanza el punto de consigna y la salida se halla prácticamente estacionaria, el ciclo de trabajo se deberá mantener constante.
6. Cuando la salida está por encima del punto de consigna, el cambio en el ciclo de trabajo debe ser negativo y viceversa.

Como puede verse, el comportamiento en el convertidor se supone *a priori* simétrico. Es decir, el modo de actuar del mismo es igual y en sentido contrario, para una misma variación en la entrada, pero de signos opuestos. Adaptando la propuesta del trabajo [4] a la estrutura utilizada en la figura 3, se obtiene una tabla de actualización o *lookup table* como la mostrada en la Tabla 1.

Tabla 1: Tabla de reglas del controlador borroso.

| | | Δe | | | | |
|---|---|---|---|---|---|---|
| | | NB | NS | ZE | PS | PB |
| e | PB | 0.30 | 0.35 | 0.45 | 0.65 | 1.00 |
| | PS | 0.00 | 0.10 | 0.20 | 0.35 | 0.50 |
| | ZE | -0.20 | -0.10 | 0.00 | 0.10 | 0.20 |
| | NS | -0.50 | -0.35 | -0.20 | -0.10 | 0.00 |
| | NB | -1.00 | -0.65 | -0.45 | -0.35 | -0.30 |

Como parámetros nominales de referencia, se considera un convertidor *buck* con los valores mostrados en la Tabla 2.

Tabla 2: Parámetros del convertidor *buck*.

| PARÁMETRO | VALOR |
|---|---|
| Inductancia L | 200 μH |
| Capacidad de salida C | 50 μF |
| Resistencia serie de la bobina $r_L$ | 0.1 Ω |
| Resistencia serie del condensador $r_C$ | 0.1 Ω |
| Resistencia de carga R | 5 Ω |
| Voltaje de salida V | 15 V |
| Voltaje de salida de referencia $V_{ref}$ | 5 V |
| Frecuencia de conmutación $f_s$ | 100 kHz |

Por su parte, los factores de escala se han fijado en las constantes reflejadas en la Tabla 3.

Tabla 3: Factores de escala.

| FACTOR | VALOR |
|---|---|
| $\eta$ | 0.012 |
| $\beta$ | 3 V |
| $\beta_e$ | 0.01 V |

# 4 ESQUEMA DE ADAPTACIÓN

El regulador propuesto en el epígrafe anterior presenta como principal inconveniente su dependencia excesiva del conocimiento *a priori* del funcionamiento del sistema para su ajuste. Con el fin de aumentar el rango de condiciones de trabajo donde es aplicable éste, se va a considerar una ley de adaptación para los centros de los conjuntos de salida $\Phi_i$.

En primer lugar, se fijará el índice de coste que se desea minimizar:

$$J = E\left[e^2(k)\right] \tag{9}$$

Como estimador instantáneo de dicho índice, se toma el error en dicho instante:

$$J(k) = e^2(k) \tag{10}$$

Los parámetros que se desea ajustar son los centros de las salidas $\Phi_i$. Esto presenta la ventaja de que el sistema es lineal en sus parámetros ajustables [6]. Para aplicar algún algoritmo de aprendizaje, se debe estimar la dependencia de dichos parámetros con respecto al índice de coste:

$$\left.\frac{\partial J}{\partial \Phi_i}\right|_k = \frac{\partial J}{\partial e}\frac{\partial e}{\partial v_o}\frac{\partial v_o}{\partial d}\frac{\partial d}{\partial \Delta d}\left.\frac{\partial \Delta d}{\partial \Phi_i}\right|_k \tag{11}$$

Para obtener dicha dependencia se empleará la regla de la cadena, calculando cada una de las dependencias parciales mostradas en la relación anterior.

Cada uno de estos términos puede ser calculado por separado. Según (10), se tendrá que:

$$\left.\frac{\partial J}{\partial e}\right|_k = 2e(k) \tag{12}$$

Del mismo modo, a partir de la expresión (3),

$$\left.\frac{\partial e}{\partial v_o}\right|_k = -1 \tag{13}$$

Para calcular la variación de la tensión de salida respecto al ciclo de trabajo, se debe considerar la relación, válida para conducción continua, obtenida a partir de la resolución de las ecuaciones de estados, suponiendo constantes los parámetros durante el tiempo de conmutación.

En primer lugar, durante el tiempo de cierre del conmutador $0<t<dT$, el sistema viene regido por la siguiente ecuación de estado:

$$\dot{x} = A_1(t)x + B_1(t)v_e(t) \tag{14}$$

Las matrices $A_1$ y $B_1$ pueden variar a lo largo del tiempo en función de la carga. Del mismo modo, la tensión de entrada $v_e(t)$ puede sufrir fluctuaciones en torno a su valor nominal. Eligiendo un tiempo de conmutación menor que todas las constantes de tiempo involucradas en el sistema, puede considerarse que durante el tiempo de conducción, las matrices de la ecuación de estado se mantienen prácticamente constantes.

De esta forma, es posible conocer la evolución que sufre el estado del sistema durante este intervalo [5] a partir de (13):

$$x(dT) = e^{A_1 dT}x_0 + e^{A_1 dT}\int_0^{dT} e^{-A_1\xi}B_1 v_e(t)d\xi \tag{15}$$

Por otra parte, durante el tiempo de apagado del conmutador, considerando conducción continua, se tendrá la ecuación de estado:

$$\dot{x} = A_2(t)x \tag{16}$$

Considerando de nuevo la matriz de estado como básicamente constante durante este semiperiodo $t=(1-d)T$, se tiene a partir de (16):

$$x(T) = e^{AT}x(dT) \tag{17}$$

En un convertidor *buck*, como el considerado en la figura 1, las matrices de estado $A_1=A_2=A$, ya que durante la conmutación no se produce un cambio en la topología del circuito que afecte a las relaciones entre variables de estado. Esto va a simplificar considerablemente los cálculos, de manera que (15) y (17) pueden ser expresados como:

$$x(T) = e^{AT}x_0 + e^{AT}\int_0^{dT} e^{-A\xi}B_1 v_e(t)d\xi \tag{18}$$

Si en esta última expresión se supone que la tensión de entrada es en promedio constante y de valor la tensión rectificada $v_e(t)= V_d$, se llega a esta nueva relación:

$$v_o(T) = C_2 x(T) = C_2 e^{AT}[x_0 + V_d A^{-1}(1 - e^{-AdT})B_1] \quad (19)$$

A partir de (19), puede ser calculada la dependencia de la tensión de salida con respecto al ciclo de trabajo:

$$\left.\frac{\partial v_o}{\partial d}\right|_k = V_d T C_2 e^{AT} e^{-Ad(k-1)T} B_1 \quad (20)$$

Es éste un valor que depende, por tanto, de la topología, de los parámetros de diseño y de las condiciones de funcionamiento. Para analizar la estabilidad de este sistema es importante considerar la influencia que pueden tener los parámetros de la carga sobre dicha variación.

A continuación se comparará el valor de esta variación a partir de los parámetros nominales frente a posibles desviaciones en la carga para el convertidor *buck* considerado.

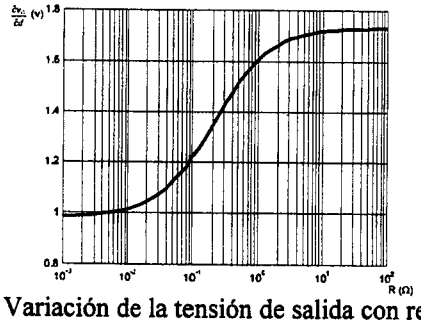



Figura 4: Variación de la tensión de salida con respecto al ciclo de trabajo para diferentes valores de la resistencia de carga.

Como puede observarse en la figura 4, esta variación sólo se separa de la nominal para valores muy bajos de la carga, no superando en ningún caso el 45%. Para el algoritmo de adaptación, se considerará constante y de valor nominal.

$$\left.\frac{\partial v_o}{\partial d}\right|_k \cong K_0 = \left(V_d T C_2 e^{AT} e^{-Ad(k-1)T} B_1\right)_{nominal} \quad (21)$$

En cuanto al resto de las variaciones, a partir de la expresión (5) se llega a:

$$\left.\frac{\partial d}{\partial \Delta d}\right|_k = \eta \quad (22)$$

Por último, a partir de la relación (8), se obtiene:

$$\left.\frac{\partial \Delta d}{\partial \Phi_i}\right|_k = \frac{z_i}{\sum z_i} \quad (23)$$

El algoritmo de aprendizaje que se va a emplear es el del *gradiente descendiente* [2], siendo $\alpha$ el factor de aprendizaje. De manera que combinando las ecuaciones (12), (13), (21), (22) y (23) se llega a:

$$\Delta C_i(k) = -\alpha \frac{\partial J}{\partial C_i} = \alpha \cdot 2e(k) \cdot K_0 \cdot \eta \cdot \frac{z_i}{\sum z_i} \quad (24)$$

# 5 SIMULACIONES

Con el fin de verificar el funcionamiento adecuado del regulador borroso adaptativo propuesto, se han realizado diversas simulaciones. Para ello se han empleado un conjunto de modelos desarrollados en *matlab/simulink*.

En primer lugar, se desea verificar la inmunidad de este sistema frente a variaciones en la carga. Para ello se variará la carga desde la nominal de 5 Ω a otra de 3 Ω. En la figura 5, como puede verse, el sistema es capaz de devolver la salida a la tensión nominal de 5 Voltios. En la figura 6 se ha representado la evolución en los centros de los conjuntos de salida de cada una de las reglas según la ley de adaptación (24). El factor de aprendizaje $\alpha$ se ha tomado unitario.

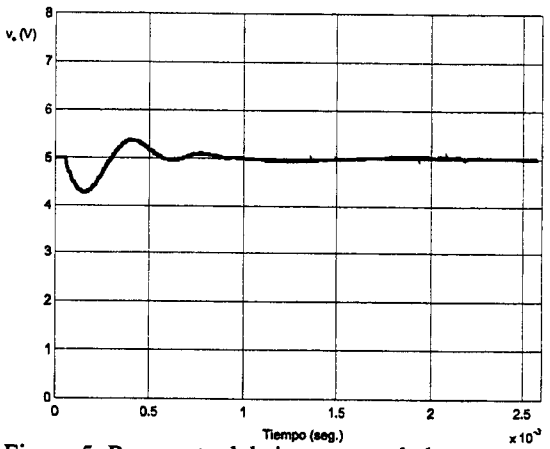



Figura 5: Respuesta del sistema regulado ante una variación en la carga de 5 Ω a otra de 3 Ω.

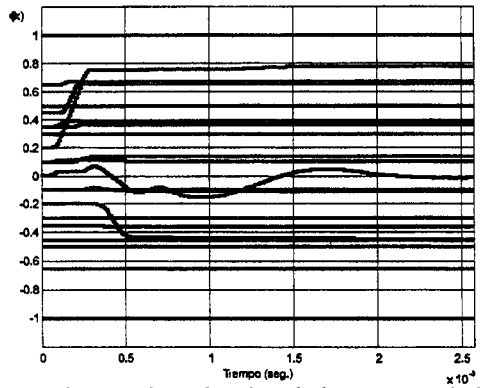


Figura 6: Autosintonización de los centros de los conjuntos de salida del regulador.

Del mismo modo, en las figura 7 puede observarse la evolución en la tensión de salida del sistema cuando la variación se produce en la tensión de alimentación rectificada de entrada que pasa de 15 V. nominales a 20V. En la figura 8 se ha representado las trayectorias que realizan los diferentes parámetros sintonizables con el fin de adaptarse a las nuevas condiciones de trabajo.

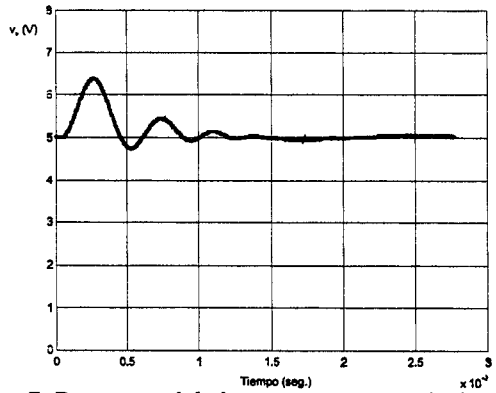


Figura 7: Respuesta del sistema ante una variación en la entrada de 15 V. a 20V.

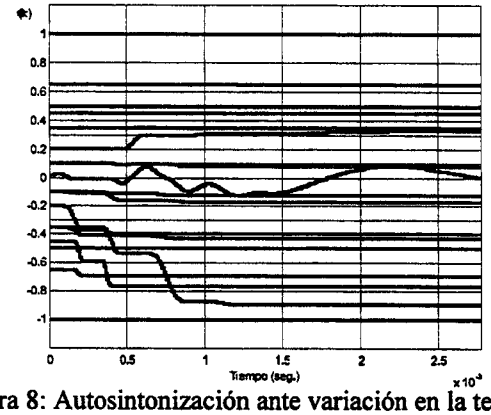


Figura 8: Autosintonización ante variación en la tensión de entrada

# 6 CONCLUSIONES

Cada vez se exigen mayores prestaciones a una fuente conmutada, como consecuencia de que se está ampliando el rango de aplicaciones donde resulta conveniente emplear éstas. Es por ello, que se precisa de técnicas de regulación cada vez más refinadas, capaces de adaptarse a condiciones de entorno cambiantes.

Los resultados obtenidos en este estudio muestran la conveniencia de la aplicación del control borroso en el campo de los convertidores dc/dc, como alternativa interesante a otras técnicas clásicas. La adición de un esquema de adaptación, hace éste aún más atractivo, ya que amplía el rango de condiciones en el que el sistema responde de forma adecuada. En comparación a los resultados obtenidos en [4], se observa una atenuación mayor de las oscilaciones, así como una velocidad mayor de establecimiento. El regulador aquí presentado se muestra como una alternativa relativamente económica y de fácil implementación.

A partir de la solución propuesta, estudios futuros deberán caminar hacia la implementación de reguladores borrosos donde todos sus parámetros presenten adaptación. Es decir, se deberá permitir el autoajuste de los parámetros de las funciones de pertenencia de los conjuntos borrosos de entrada, adicionalmente a los conjuntos de salida presentados en este trabajo. Esto, si bien hará más potente el esquema de control, éste presentará por el contrario un algoritmo de aprendizaje más complejo y con mayor carga computacional.

## Referencias

[1] Bose, B.K. "Expert system, fuzzy logic, and neural network applications in Power Electronics and motion control", *Proc. IEEE*, vol. 82, no. 8, pp. 1303-1323, 1994.
[2] Brown, M., Harris, C. *Neurofuzzy Adaptive Modelling and* Control, Englewood Cliffs, NJ: Prentice Hall, 1994.
[3] Mohan, N., Undeland, T.M., Robbins, W.P. *Power Electronics. Converters, applications and design.* New York: John Wiley, 1995.
[4] So, W.C. "Development of a fuzzy logic controller for DC/DC converters: design, computer simulation, and experimental evaluation", *IEEE Trans. on Power Electronics*, vol. 11, no. 1, pp. 24-31, 1996.
[5] Tse, C.K., Adams, K.M. "Quasi-Linear Modeling and Control of DC-DC Converters", *IEEE Trans. On Power Electronics*, vol. 7, no. 2, pp. 315-323, 1992.
[6] Wang, L.X. *A course in fuzzy systems and control*, Englewood Cliffs, NJ: Prentice Hall, 1997.

# DETECCIÓN DE FALLOS EN CAJAS DE ENGRANAJES MEDIANTE BASES DE REGLAS DIFUSAS

**Ferreiro García, Ramón; Pardo Martínez, Xoán C.; Vidal Paz, José; Carneiro Diaz, Victor**
Departamento de Electrónica e Sistemas. Universidade da Coruña.
E.S. Marina Civil. Paseo de Ronda 51, 15011. A Coruña.
E-mail: {ferreiro, pardo, vidal, victor}@des.fi.udc.es

## Resumen

En este artículo se describe el estudio y aplicación de un procedimiento para la diagnosis de sistemas mecánicos mediante la aplicación de una aproximación basada en el reconocimiento de patrones, a fin de validar su capacidad para resolver el problema de la diagnosis de un tipo específico de sistemas tecnológicos. El tipo de sistemas objetivo de este trabajo es el de los mecanismos rotativos acoplados mediante cajas de engranajes. El método de diagnóstico utiliza un conjunto de bases de reglas, deterministas y difusas, para la detección de fallos en los engranajes, aplicando un conjunto de patrones de fallo conocidos a priori, que han de ser reconocidos en los datos contenidos en un espectro, resultado de aplicar un algoritmo FFT a la información obtenida con un equipo de medición de vibraciones adecuado.

**Palabras Clave:** Diagnosis Automatizada de Fallos, Lógica Difusa, Reconocimiento de Patrones.

## 1 INTRODUCCIÓN

El diagnóstico de fallos mecánicos asistido por ordenador está siendo una de las técnicas más populares utilizada en el mantenimiento global de equipos mecánicos. En este trabajo se propone un método para diagnosticar el funcionamiento de los engranajes de una máquina rotatoria analizando el espectro obtenido a partir de los datos capturados mediante un equipo de monitorización de vibraciones en la ubicación apropiada del eje de rotación [2].

Cortina [4] y Collacot [3] entre otros han examinado la posibilidad de aplicar las técnicas de clasificación de señal del procesador de Bayes al diseño de un sistema automático de detección de fallos optimizado estadísticamente, usando datos aprendidos como resultado de la realización de pruebas. La técnica de decisión del procesador de Bayes está basada en el procedimiento estadístico Bayesiano [4], y se ha aplicado en diagnosis de fallos durante mucho tiempo.

La arquitectura general de un procedimiento de diagnosis basado en el reconocimiento de patrones está compuesto de las etapas siguientes [7]:

a) Obtención de los datos pertenecientes a las mediciones mediante un conjunto de sensores.

b) Procesamiento de los datos, que se compone de los pasos siguientes: parametrización de los datos, formación de las clases de patrones, selección de características y diseño del clasificador.

c) Diagnóstico 'on-line', en la que el clasificador asigna cada nueva observación a una de las clases predefinidas o la rechaza para su posterior análisis.

Este proceso puede resumirse de la siguiente manera: dado un patrón de fallo, la estrategia de resolución del problema consiste en aplicar un método estructurado para determinar si tal patrón está incluido en la información medida. La selección de la estrategia propuesta depende de cierto conocimiento a priori, que es esencial para la deducción de los consecuentes. El conocimento necesario está relacionado con los conceptos siguientes: tipo de máquina, características estáticas y dinámicas, tipo de sensor, localización y posición, etc. La diagnosis por medio de procedimientos basados en reglas difusas puede usarse satisfactoriamente como alternativa a los métodos estadísticos clásicos [6].

La secuencia a procesar está formada por las tareas siguientes:

a) Obtención de la respuesta temporal de la máquina sometida a una excitación para encontrar el espectro de frecuencias de excitación por medio del calculo de la FFT.

b) Diagnosis y toma de decisiones basada en el análisis del espectro obtenido.

Este artículo se centra en esta última tarea: cada tipo de fallo tiene un espectro individual y particular. Entonces, buscar una fallo particular implica reconocer y discriminar únicamente su espectro. La técnica consiste en determinar si el espectro del fallo particular está incluido en el espectro general. De esta manera, por comparación del espectro del patrón de fallo con el general, y excluyendo las frecuencias no relevantes, se decide si el fallo existe.

Para encontrar la información necesaria sobre un fallo se utiliza un espectro patrón para un fallo particular. Por medio de estimaciones empíricas se obtienen las funciones de pertenencia para la base de reglas difusa. El universo de discurso se obtiene a partir del espectro patrón. Esto implica que para cada fallo debe existir un espectro patrón conocido a priori. La difusividad de la información obtenida mediante medición es mostrada en la figura 1 (a), en la que puede observarse la apariencia general de un espectro obtenido por aplicación de la FFT.

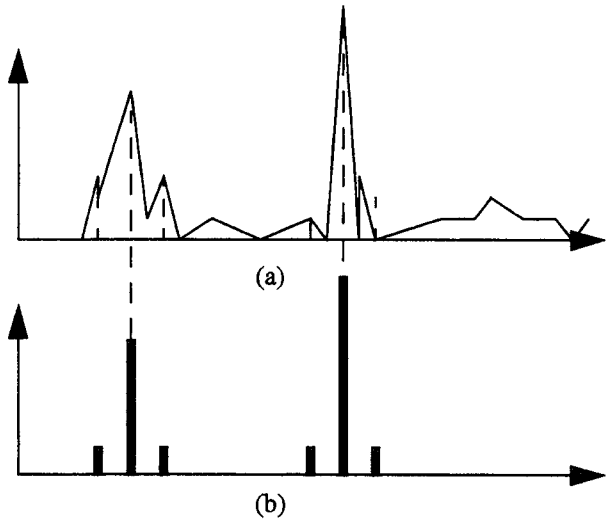



Figura 1: (a) Espectro de un engranaje que presenta fallos en su funcionamiento. (b) Espectro del patrón del fallo.

Resulta interesante el hecho de aplicar una técnica de diagnóstico similar a la que el ser humano usa en los procesos con incertidumbre. Con referencia a la figura 1, resulta que en (a) se observan dos picos con una distancia entre ellos igual a un múltiplo de la frecuencia fundamental de funcionamiento del mecanismo bajo diagnóstico. De la misma manera, se observan sendos picos de menor magnitud a ambos lados de los picos principales, y dispuestos simétricamente respecto a ellos. Los valores máximos de todos estos picos se muestran en la figura (b). El patrón mostrado en esta figura es el de un fallo típico de los engranajes, y por lo tanto, constituye un patrón de fallo.

La idea fundamental en nuestro estudio reside en la separación de las componentes difusas para deducir el tipo de fallo en base a su patrón. Por tanto, a partir de ciertos conocimientos a priori como el tipo de equipo, sus características de operación y los patrones de fallo más relevantes, es posible comenzar una tarea de discriminación de información para determinar si alguno de los patrones de fallo está incluido en el espectro asociado a la información capturada.

## 2 PROCEDIMIENTO GENERAL

El proceso general para realizar una tarea de diagnóstico basada en una estrategia de reconocimiento de patrones, en la que hay involucrada información afectada de incertidumbre, está basado en una secuencia de subtareas que se relacionan a continuación, y que se describirán posteriormente:

a) Medición de las vibraciones.

b) Calculo de la FFT.

c) Clasificación de las mediciones mediante lógica booleana de acuerdo a los patrones de fallos conocidos a priori.

d) Calculo del conjunto de funciones de pertenencia.

e) Desborrosificación de la base de reglas difusa.

f) Correción de fallos de acuerdo a los resultados de la desborrosificación.

Esta metodología es clara y bien estructurada para aquellas máquinas cuyos patrones de fallos son conocidos y las frecuencias que definen los fallos están claramente separadas de aquellas que pertenecen al funcionamiento normal de la máquina [5].

## 3 APLICACIÓN

Como aplicación práctica se propone en este artículo el diagnóstico de una máquina rotatoria industrial. Consideremos una máquina de rotación compuesta por un motor, una caja de engranajes y una sierra Diamond, cuyas

características son las siguientes:

Potencia del motor = 90 kW.

rpm motor = 1496, (f1 = 24.78 Hz).

rpm sierra = 272, (f2 = 4.51 Hz).

Número dientes rueda 1 (t1) = 8.

Número dientes rueda 2 (t2) = 44.

Frecuencia del dentado (ft =t1*f1=t2*f2)=198.25 Hz.

Localización de la medida = En la carcasa de la caja de engranajes, en posición axial al eje de la sierra Diamond

La figura 2 muestra la configuración del acoplamiento mediante la caja de engranajes, así como la localización del punto de medida en la carcasa de la misma, con el fin de detectar la familia de fallos cuyo patrón pertenece al de los fallos en el contacto entre los dientes del engranaje.

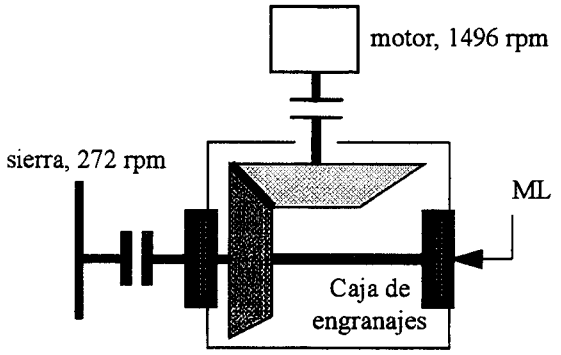



Figura 2: Acoplamiento entre el motor y la sierra Diamond mediante un tren de engranajes.

La figura 3 (incluida al final del artículo) muestra el espectro resultado de la aplicación de la FFT a la respuesta temporal de la máquina, que previamente fué obtenida utilizando el equipo de medición apropiado. Los síntomas típicos de las vibraciones son picos en, aproximadamente, 3xft y 4xft. También están presentes bandas laterales alrededor de esos picos, dispuestas simétricamente a una distancia aproximada al valor de la frecuencia de rotación (f1) del motor. Estas bandas son provocadas por la existencia de roturas en alguno de los dientes del engranaje.

La figura 4 muestra la información relevante que un experto extraería del espectro de la figura 3, y en la que puede apreciarse la presencia de frecuencias que, según los patrones de fallo conocidos, son indicativas de fallos en la máquina.

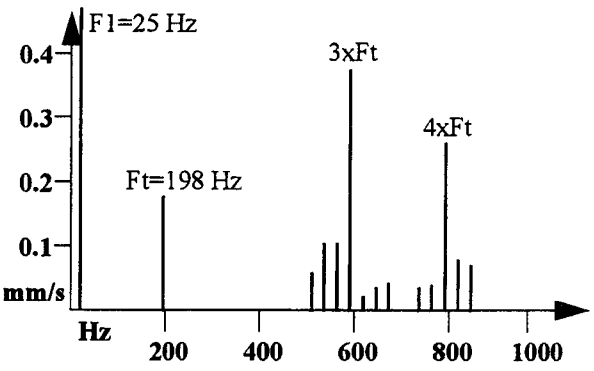



Figura 4: Espectro para el diagnóstico del tren de engranajes.

### 3.1 CARACTERÍSTICAS DE LA BASE DE REGLAS

Es necesaria una base de reglas para cada familia de fallos. De manera que si se están buscando fallos propios de la caja de engranajes (una familia de fallos relacionados por su espectro patrón), entonces será necesaria una base de reglas para diagnosticar tal tipo de fallos.

La información más relevante y útil en la diagnosis de fallos en los engranajes de la maquina de rotación, y que será la que se almacenará en la base de reglas difusas, está compuesta por: el significado que tiene la presencia en el espectro de ciertos harmónicos (Ha) y, como síntoma de deterioro, la relación de variación entre ellos (VHa). Esta seria la información que un experto humano usaría para concluir el estado de la máquina.

El reconocimiento del patrón de un fallo particular será reconocido posteriormente por medio de la base de reglas que se describe a continuación y que está formada por dos variables difusas de entrada y una de salida.

### 3.2 VARIABLES DE ENTRADA

*El promedio de harmónicos (Ha)*, definido como:

$$Ha(i) = [Mag(Tf) + Mag((i+1).Tf)]/2$$

donde i = 1,2,..n, es decir,

$$Ha1 = (Mag(Tf) + Mag(2Tf))/2$$

$$Ha2 = (Mag(Tf) + Mag(3Tf))/2$$

$$Ha3 = (Mag(Tf) + Mag(4Tf))/2$$

y los conjuntos de las funciones de pertenencia para esta variable serán:

Muy Bajo (VL) $0 < Ha < 0.1$

| | |
|---|---|
| Bajo (L) | 0.1 < Ha < 0.2 |
| Medio (M) | 0.2 < Ha < 0.3 |
| Alto (H) | 0.3 < Ha |

*La relación de variación entre harmónicos (VHa),* definida como:

$$VHa(i) = [Ha(t) - Ha(t-1)]/ti$$

donde x = 1,2,..n, y ti es el intervalo de tiempo transcurrido entre mediciones. Es decir,

$$VHa1 = (Ha1(t) - Ha1(t-1))/ti$$

$$VHa2 = (Ha2(t) - Ha2(t-1))/ti$$

$$VHa3 = (Ha3(t) - Ha3(t-1))/ti$$

y los conjuntos de las funciones de pertenencia para esta variable serán:

| | |
|---|---|
| Bajo (L) | 0.0 < VHa < 0.1 |
| Medio (M) | 0.1 < VHa < 0.2 |
| Alto (H) | 0.2 < VHa |

### 3.3 VARIABLE DE SALIDA

La variable difusa de salida de esta base de reglas será *el estado de la máquina (MC)*, y sus funciones de pertenencia son definidos como:

G = buena,
U = útil,
S = todavía aceptable y
N = no aceptable.

### 3.4 DISCRIMINACIÓN DE LA INFORMACIÓN IRRELEVANTE

La clasificación de la información, por medio de la discriminación de aquellos datos no relevantes, se realiza mediante lógica booleana de la manera que se describe a continuación:

(a) *Búsqueda de la frecuencia del dentado:* Tf, 2Tf, 3Tf, etc.

IF Tf and 2Tf then process Ha1
else IF Tf and 3Tf then process Ha2
else IF Tf and 4Tf then process Ha3
end IF function

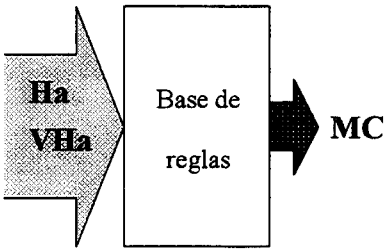
(b) *Búsqueda de bandas laterales* a una distancia igual a la frecuencia de rotación F1 alrededor de 2Tf, 3Tf, 4Tf, etc.

IF Tf $\pm f1$ then process VHa1
else IF 2Tf $\pm f1$ then process VHa2
else IF 3Tf $\pm f1$ then process VHa3
end IF function

(c) *Búsqueda de una condición global* para la determinación del fallo.

IF Ha1 or Ha2 or Ha3 or VHa1 or VHa2 or VHa3
then fallo de la caja de engranajes reconocido.

En las figuras 5 (a) y (b) se muestra la estructura de la base de reglas para la monitorización de la condición del tren de engranajes del caso particular propuesto.




(a)

| Ha \ VHa | VL | L | M | H |
|---|---|---|---|---|
| VL | G | U | S | N |
| L | U | S | N | N |
| M | S | N | N | N |
| H | N | N | N | N |

(b)

Figura 5: (a) Entradas/Salidas de la base de reglas difusas. (b) Estructura de las FdP.

El resultado de procesar las reglas en la función IF, asociado con la consecuente expresión matemática, proporcionará el valor de pertenencia de las variables difusas de entrada Ha y VHa. De acuerdo a esos valores, se procede a la desborrosificación de la base de reglas, obteniendo un estado para la máquina en la gama (G, U, S o N) como se ha explicado previamente.

## 4 RESULTADOS Y CONCLUSIONES

El espectro mostrado en la figura 3 ilustra la apariencia de un fallo en el tren de engranajes. Los valores obtenidos a partir de este espectro para Ha y VHa son Ha=L, VHa=L, y al procesar la base de reglas difusas con esos valores da como resultado, como se muestra en la figura 5 (b), MC = S, lo que significa que el estado de la máquina es todavía aceptable, aunque un fallo en el engranaje está presente, y se aconseja su corrección.

En este trabajo se ha mostrado que la monitorización del estado de una máquina rotatoria puede realizarse con la asistencia de un sistema experto sencillo (ver figura 6) integrado en el equipo de monitorización. Como en todo proceso difuso, hay muchas incertidumbres asociadas a la información manejada y los resultados pueden no ser deterministas, lo que exige la supervisión de un experto humano por medio de inspección visual del espectro. Esta estrategia puede ser mejorada por medio del procesamiento de mayor cantidad de información

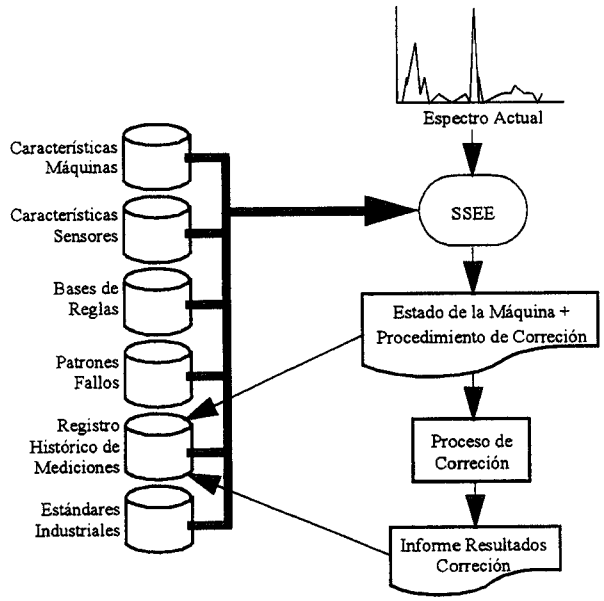



Figura 6: Propuesta para un SSEE integrado en el equipo de medición de vibraciones, que asista en la diagnosis de fallos en maquinas acopladas por ejes.

**Referencias**


[1] Anderson, T. W. (1958). *An Introduction to Multi-Variate Statistical Analysis*. John Wiley & Sons Inc, New York.

[2] Barschdorff, D. and Krämer, S. (1993) Preprocessing for Gearbox Monitoring with a DSP-Based Diagnostic System. *International Conference on Fault Diagnosis*. Tooldiag'93, CERT ONERA, Toulouse, France.

[3] Collacot, R.A. (1977) *Mechanical Fault Diagnosis and Condition Monitoring*. Ed. Chapman and Hall, Great Britain.

[4] Cortina, W. E., Engel, H. L. and Scott, W. K. (1970) *Pattern Recognition Techniques Applied to Diagnosis*. SAE Paper 700497, May.

[5] Ferreiro Garcia and Pardo Martínez (1996) Diagnosis de equipos mecánicos basada en lógica difusa. *ESTYLF'96*. Ed Universidad de Oviedo. pp 125-130.

[6] Isermann, R. and Ulieru, M. (1993) Integrated Fault Detection and Diagnosis. *Proceedings of the IEEE International Conference on Systems Man and Cybernetics* .

[7] Mourot, G., Bousghiri, S. and Ragot, J. (1993) Pattern Recognition for Diagnosis of Technological Systems: A review. *IEEE. SMC. Conference Proceedings*. Le Touquet. France. Vol 5, pp 275-281.

# CONTROLADOR HÍBRIDO DIFUSO-DESLIZANTE DE MOTORES DE CONTINUA

E.Vidal-Idiarte, J.Calvente, R.Giral, R.Leyva y L.Martínez
Dept. Enginyeria Electrònica,Elèctrica i Automàtica
ETSE-URV
Carretera de Salou, s/n
43006-Tarragona
e-mail:envidal@etse.urv.es

## Resumen

En este trabajo se presenta un controlador híbrido difuso-deslizante para el control de velocidad en motores de continua. El sistema difuso decide cual es la mejor consigna de intensidad que ha de seguir la superficie deslizante de control. De esta manera se obtiene un control robusto a perturbaciones en la carga y en la alimentación de entrada.

**Palabras Clave:** Control difuso, control deslizante, control no-lineal, control de motores.

## 1 INTRODUCCIÓN

A causa de su simplicidad y versatilidad los motores de continua ocupan un amplio espectro de las aplicaciones con velocidad variable. Esta simplicidad es debida a la típica construcción mecánica de una máquina de continua. El conmutador mecánico en la máquina continua y la orientación de las escobillas, que suministran corriente hacia la armadura, fuerzan ortogonalidad entre la fuerza magnetomotriz producida por la corriente de armadura y la excitación principal. Como resultado, para cualquier excitación a la entrada de la máquina de continua, el voltaje de armadura determina la velocidad operativa mientras que la fuerza magnetomotriz de ésta determina el par desarrollado. Bajo una excitación constante, la máquina de continua es esencialmente un sistema lineal. De aquí proviene la simplicidad de su control.

Desde hace años y para incrementar la eficiencia del sistema, el control en anchura de pulsos(PWM) ha sido extensamente utilizado en mecanismos de velocidad variable, resultando entonces un comportamiento dinámico global no-lineal. Esta técnica está siendo sucesivamente sustituida por algoritmos de control en modo deslizante[10][8] debido a la facilidad de implementación de la acción de control discontinua. Además el modo deslizante proporciona una baja sensibilidad a perturbaciones y variaciones en los parámetros del motor.

Por otra parte, muchas aplicaciones requieren un motor con una rápida respuesta a consignas externas con mínimo rebasamiento, lo qual es difícil de conseguir con algoritmos de control PID y PWM. Para conseguir ésto último, se suele utilizar una combinación de superficies deslizantes, una que nos asegure la respuesta rápida del sistema y que al mismo tiempo proteja al motor contra sobreintensidades y otra que es la que regirá el comportamiento en régimen estacionario del sistema.

En este trabajo, se presenta un controlador híbrido difuso-deslizante que realiza de manera suave el cambio entre las diferentes superficies que se utilizan en el control de velocidad del motor, consiguiendo un controlador robusto a perturbaciones en el par debido en la carga y que no presenta sobrepicos de corriente perjudiciales para el motor. En este trabajo , al igual que en [1],[3],[2] y [7], se utiliza la lógica difusa como segundo lazo de control para mejorar el comportamiento de otros tipos de control.

## 2 CONTROL DIFUSO-DESLIZANTE

### 2.1 MODELO DEL MOTOR

Las ecuaciones que describen la dinámica del sistema que se muestra en la figura 1 son las siguientes:

$$iR + L\frac{di}{dt} + Kw = \begin{array}{l}\nearrow \text{ V durante Ton} \\ \searrow \text{ - V durante Toff}\end{array} \qquad (1)$$

$$Ki = J\frac{dw}{dt} + Bw + T_L \qquad (2)$$

donde J es la inercia del rotor, B es el coeficiente de fricción viscosa, i es la corriente de armadura, R es la resistencia de armadura, w es la velocidad angular del motor, $T_L$ es el par debido a la carga, L es la inductancia de armadura y K es el coeficiente que caracteriza la conversión electromecánica de la corriente de armadura en par.

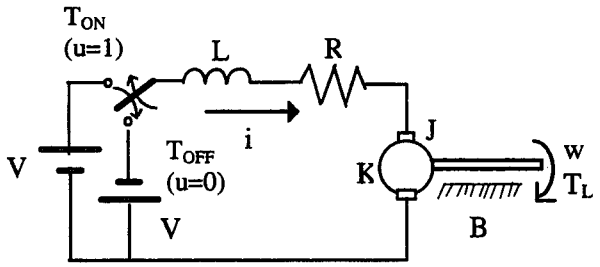



Fig. 1: Variables electromecánicas y parámetros en un motor dc chopper-fed.

## 2.2 DINÁMICA EN MODO DESLIZANTE

Un análisis del modo de deslizamiento de las ecuaciones anteriores, muestra que la superfície correspondiente a velocidad angular constante, no nos proporciona un régimen de deslizamiento a causa de que no se cumple la condición de transversalidad. Esto es debido a que el sistema a controlar es de grado relativo dos lo que obliga a que para que exista régimen deslizante la superfície S(x) ha de estar definida como:

$$S(x) = (w_R - w) - \frac{dw}{dt} \qquad (3)$$

siendo $w_R$ la velocidad angular deseada.

Utilizando esta superficie y bajo régimen de deslizamiento se obtendrá una respuesta transitoria de primer orden con error estacionario nulo. El problema yace en que en un motor de continua la derivada de la velocidad de salida no es físicamente accessible. El método alternativo utilizado[10] es construir la superficie de deslizamiento en términos de la velocidad angular y la corriente de armadura, debido a que si la fricción es baja y el par debido a la carga es constante o varía lentamente entonces la ecuación (3) puede ser aproximada por:

$$S(x) = K_1 ( w_R - w) - R_s i \qquad (4)$$

Utilizando combinaciones lineales de la velocidad angular y la corriente de armadura del tipo:

$$S(x) = K_1 ( w_R - w) - i \qquad (5)$$

se cumple la condición de transversalidad y la dinámica de deslizamiento ideal que se obtiene es la siguiente[9]:

$$i = K_1 (w_R - w) \qquad (6)$$

$$\frac{dw}{dt} = \frac{K}{J} i - \frac{B}{J} w - \frac{T_L}{J} \qquad (7)$$

Como se puede observar, el sistema original de segundo orden ha sido reducido a una dinámica estable de primer orden. Si se observa la superficie se puede ver que en régimen de deslizamiento( S(x)=0) siempre nos quedará un error estacionario que depende del valor de $K_1$ , el qual habría de ser infinito para que el error estacionario fuera cero. También podemos observar que la dinámica de deslizamiento ideal no depende de la tensión de alimentación, pero sí que depende del par debido a la carga, el qual hemos supuesto constante para poder utilizar la corriente de armadura como estimación de la aceleración angular.

## 2.3 CONTROL CON LÓGICA DIFUSA

Como hemos dicho anteriormente para controlar motores en modo deslizante se suele utlizar una combinación de superficies. Unas ((8) y (10)) que nos aseguran que el sistema tendrá un arranque rápido (arranque a intensidad máxima constante) y otras que regirán el régimen estacionario(9).

$$S(x) = i_{max} - i \qquad (8)$$

$$S(x) = K_1 * (w_R - w) - i \qquad (9)$$

$$S(x) = -i_{max} - i \qquad (10)$$

Un problema que suele aparecer al realizar la conmutación brusca entre las diferentes superficies es la aparición de sobrepicos de corriente que pueden dañar al motor. Para evitar ésto se propone la realización de una transición más suave entre las diferentes superficies de deslizamiento.

Utilizando el método difuso de Takagi-Sugeno se realizará la transición suave entre las superficies de deslizamiento, ya que éstas son combinaciones lineales de las variables de estado y pueden formar parte de los consecuentes de las reglas de un sistema difuso de este tipo. El problema que se presenta es que de S(x) nos interesa sólo su signo y como cada una de las diferentes superficies posee frecuencias de conmutación diferentes, se puede dar el caso de que en algún momento se enmascare el valor de la superficie a utilizar. Para solucionar ésto se propone utilizar el sistema difuso como segundo lazo de control(sistema de decisión) y utilizar como superficie deslizante una que esté formada por variable común a todas las superficies y una referencia proporcionada por el sistema difuso. De esta manera, interpolaremos entre los valores que habría de tener la intensidad si estuvieramos en régimen deslizante en cada una de las superficies y obtendremos así el valor de referencia más adecuado en cada momento.

Siguiendo esta idea se propone una superficie deslizante de control del motor del tipo:

$$S(x)= i_{ref} - i \quad (11)$$

A partir de las ecuaciones (8),(9) y (10) obtenemos las intensidades correspondientes a cada una de las superficies a utilizar y utilizando un sistema de lógica difusa del tipo Takagi-Sugeno decidimos cual es la $i_{ref}$ más adecuada en cada momento. Al realizar una interpolación entre las diferentes superficies se evita la aparición de sobreimpulsos de corriente que pueden ser perjudiciales para el motor.

Para eliminar el error estacionario y hacer que el sistema sea robusto a perturbaciones en la carga, se suele añadir un término integral($K_2$) a la superfície de control que lo compense:

$$S(x)= K_1 ( w_R - w) - i + K_2 \quad (12)$$

El problema en la implementación analógica es que este término integral acumula todo el error del transitorio y no solamente el error estacionario, lo que provoca sobreimpulsos no deseados en la respuesta transitoria. Para solucionar este problema se propone añadir a la superficie un término integral que solamente integre a partir de la aparición del error estacionario y que esté desactivado durante el arranque. Para conseguir ésto ampliamos nuestro sistema difuso con dos variables de entrada más ( cerror, i ), que son respectivamente el cambio en el error y en la corriente de armadura, las cuales nos permitirán implementar el término integral y decidir su zona de actuación para conseguir al final un controlador robusto, con mínimo rebasamiento y de rápida respuesta transitoria. Éste término integral aparece en las ecuaciones (16), (19) y (21). También se le ha añadido un término derivativo que intenta reducir los sobreimpulsos. Éste término aparece en la ecuaciones (22) y (23).

If (error is Pos) then (iref =imax) (13)
If (error is PS) and (cerror is P) then (iref =imax) (14)
If (error is PS) and (cerror is N) then (iref=$k_1$ *error) (15)
If (error is PS) and (cerror is Z) then (iref= i+δ*error) (16)
If (error is NS) and (cerror is N) then (iref=-imax) (17)
If (error is NS) and (cerror is P) then (iref= $k_1$ *error) (18)
If (error is NS)and (cerror is Z) then (iref = i+δ*error) (19)
If (error is Neg) then (iref =-imax) (20)
If (error is Ze) and (cerror is Ze) then (iref= i+δ*error) (21)
If (error is Ze) and (cerror is N) then (iref= i+γ*cerror) (22)
If (error is Ze) and (cerror is P) then (iref= i+γ*cerror) (23)

Las funciones de pertenencia de las variables error y cerror son las siguientes:

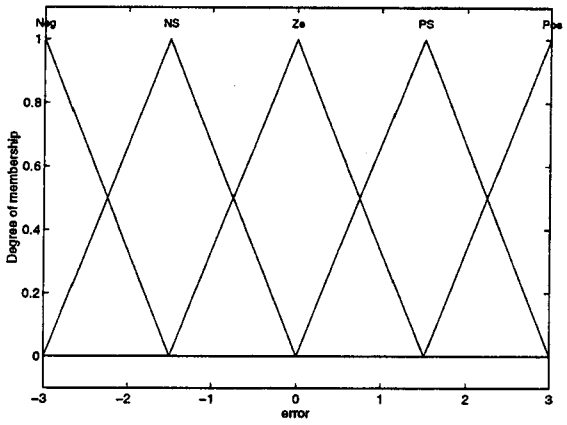



Figura 2: Funciones de pertenencia de la variable error

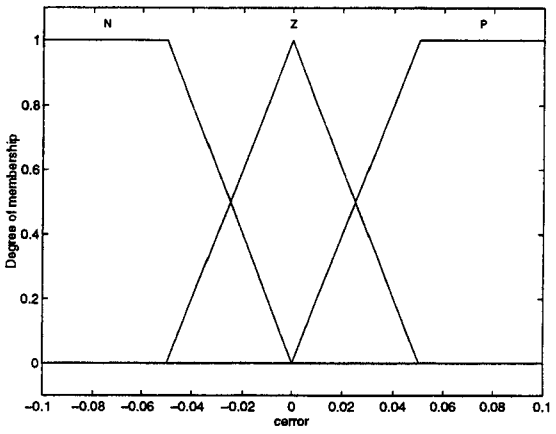



Figura 3: Funciones de pertenencia de la variable cambio en el error.

## 2.4 SIMULACIÓN

Para la simulación se han utilizado los parámetros de un motor DC de imanes permanentes de 200 W de potencia máxima( MAE S.p.A M588 110) cuyos parámetros son $K_T=K_e$=0.105 N*m/A, V=60 V, R=1.6 Ω, L=5.2 mH, B=6e-6 (N*m)/(rad/s) siendo la corriente continua máxima igual ± 3 A. Por esta causa y habiendo decidido de provocar el cambio de superficie cuando el error sea igual a ±3 se ha escogido un valor de $K_1$=1 para evitar sobreimpulsos de corriente. También se ha escogido δ=2, γ=20 y 50 KHz como frecuencia de muestreo del sistema difuso.

En las figuras 4 y 5 podemos observar la respuesta rápida del sistema con mínimo rebasamiento a un cambio en la consigna del motor.

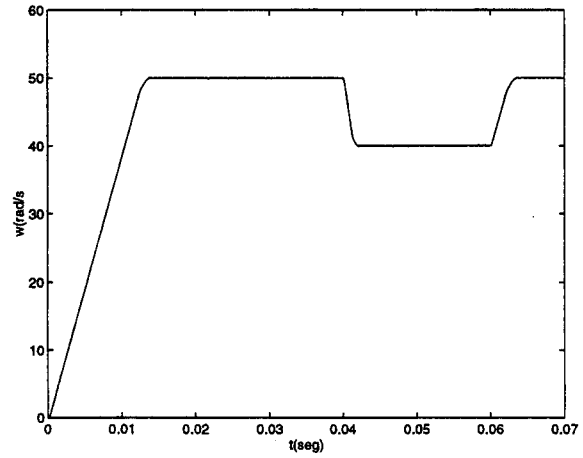



Figura 4: Arranque y respuesta del motor a un cambio en la consigna.

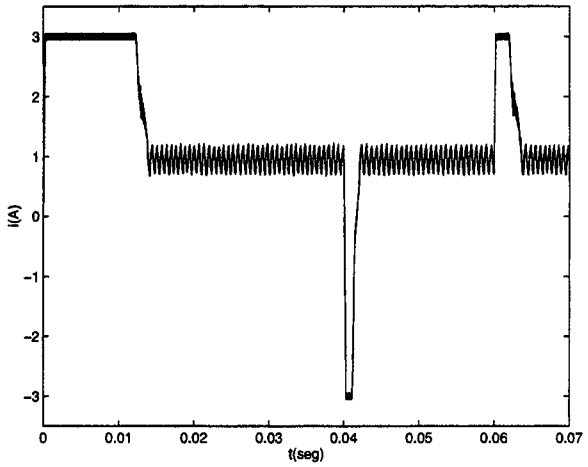



Figura 5: Corriente de armadura en el arranque y en el cambio de la consigna.

En las figuras 6 y 7 se puede observar el transitorio de arranque del motor para velocidades iguales a 18 y 200 rad/s respectivamente. En el estado estacionario se provoca al motor un decremento de 10 V en la tensión de alimentación entre t=0.02-0.04 s (w=18 rad/s) y entre t=0.08-0.12 s (w=200 rad/s). También se provoca un aumento del par debido a la carga del 100%(de 0.1 a 0.2 ) en t=0.06 s (w=18 rad/s) y en t=0.16 s (w=200 rad/s). Tal como se puede observar el sistema es robusto a variaciones en la tensión de alimentación y recupera el estado estacionario después de un aumento brusco de la carga. Podemos observar la evolución de la corriente de armadura en las figuras 7 y 8 para w=18 y w=200 rad/s respectivamente.

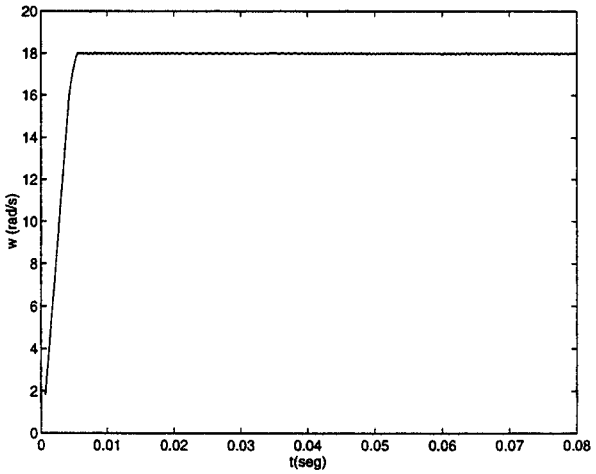



Figura 5: Velocidad angular para $w_{ref}$= 18 rad/s.

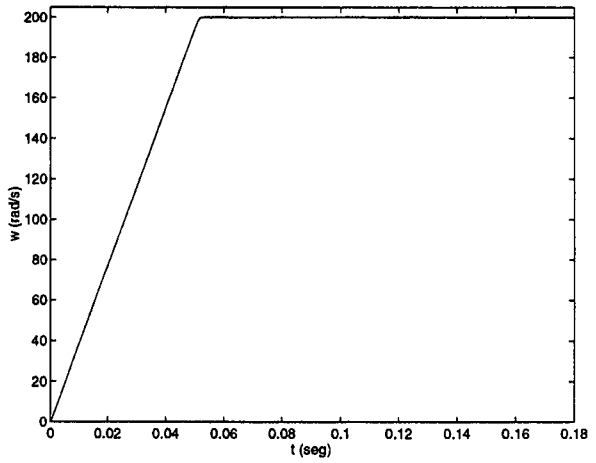



Figura 6: Velocidad angular para $w_{ref}$= 200 rad/s.

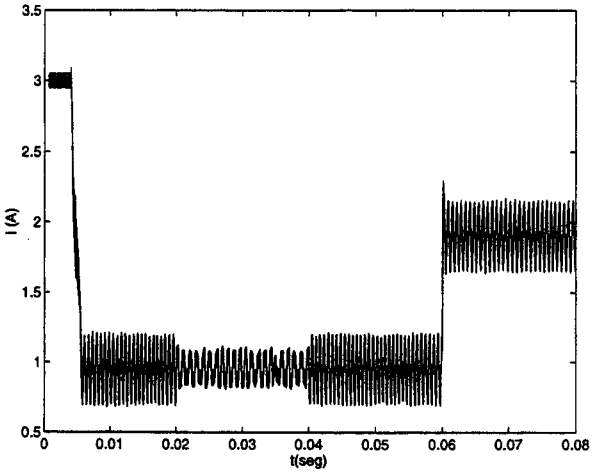



Figura 7: Corriente de armadura para $w_{ref}$= 18 rad/s.

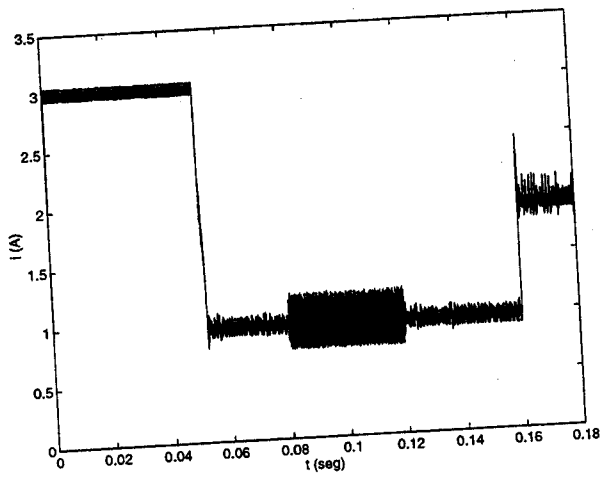



Figura 8: Corriente de armadura para $w_{ref}$= 200 rad/s.

Se puede observar en las gráficas correspondientes a las corrientes de armadura la ausencia de sobreimpulsos de corriente en el momento del cambio de superficie, lo cual repercute positivamente en la duración del motor.

## 3 CONCLUSIONES

En este trabajo se ha propuesto un mecanismo difuso para la implementación de controladores de motores en modo deslizante en los cuales se realiza una conmutación entre diferentes superficies de control. En los resultados de simulación se puede observar que se obtiene un transitorio de arranque a máxima velocidad y una respuesta en régimen estacionario robusta a perturbaciones en la carga y la alimentación.

### Referencias

[1] A. Ishigame, T. Furukawa, S. Kawamoto and T. Taniguchi, "Sliding Mode Controller Design Based on Fuzzy Inference for Nonlinear Systems,". IEEE Trans. Ind. Electronics, vol 0, n°. 1, February 1993 pp 64-70.

[2] J. Kim, K. Kim and E. Chong, "Fuzzy Precompensated PID Controllers," IEEE Trans. on Control Systems Technology, vol. 2, no.4, December 1994 pp 406-411.

[3] Y. Kung and C. Liaw, "A Fuzzy Controller Improving a Linear Model Following Controller for Motor Drives," IEEE Trans. Fuzzy Systems. vol. 2. no. 3, August 1994 pp 194-202.

[4] Kuo, Tal. DC Motors and Control Systems. SRL Publishing Company, 1978.

[5] C.C.Lee. "Fuzzy logic in control system: Fuzzy logic controller-Part I & II.," IEEE Trans. Syst. Man. Cybern.. vol 20. n0. 2. pp. 404-435. 1990.

[6] H. Sira-Ramirez, "Differential geometric methods in variable-structure control," Int. J. Control, vol. 48, n° 4, pp 1359-1390.

[7] A. Suyitno, J. Fujikawa, H. Kobayashi and Y. Dote, "Variable-Structured Robust Controller by Fuzzy Logic for Servomotors," IEEE Trans. Ind. Electronics, vol.40, no.1, February 1993 pp80-88.

[8] V. I. Utkin, "Sliding Mode Control Design Principles and Applications to Electric Drives" IEEE Trans. on Industrial Electronics, vol. 40, n° 1, February 1993 pp 23-36.

[9] E.Vidal-Idiarte, R.Leyva, J.Calvente, R.Giral and L.Martínez, "Sliding Adaptive Controller by Fuzzy Logic for Servomotor". Proceedings of Int. Conference on Automatic Control (Controlo'96), Oporto, September 1996, Vol. 2, pp. 683-686.

[10] R. Venkataraman. "Sliding Mode Control of Power Converters", Ph. D. Dissertation, California Institute of Technology 1986.

# SEGUIMIENTO DEL PUNTO DE MÁXIMA POTENCIA DE UN SISTEMA FOTOVOLTAICO MEDIANTE CONTROL DIFUSO.

**Spartacus Gomáriz**
Escuela Universitaria Politécnica.
UPC. Vilanova i la Geltrú
Av. Victor Balaguer s/n 08800
Vilanova i la Geltrú. Barcelona.
e-mail: spartacus@eel.upc.es

**Juan Andrés Guerrero**
Escuela Universitaria Politécnica.
UPC. Vilanova i la Geltrú
Av. Victor Balaguer s/n 08800
Vilanova i la Geltrú. Barcelona.

**Francisco Guinjoan**
E.T.S. d'Enginyers de
Telecomunicació. UPC.
Departament d'Enginyeria
Electrònica. Modul C4. C/Gran
Capità s/n. 08034. Barcelona
e-mail: guinjoan@eel.upc.es

## Resumen

En este trabajo se presenta un controlador difuso que permite el Seguimiento del Punto de Máxima Potencia (SPMP) de un sistema fotovoltaico constituido por un panel solar, un convertidor conmutado dc-dc y una batería.

Los resultados de simulación obtenidos incluyen modelos depurados tanto del panel solar como de la batería, y muestran las buenas prestaciones del control diseñado en términos de respuesta dinámica y rendimiento energético.

**Palabras Clave:** Sistema fotovoltaico, control difuso, convertidor dc-dc, SPMP, acumulador.

## 1 INTRODUCCIÓN.

El estudio de los sistemas fotovoltaicos se ha intensificado mucho en los últimos tiempos debido a las características de esta fuente de energía, limpia, segura, y virtualmente inagotable.

Aun así, los módulos fotovoltaicos se ven afectados por condiciones tanto de fabricación como ambientales que influyen en su comportamiento y eficiencia. La dificultad de modelar estas condiciones de comportamiento permite pensar en diseñar un algoritmo difuso de control basado en reglas lingüísticas que describa la estrategia de control de seguimiento del punto de máxima potencia.

El sistema analizado se representa en la figura 1 y corresponde a un sistema fotovoltaico autónomo con acumulación por batería, para aplicaciones terrestres.

De la figura se desprende que el sistema consta de:

1) Sistema fotovoltaico.

2) Sistema acumulador.

3) Sistema regulador.

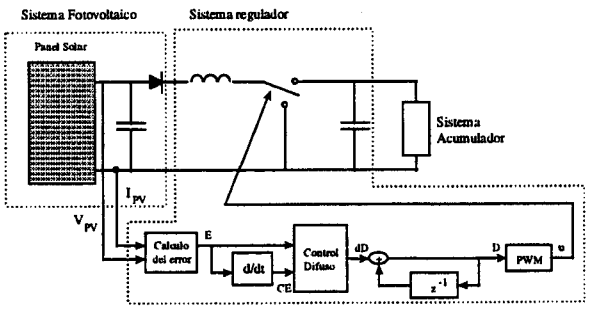



Figura 1: Sistema fotovoltaico con acumulación por batería.

El objetivo de este control es polarizar al generador fotovoltaico en su punto de máxima potencia en todo momento y bajo cualquier condición. Se trata pues de un seguidor del punto de máxima potencia (SPMP) que realiza su función mediante la realimentación de algunas

señales del convertidor conmutado elevador dc/dc con un control difuso. Se utiliza el rizado de conmutación, intrínseco a los convertidores conmutados, como alternativa a la perturbación externa para conducir el sistema al punto de máxima potencia.[4]

# 2 MODELADO DEL SISTEMA

## 2.1 SISTEMA FOTOVOLTAICO.

Este sistema realiza la conversión fotovoltaica mediante una matriz de células solares.

El modelo del módulo fotovoltaico utilizado describe el comportamiento de la tensión de la matriz de células solares, $V_{PV}$, a partir de la corriente extraída, $I_{PV}$. [1]

$$V_{PV} = V_t \cdot \ln\left(\frac{I_{ph} + I_o - I_{PV} - \frac{(I_{PV} \cdot R_s + V_{PV})}{R_{sh}}}{I_o}\right) - I_{PV} \cdot R_s \quad (1)$$

Siendo :

Iph : intensidad de iluminación.[A]

Io : Corriente inversa de saturación. [A]

$I_{PV}$ : Corriente extraída del módulo fotovoltaico. [A]

Rs : Resistencia serie del módulo. [Ω]

Rsh : Resistencia paralelo del módulo. [Ω]

Vt : Potencial térmico. [V]

La representación gráfica del modelo expresado en (1) se puede ver en la curva dada en la figura 2.

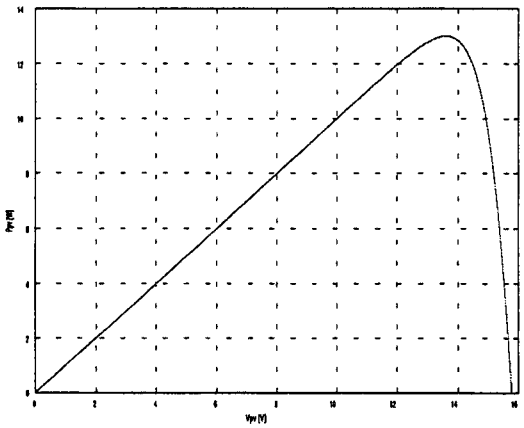


Figura 2: Característica del módulo fotovoltaico, potencia-tensión.

En la figura 2. Se refleja la problemática de los SPMP en lo que respeta al módulo fotovoltaico. En la curva Vpv-Ppv, se observa como el máximo de tensión y el máximo de potencia no coinciden, al contrario, la maximización de la tensión provoca que el modulo fotovoltaico entre en cortocircuito y que no se extraiga ninguna potencia de él.

De modo similar la maximización de la corriente no logra extraer potencia alguna del módulo.

Consecuentemente es necesario obtener la potencia del módulo fotovoltaico como variable útil para el diseño del controlador.

## 2.2 EL SISTEMA ACUMULADOR.

Uno de los mayores problemas de la energía solar es el consumo de esta energía en momentos de escasez energética, es decir, el consumo en diferido. Una solución a este problema, es la acumulación de la energía extraída de los módulos en baterías. En este trabajo se modela la batería electroquímica plomo-ácido.

La batería esta formada por una serie de celdas electroquímicas individuales de 2V. Su funcionamiento viene modelado dependiendo del estado de carga o descarga de la batería [5]. La ecuación que rige la carga de la batería es:

$$V_b = V_r + \frac{I_b}{C} \cdot \left(R_i + \frac{0.189}{1.142 - EDC}\right) + (EDC - 0.9) \cdot \ln\left(1 + 300 * \frac{I_b}{C}\right) \quad (2)$$

Siendo :

Vb: la tensión en terminales de la batería. [V]

Ib: la corriente que circula por la batería. [A]

Vr: la tensión interna del circuito equivalente de Thevenin. [V]

C: la capacidad de almacenamiento de la celda en amperios-hora, referenciada a un régimen de descarga de 500 horas.

Ri: Resistencia interna de las celdas. [Ω]

EDC: estado de carga. Es el cociente entre la capacidad de la celda en el momento de la medida y la capacidad máxima, su valor esta entre 0 y 1 y es adimensional.

## 2.3 SISTEMA REGULADOR.

El comportamiento dinámico del convertidor elevador puede ser descrito mediante el siguiente conjunto de ecuaciones bilineales. [3]

$$\begin{bmatrix} di(t)/dt \\ dv(t)/dt \end{bmatrix} = \left( \begin{bmatrix} 0 & -1/L \\ 1/C & -1/RC \end{bmatrix} \cdot \begin{bmatrix} i(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} E/L \\ 0 \end{bmatrix} \right) \cdot (u) + \left( \begin{bmatrix} 0 & 0 \\ 0 & -1/RC \end{bmatrix} \cdot \begin{bmatrix} i(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} E/L \\ 0 \end{bmatrix} \right) \cdot (1-u) \quad (3)$$

Siendo u el estado del interruptor (u=1 ON, u=0 OFF).

A partir de las ecuaciones (3) se puede ver que la corriente fluye continuamente, tan solo afectada por el rizado de conmutación. Este hecho permite polarizar al módulo fotovoltaico en un solo punto, sin que exista ningún corte de corriente brusco.

Refiriéndonos a la figura 1, el interruptor del convertidor elevador viene controlado por la señal generada por un modulador de ancho de pulso (PWM), en la entrada del cual se introduce el ciclo de trabajo D, del convertidor, generado por el control difuso.

Los componentes (L, C, transistor y diodo) y el parámetro $T_s$ (Periodo de trabajo del convertidor) del sistema regulador deben diseñarse para disminuir los rizados y las disipaciones de potencia, además de asegurar la no anulación de la corriente en el inductor.

# 3 SISTEMA SEGUIDOR DEL PUNTO DE MAXIMA POTENCIA, SPMP, MEDIANTE EL RIZADO DE CONMUTACIÓN.

El problema del seguimiento del punto de máxima potencia se ha resuelto habitualmente basándose en técnicas de perturbación del ciclo de trabajo, es decir, mediante la variación sistemática del ciclo de trabajo del convertidor. El objetivo es detectar el punto de trabajo instantáneo del sistema fotovoltaico, posición instantánea y la dirección en el que debe evolucionar para alcanzar la máxima potencia. Esto se realiza mediante la variación del ciclo de trabajo del convertidor que ajusta la tensión $V_{PV}$ o la corriente $I_{PV}$ a los valores deseados.

Este proceso consiste de hecho en la medición de la derivada de $P_{PV}$ con respecto a una de las variables del módulo fotovoltaico, $I_{PV}$ o $V_{PV}$.

La perturbación es necesaria para obtener las variaciones de potencia extraída que permitirán calcular $(dP_{PV}/dV_{PV})$ o $(dP_{PV}/dI_{PV})$.

Posteriormente se conduce al sistema de manera que $dP_{PV}/dI_{PV}$ o $dP_{PV}/V_{PV} \cong 0$. Es fácil interpretar esta derivada como el error del seguimiento. Ahora bien , este proceso es difícil de implementar sin un control activo y memoria debido a que la medida de la variación de las variables se hace a través de varios ciclos de trabajo para obtener una buena relación señal-ruido.[6][7][8]

El sistema descrito anteriormente sigue el máximo punto de potencia en escalas de segundos e incluso mayores Además de su lentitud hay que considerar que el método de la perturbación externa no permite desarrollar un control óptimo ya que para su buen funcionamiento es necesario realizar la perturbación del convertidor de forma continua incluso cuando el módulo fotovoltaico se encuentra polarizado en el punto deseado .

Alternativamente el método desarrollado por Pallab Midja et al.[4] prescinde de la perturbación externa del ciclo de trabajo ya que encuentra y explota una perturbación intrínseca a los convertidores conmutados, esto es, el rizado de conmutación.

Este rizado puede ser considerado como una perturbación dinámica y es por tanto posible sustituir la perturbación externa por esta, con la importante ventaja de que de este modo se extrae información de las derivadas de $P_{PV}$ en cada ciclo y es en cada ciclo cuando se activa la señal de error. De esta manera las medidas realizadas mediante el rizado permiten el seguimiento a escalas mucho menores (milésimas de segundo) evitando así las necesidades de memoria y promediación de las variables de estado.

# 4 CONTROLADOR DIFUSO PROPUESTO

El método se basa en el cálculo de la pendiente del rizado de la potencia en relación a la pendiente de la otra variable que se utiliza para definir el error. Se escoge $V_{PV}$ y no $I_{PV}$ debido a que en puntos cercanos al máximo de potencia, el rizado de $V_{PV}$ es mayor que el de $I_{PV}$, consiguiendo de esta manera una aproximación de la derivada discreta más efectiva. Se define entonces el error del sistema "E" como :

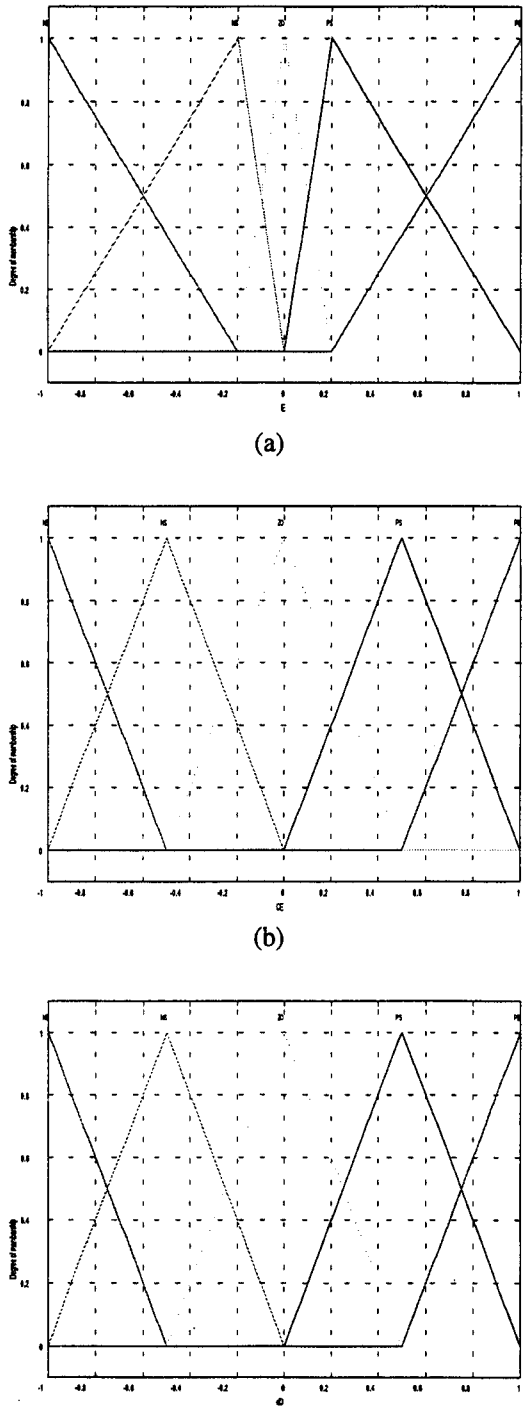
$$E = -\frac{P_{PV}(k) - P_{PV}(k-1)}{V_{PV}(k) - V_{PV}(k-1)} \qquad k = n \cdot \frac{T_s}{2} \qquad T_s = 20KHz \quad (4)$$

Se define también "CE", el cambio del error, de la siguiente manera.

$$CE = E(k) - E(k-1) \quad (5)$$

El control difuso que se propone toma como variables de entrada E y CE, y como única variable de salida dD correspondiente a la variación del ciclo de trabajo del convertidor.

Para cada variable E, CE y dD se han definido 5 funciones de pertenencia que se pueden ver en la figura 3.



(a)

(b)

(c)

Figura 3: funciones de pertenencia para (a) entrada E, (b) entrada CE, (c) salida dD. NB: Negativo grande. NS: Negativo pequeño. ZO: Zero. PS: Positivo pequeño. PB: Positivo grande

El conjunto de reglas difusas diseñadas y los pesos asociados a cada regla son:

1. If (E is NB) and (CE is NB) then (dD is ZO) (1)
2. If (E is NB) and (CE is NS) then (dD is ZO) (1)
3. If (E is NB) and (CE is ZO) then (dD is NB) (1)
4. If (E is NB) and (CE is PS) then (dD is NB) (1)
5. If (E is NB) and (CE is PB) then (dD is NB) (1)
6. If (E is NS) and (CE is NB) then (dD is NS) (1)
7. If (E is NS) and (CE is NS) then (dD is ZO) (1)
8. If (E is NS) and (CE is ZO) then (dD is NS) (0.7)
9. If (E is NS) and (CE is PS) then (dD is NS) (0.7)
10. If (E is NS) and (CE is PB) then (dD is NS) (1.5)
11. If (E is ZO) and (CE is NB) then (dD is NB) 1.3)
12. If (E is ZO) and (CE is NS) then (dD is ZO) 1.3)
13. If (E is ZO) and (CE is ZO) then (dD is ZO) 1.7)
14. If (E is ZO) and (CE is PS) then (dD is ZO) (1.3)
15. If (E is ZO) and (CE is PB) then (dD is PS) (0.7)
16. If (E is PS) and (CE is NB) then (dD is ZO) (1)
17. If (E is PS) and (CE is NS) then (dD is PS) (0.7)
18. If (E is PS) and (CE is ZO) then (dD is PS) (0.7)
19. If (E is PS) and (CE is PS) then (dD is ZO) (1)
20. If (E is PS) and (CE is PB) then (dD is PS) (1)
21. If (E is PB) and (CE is NB) then (dD is PB) (0.5)
22. If (E is PB) and (CE is NS) then (dD is PB) (1)
23. If (E is PB) and (CE is ZO) then (dD is PB) (1)
24. If (E is PB) and (CE is PS) then (dD is ZO) (1)
25. If (E is PB) and (CE is PB) then (dD is ZO) (1)

El método de inferencia aplicado es el de Mamdani con composición MAX-MIN y en el proceso de defucificación es el método de centro de áreas (COA). [2]

La ley de control generada por este control difuso puede observarse en la figura 4.

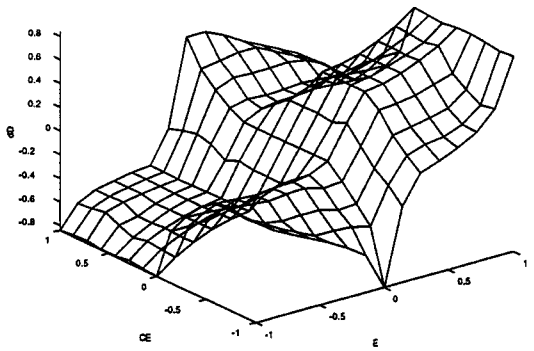


Figura 4: función generada por el control difuso

## 5 RESULTADOS DE SIMULACION

En la figura 5, 6 y 7 se dan las evoluciones temporales de la potencia extraída (P), corriente (I) y tensión (V)

suministrada por el módulo fotovoltaico ante transiciones bruscas de iluminación (Q) de 0 a 100 W/m$^2$ (Condiciones de madrugada o atardecer), de Q=100W/m$^2$ a Q=1000W/m$^2$ (Condiciones de mediodía) y ante una transición suave de Q=1000W/m$^2$ a Q=100W/m$^2$.

Los rendimientos de potencia extraídos del módulo fotovoltaico son de 97.13% para la primera zona de Q=100W/m$^2$ de 99.98% para la zona de Q=1000W/m$^2$ y otra vez 97.13% para la segunda zona de Q=100W/m$^2$

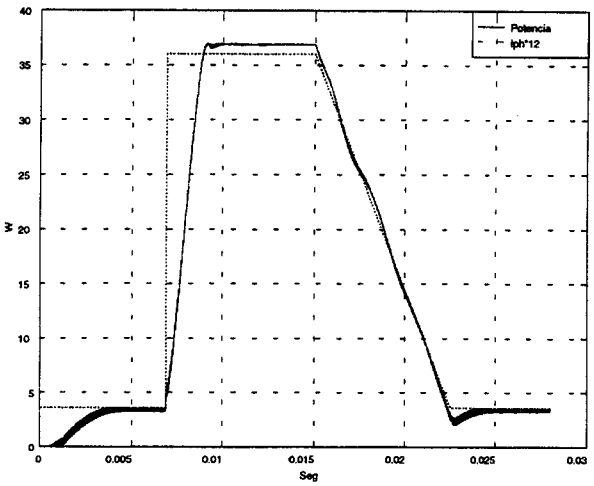



Figura 5: evolución temporal de la potencia extraída del módulo fotovoltaico, referenciada a la intensidad de iluminación.

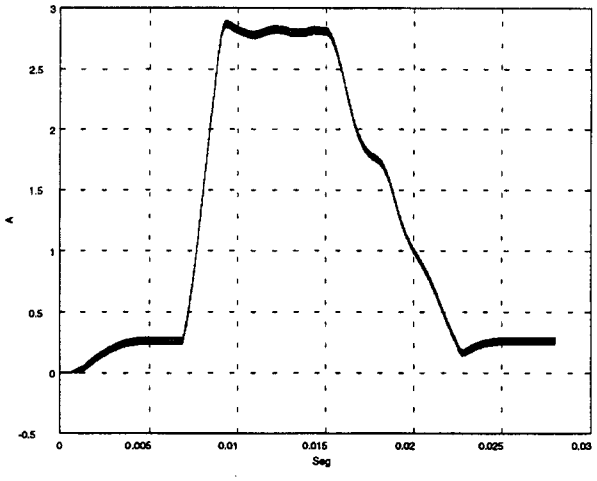



Figura 6: evolución temporal dela corriente extraída del módulo fotovoltaico.

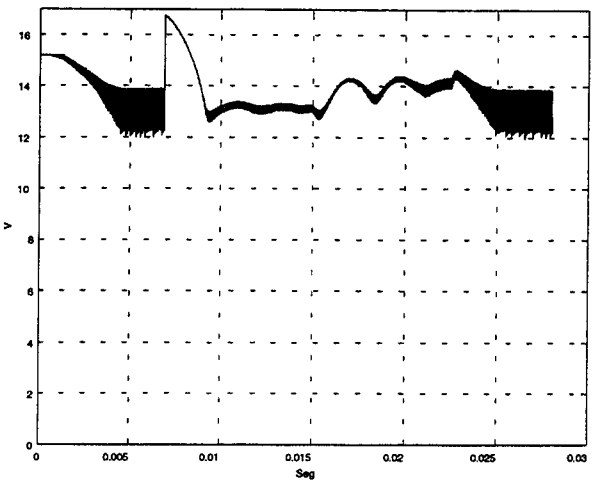



Figura 7: evolución temporal de la tensión extraída del módulo fotovoltaico.

# 6 CONCLUSIONES

En este trabajo se ha presentado un Seguidor del Punto de Máxima Potencia utilizando un control difuso aplicado sobre un convertidor elevador, utilizando el rizado de conmutación intrínseco del propio convertidor.

Los resultados obtenidos en la simulación garantizan el seguimiento de los diferentes puntos de máxima potencia tanto en transiciones bruscas como suaves de las condiciones medioambientales de iluminación, con respuestas dinámicas satisfactorias.

Los rendimientos de extracción de potencia del módulo fotovoltaico están comprendidos entre el 97.13% y el 99.98%.

**Agradecimientos**

Este trabajo ha sido subvencionado parcialmente por la comisión Internacional de Ciencia y Tecnología (CICYTI) TIC96-1195.

**Referencias**

[1] Castañer, Luis. (1994). Energía solar fotovoltaica. *Ediciones UPC, Politext, 21.* 1ª edición. Barcelona.

[2] Driankov D., Helledoorn H., Reinfrank M. (1993). Introduction to fuzzy control. *Springer Verlarg. Heidelberg.*

[3] Erickson R.W., CuK S., Middlebrook R.D. (1982). Large-signal modelling and analysis of switching regulators. *IEEE Proc. PESC-82.* Pág 240-250.

[4] Pallab Midya and et al.(1996). Dynamic Power Point Tracker for Photovoltaic Applications. *Aerospace Power Electronics II.* Pág. 1710-1716

[5] Ravi Varma. (1991). Techniques for characterization of electrodes and elecrochenical proceses. *New York Wiley Cop.*

[6] Sullivan C.R. and Powers M.J.(1993). A High-Efficiency Maximun Power Point Tracker for Photovoltaic Arrays in a solar-Powered Race Vehicle. *IEEE Proc. PESC-93.* Pág. 574 -580.

[7] Vittorio V., Corsi S., Lambri L. (1982). Maximun Power Point Tracker for Photovoltaic Plants. *Proc IEEE PSC-82.* Pág. 507-512

[8] Won C.Y., Kim D.H., Kim S.C., Kim W.S., Kim H.S.(1994). A New Maximun Power Point Tracker of Photovoltaic Arrays Using Fuzzy Controller. *IEEE Proc. PESC-94.* Pág. 396-403.

# XFUZZY 2.0: UN ENTORNO DE DESCRIPCIÓN, VERIFICACIÓN Y SÍNTESIS DE SISTEMAS BASADOS EN LÓGICA DIFUSA

**D. López[*], F. J. Moreno[**], C. J. Jiménez[**], S. Sánchez-Solano[**], A. Barriga[**]**

[*] Centro Informático Científico de Andalucía
[**] Instituto de Microelectrónica de Sevilla - Centro Nacional de Microelectrónica.
Edificio CICA, Avda. Reina Mercedes s/n, 41012-Sevilla.
email: barriga@imse.cnm.es

## Resumen

Xfuzzy 2.0 constituye un entorno para la descripción, verificación y síntesis de sistemas basados en lógica difusa. Entre sus características cabe destacar la utilización de un lenguaje de especificación potente y flexible para describir la estructura de un sistema y definir las operaciones difusas involucradas. El entorno Xfuzzy incluye una serie de herramientas gráficas que facilitan la descripción de un sistema difuso, así como utilidades para verificar y refinar su comportamiento. También incorpora módulos de síntesis de sistemas difusos, que permiten obtener realizaciones software y hardware de los mismos.

**Palabras Clave:** Lógica difusa, Herramientas de CAD, Aprendizaje.

## 1 INTRODUCCIÓN

La lógica difusa viene utilizándose con notable éxito desde hace algún tiempo en campos como el control, los sistemas de toma de decisiones, el reconocimiento de imágenes y el modelado de sistemas no lineales [7]. Ello se debe en gran medida a su capacidad para capturar en un formalismo "natural" el conocimiento que los seres humanos expresan a través del lenguaje, lo que proporciona los mecanismos necesarios para realizar una tarea determinada a partir de una descripción lingüística de la misma.

A pesar de que actualmente existe un considerable número de herramientas para el desarrollo de sistemas basados en lógica difusa, la mayor parte de estas herramientas contemplan un conjunto limitado de operadores difusos (funciones de implicación, conectivas difusas, métodos de defuzzificación, etc [5]). En este sentido el software disponible suele estar muy ligado a aplicaciones específicas que generan código para algunos de los microprocesadores o microcontroladores estándares.

Por otro lado, aunque estos programas permiten simular los sistemas a nivel de comportamiento, desde la perspectiva de la realización de sistemas difusos mediante hardware dedicado [3], no proporcionan enlaces eficientes hacia los entornos convencionales de diseño de circuitos integrados.

El entorno de diseño de sistemas difusos que describimos en esta comunicación intenta resolver las limitaciones anteriores. Para ello, en primer lugar, utiliza un método de especificación independiente de las operaciones concretas empleadas en cada caso y que no impone limitación alguna en cuanto a la complejidad del sistema especificado, tanto en la estructura de la base de regla, como en la definición de las propias reglas. Además, dicho método de especificación es lo suficientemente flexible y general como para que pueda ser empleado por las diferentes herramientas de verificación y síntesis que configuran el entorno. Por último, se trata de un entorno abierto, lo que permite la incorporación de nuevos formalismos y métodos de desarrollo y verificación, así como la inclusión de nuevos soportes de destino, software o hardware, para las implementaciones obtenidas con él.

## 2 EL ENTORNO XFUZZY 2.0

Xfuzzy 2.0 es un entorno de diseño que facilita la especificación y verificación de sistemas difusos, así como la implementación física de los mismos. Los módulos que integran Xfuzzy están basados en el lenguaje de especificación XFL [6]. La potencia y flexibilidad del lenguaje, junto con la independencia del modelo soportado por XFL con respecto a las decisiones de diseño, hacen de Xfuzzy un entorno con un amplio campo de aplicaciones, tanto para la evaluación de diferentes operadores y formalismos difusos, como para la síntesis software y hardware de sistemas basados en lógica difusa [1].

La figura 1 muestra la estructura general de Xfuzzy. El núcleo del entorno está constituido por un conjunto de funciones comunes que denominamos la *librería XFL* y que constituye el elemento de enlace entre los diferentes módulos. Los elementos de esta librería se encargan de hacer el aná-

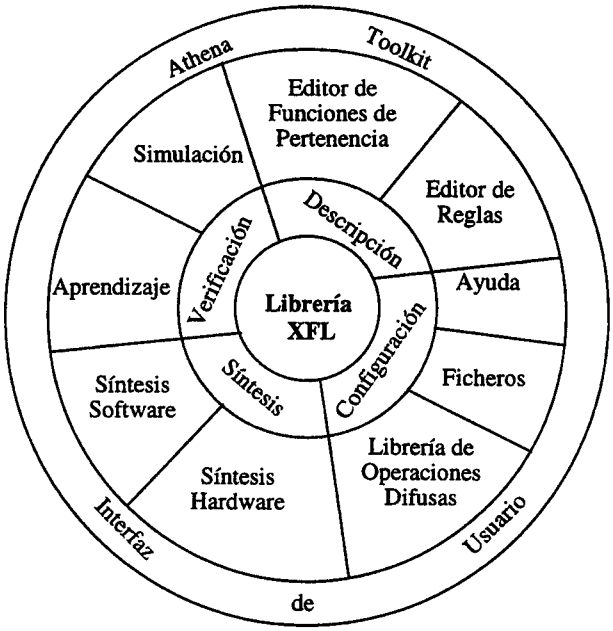



Fig. 1: Estructura del entorno Xfuzzy 2.0

lisis sintáctico y semántico de las especificaciones XFL y almacenarlo en forma de un árbol de sintaxis abstracta, empleado por el resto de elementos. Alrededor de este núcleo, y utilizando los servicios que ofrece, se disponen las utilidades o programas que conforman la herramienta y facilitan las sucesivas etapas de diseño: descripción, verificación y síntesis. Finalmente, el interfaz de usuario incluye un sistema de ventanas que permite acceder de manera simple e intuitiva a cada uno de los elementos del entorno. Está basado en X-Windows, utilizando el *toolkit* Athena-3D. La versión actual de Xfuzzy puede ejecutarse en cualquier sistema operativo compatible Unix y que disponga de X-Windows.

Las diferentes herramientas del entorno son accesibles desde la ventana principal de Xfuzzy (figura 2), que proporciona asimismo los mecanismos para configurar el entorno en sí, para manejar ficheros y para gestionar las definiciones de operaciones difusas. Estas últimas son completamente controladas por el usuario, aunque se incluye un amplio número de conectivas, funciones de implicación y métodos de defuzzificación ya predefinidos.

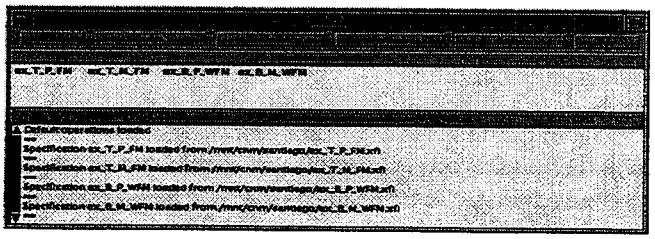


Fig. 2: Ventana principal de Xfuzzy 2.0.

La especificación XFL de un sistema difuso puede realizarse de forma textual, mediante el editor de textos elegido por el usuario, o mediante el uso de editores de funciones de pertenencia y de reglas que capturan de forma gráfica la especificación del sistema y generan la correspondiente descripción XFL. El resto de los elementos que componen Xfuzzy serán tratadas con mayor detalle en el apartado siguiente.

# 3 MÓDULOS QUE COMPONEN XFUZZY

## 3.1 XFC: COMPILADOR DE XFL A C

A partir de una especificación XFL genera un fichero que contiene la función en C que implementa el motor de inferencia difuso correspondiente a dicha especificación. Esta función puede ser empleada para producir cualquier programa para el que se desee utilizar las características de la lógica difusa. El único requerimiento que impone el empleo de uno de estos ficheros C es que el programa final debe ser compilado usando la librería estándar *libm.a* que contiene funciones matemáticas específicas.

El módulo que llama a la función que implementa el motor de inferencia se comunica con ella mediante los parámetros definidos en el bloque *system* de la descripción XFL. Los parámetros se pasan a la función por referencia, en el mismo orden en que han sido declarados en la descripción fuente XFL. Tras el retorno de la función, los valores de aquellos parámetros identificados como salidas en la descripción XFL son actualizados de acuerdo con sus definiciones.

## 3.2 XFSIM: SIMULADOR DE SISTEMAS DIFUSOS

El módulo de simulación permite la integración de uno o varios sistemas difusos especificados mediante XFL en un sistema de bucle cerrado que simule el comportamiento total del (o de los) sistemas difusos objeto de estudio, con el

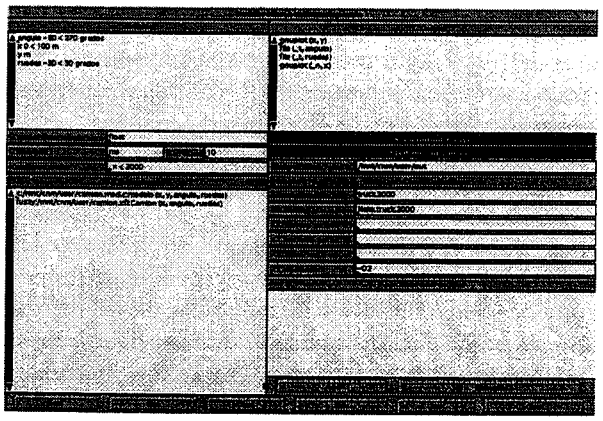


Fig. 3: Ventana del módulo de simulación

resto del entorno con el que deben interactuar. Recibe como entrada un fichero de definición del sistema con las siguientes especificaciones (figura 3): 1) las variables que definen el estado del sistema; 2) el comportamiento del sistema simulado, a través de la interacción de diferentes módulos definidos por medio de una combinación arbitraria de sistemas difusos basados en XFL, o bien mediante llamadas a procedimientos con un interfaz en lenguaje C o lecturas de datos almacenados en un fichero; 3) los resultados que se desea estudiar, especificados en forma de salidas de la simulación; y 4) una serie de parámetros que controlan la construcción y el comportamiento de la simulación.

A partir de esta definición, se construye un fichero fuente C, que contiene la función principal del programa de simulación y un *Makefile* para la construcción del mismo. El *Makefile* generado es un fichero fuente para el programa *make*, que permite reconstruir el ejecutable de la simulación en cualquier momento, teniendo en cuenta los cambios que se hayan producido en sus partes constituyentes. El *Makefile* utiliza el compilador de XFL a C, el compilador C seleccionado por el usuario y las librerías *libfrt.a* (que contienen las funciones de *runtime*) y *libm.a*.

La función principal de la simulación distingue tres fases: inicialización, bucle de simulación y generación de resultados. En la primera fase las variables definidas en el fichero de definición son inicializadas de acuerdo con los valores por defecto o por medio de los parámetros con los que se ha invocado la simulación.

Mientras no se cumpla la condición de final o no se produzca algún error, se ejecuta el bucle de simulación. Al principio de cada iteración se almacenan los valores de las variables de acuerdo con las especificaciones de salida en el fichero de definición. A continuación se invocan los procedimientos que definen el comportamiento del sistema. Estos procedimientos se traducen en: 1) llamadas a la función C que implementa el motor de inferencia (generado mediante el compilador de XFL a C), para las especificaciones XFL; 2) llamadas directas a la interfaz C, para los procedimientos así definidos; y 3) llamadas a unas funciones internas de *runtime* que leen los datos del fichero especificado, cuando se emplee esta clase de procedimiento. Todos los parámetros se pasan a estas funciones por referencia de manera que cualquiera de estos procedimientos puede leer y modificar el valor de cualquier variable del sistema.

En la tercera fase se procede a cerrar los ficheros de salida y a la ejecución (si así lo especifica el fichero de definición) de los procedimientos de visualización de los valores de salida. Actualmente Xfuzzy soporta dos formatos de salida: un formato en el que los datos se almacenan en líneas sucesivas de un fichero de texto y un formato compatible con el visualizador gráfico *gnuplot*.

### 3.3 XFBPA: MÓDULO DE APRENDIZAJE

Xfuzzy incorpora un módulo de ajuste automático para sistemas difusos basado en algoritmos de retropropagación de errores. La idea fundamental de este tipo de algoritmos es minimizar una función de error del sistema, haciéndola tender a un mínimo local. En el algoritmo de retropropagación convencional la variación del valor de cada parámetro en cada iteración viene dada por:

$$\Delta p = -\eta \cdot \frac{\partial E}{\partial p}$$

donde $\eta$ es la tasa de aprendizaje (*Learning Rate*) y E es la función de error del sistema. La función de error usada en Xfuzzy es:

$$E = \frac{1}{r} \cdot \sum_{i=1}^{r} \sum_{j=1}^{n} w_j \cdot \left( \frac{y_{ji} - Y_{ji}}{range_j} \right)^2$$

Xfuzzy soporta también otros tipos de aprendizaje referenciados en la literatura: retropopagación con momento, quick-prop [2], R-prop [8], el algoritmo manhattan y el algoritmo utilizado en ANFIS [4].

El módulo de aprendizaje permite personalizar la función de error, potenciando aquellas variables de salida que el usuario considere más importantes en la desviación del sistema (configurando los parámetros de la función de error). Asimismo, permite eliminar aquellas reglas que hayan quedado obsoletas tras el ajuste, entendiendo por obsoletas aquellas reglas que nunca se activan por encima de un determinado umbral, configurable por el usuario. Si tras el proceso de aprendizaje alguna regla se eliminara por este motivo, se lleva a cabo un nuevo ajuste, ya que la modificación de la base de reglas puede afectar al resultado del aprendizaje.

Xfbpa permite también incluir un fichero de aprendizaje específico para seleccionar las variables, funciones de pertenencia y parámetros que se desean ajustar. De esta manera es posible explorar distintas estrategias de aprendizaje para seleccionar la que mejor se adapte a un determinado problema.

El funcionamiento deseado del sistema en el proceso de ajuste se expresa por medio de un conjunto de datos de entrada-salida correctos. Estos datos se almacenan secuen-

cialmente en un fichero de entrenamiento, de manera que los datos se asignan a las variables en el mismo orden en que éstas se definen en el módulo *system* de la especificación XFL.

La herramienta requiere una condición de finalización del proceso de ajuste y ofrece cuatro posibilidades: límite (superior) de iteraciones, límite (inferior) de la desviación media, límite (inferior) de la desviación máxima y límite (inferior) de la variación relativa del error. El proceso de ajuste termina cuando se alcanza alguno de estos límites.

Cabe destacar que no existe ningún límite respecto al número de entradas y salidas del sistema ni respecto a la complejidad de la base de reglas. Los antecedentes de las reglas pueden contener cualquier combinación de términos, empleando las conectivas de conjunción, disyunción o negación. El aprendizaje puede aplicarse a sistemas que empleen un amplio número de conectivas difusas que pueden ser seleccionadas arbitrariamente por el usuario. Las derivadas para esas conectivas se calculan en tiempo de ejecución del programa. La herramienta soporta todas las clases de funciones de pertenencia definidas en XFL y el conjunto de T-normas, T-conormas y funciones de implicación cubre la mayoría de las funciones descritas en la literatura. Con respecto a los métodos de defuzzificación, el módulo de aprendizaje permite el uso de métodos que recorren el universo de discurso de las variables de salida (como el método del Centro de Área) y aquellos que calculan el resultado de la inferencia como una media ponderada de las conclusiones de cada regla (los llamados métodos simplificados como la Media Difusa).

La figura 4 muestra la apariencia de la ventana de aprendizaje. Se distinguen cuatro zonas con funcionalidad diferente. La zona superior izquierda permite la configuración de los parámetros del algoritmo de aprendizaje seleccionado, mientras que la superior derecha muestra información sobre el estado del proceso de aprendizaje. La zona central de la ventana provee información gráfica de las desviaciones media y máxima con respecto al número de iteraciones durante el bucle de aprendizaje. La zona inferior es una barra de comandos de control. Esta interfaz no sólo permite al usuario configurar y ejecutar el proceso de aprendizaje sino que también permite cambiar dinámicamente la configuración y monitorizar su evolución.

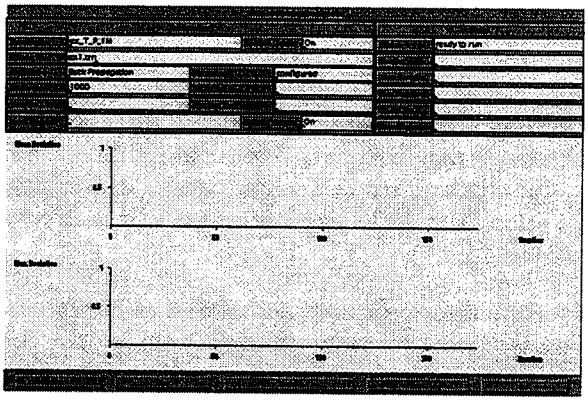


Fig. 4: Ventana de aprendizaje de Xfuzzy

### 3.4 MÓDULOS DE SÍNTESIS HARDWARE

Xfuzzy 2.0 incluye una serie de módulos para automatizar la realización hardware de sistemas difusos. En este caso se parte de una descripción XFL y se obtiene como salida una definición de la implementación del sistema en el formato sintetizable correspondiente. Xfuzzy ofrece módulos de síntesis para diferentes metodologías de diseño: implementaciones basadas en tablas a las que se aplica minimización lógica, implementaciones sobre FPGAs o implementaciones de ASICs. La descripción detallada de estas herramientas así como la arquitectura empleada en las dos últimas alternativas será objeto de una comunicación específica.

## 4 EJEMPLO DE DESARROLLO BASADO EN XFUZZY

Vamos a ilustrar en este apartado la utilización de Xfuzzy para desarrollar un sistema difuso que aproxime la función:

$$z=0.5(1+\sin(2\pi x)\cos(2\pi y)$$

El flujo de diseño mediante Xfuzzy parte de la especificación del sistema bajo desarrollo. Ello requiere describir las funciones de pertenencia que definen los conjuntos difusos de los antecedentes y consecuentes, y establecer la base de reglas del sistema. Asimismo, se deberá especificar las operaciones difusas que definen el mecanismo de inferencia empleado (conectivos de antecedentes y reglas, función de implicación y método de defuzzificación).

En el ejemplo que estamos considerando suponemos una distribución homogénea de los universos de discurso de la variables de entrada utilizando 7 variables lingüísticas representadas por funciones de pertenencia triangulares, y partimos de una descripción XFL que asocia un consecuente distinto (inicialmente representado por una función gausiana localizada en el valor intermedio 0.5) para cada posible regla.

La figura 5 muestra el resultado del ajuste. Como puede observarse, las funciones de pertenencia de los antecedentes apenas han sufrido modificaciones, mientras que las de los consecuentes pueden clasificarse en 5 grupos homogéneamente distribuidos a lo largo del universo de discurso de la variable de salida. Este procedimiento nos permite simpli-

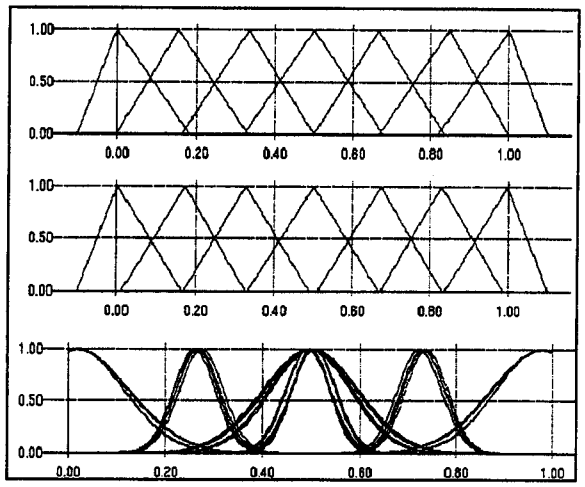



Fig. 5: Funciones de pertenencia tras el aprendizaje.

| y \ x | x1 | x2 | x3 | x4 | x5 | x6 | x7 |
|---|---|---|---|---|---|---|---|
| y1 | z3 | z5 | z5 | z3 | z1 | z1 | z3 |
| y2 | z3 | z4 | z4 | z3 | z2 | z2 | z3 |
| y3 | z3 | z2 | z2 | z3 | z4 | z4 | z3 |
| y4 | z3 | z1 | z1 | z3 | z5 | z5 | z3 |
| y5 | z3 | z2 | z2 | z3 | z4 | z4 | z3 |
| y6 | z3 | z4 | z4 | z3 | z2 | z2 | z3 |
| y7 | z3 | z5 | z5 | z3 | z1 | z1 | z3 |

Fig. 6: Base de reglas obtenida tras el aprendizaje.

ficar la base de reglas utilizando sólo 5 consecuentes distintos, como se muestra en la tabla de la figura 6.

La figura 7 muestra la descripción XFL resultante para el ejemplo considerado. La descripción del sistema difuso incluye en primer lugar las directivas que identifican los operadores difusos empleados: el tipo de conectiva (producto), la composición (máximo) y el método de defuzzificación a utilizar (media difusa ponderada). A continuación, en las definiciones de tipos se especifican las funciones de pertenencia de los antecedentes y consecuentes. También se especifican los rangos de valores de los universos de discurso. Por otro lado, el módulo *system* define la base de reglas. XFL permite describir sistemas con una estructura arbitraria. Nuestro ejemplo corresponde a un sistema simple que contiene un único módulo con su base de reglas. Las reglas tienen el formato "if ... then ..." y pueden construirse haciendo uso de conectivos *and* (&) y *or* (|) para simplificar la descripción.

```
#and product
#composition max
#defuzzification WeightedFuzzyMean

type tx : real [101] ( 0 < 1 ) {
  x1 triangle (-0.100, 0.000, 0.166)
  x2 triangle (0.000, 0.166, 0.333)
  x3 triangle (0.166, 0.333, 0.500)
  x4 triangle (0.333, 0.500, 0.666)
  x5 triangle (0.500, 0.666, 0.833)
  x6 triangle (0.666, 0.833, 1.000)
  x7 triangle (0.833, 1.000, 1.100) }
type ty : real [101] ( 0< 1) {
  y1 triangle (-0.100, 0.000, 0.166)
  y2 triangle (0.000, 0.166, 0.333)
  y3 triangle (0.166, 0.333, 0.500)
  y4 triangle (0.333, 0.500, 0.666)
  y5 triangle (0.500, 0.666, 0.833)
  y6 triangle (0.666, 0.833, 1.000)
  y7 triangle (0.833, 1.000, 1.100) }
type tz : real [101] ( 0 < 1 ) {
  z1 bell (0.0, 0.1)
  z2 bell (0.1, 0.1)
  z3 bell (0.5, 0.1)
  z4 bell (0.9, 0.1)
  z5 bell (1.0, 0.1)      }
```

```
system (tx ? x, ty ? y, tz ! z)
 rulebase {

  if (x is x2 & y is y4)                  |
    (x is x3 & y is y4)                   |
    (x is x5 & (y is y1 | y is y7))       |
    (x is x6 & (y is y1 | y is y7))           -> z is z1

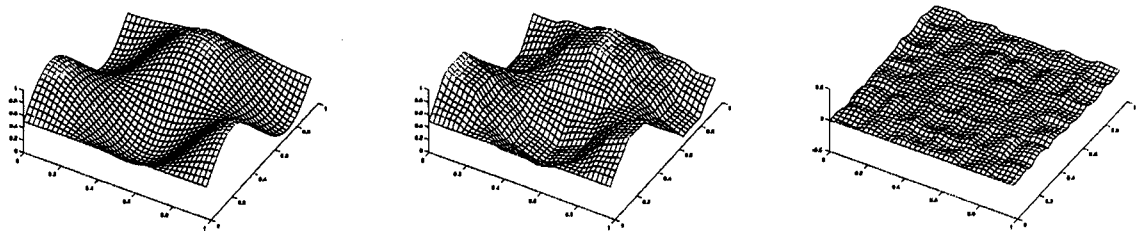
  if (x is x2 & (y is y3 | y is y5))      |
    (x is x3 & (y is y3 | y is y5))       |
    (x is x5 & (y is y2 | y is y6))       |
    (x is x6 & (y is y2 | y is y6))           -> z is z2

  if x is x1 | x is x4 | x is x7                -> z is z3

  if (x is x2 & (y is y2 | y is y6))      |
    (x is x3 & (y is y2 | y is y6))
    (x is x5 & (y is y3 | y is y5))
    (x is x6 & (y is y3 | y is y5))           -> z is z4

  if (x is x2 & (y is y1 | y is y7))
    (x is x3 & (y is y1 | y is y7))
    (x is x5 & y is y4)
    (x is x6 & y is y4)                       -> z is z5 }
```

Fig. 7: Ejemplo de descripción XFL.



(a) (b) (c)

Fig. 8: Resultados experimentales: a) función objetivo. b) función aproximada. c) superficie de error.

Una vez identificado el sistema es posible aplicar de nuevo la herramienta de aprendizaje para refinar su comportamiento. En el ejemplo que estamos describiendo se ha obtenido un error cuadrático medio del 0.76%, siendo la desviación máxima del 2.57%

El comportamiento del sistema puede ser verificado en cada iteración del proceso de diseño utilizando el módulo de simulación de Xfuzzy. A modo de ejemplo, la figura 8 muestra algunos de los resultados obtenidos. La gráfica de la figura 8a muestra la superficie de la función objetivo, mientras que las gráficas 8b y 8c muestran, respectivamente, la superficie aproximada por el sistema difuso y la superficie de error.

## 5 CONSIDERACIONES FINALES

Xfuzzy 2.0 es un entorno abierto pensado para ayudar al desarrollo de herramientas CAD para sistemas que operan en lógica difusa. Nuestro objetivo es que la versión actual sea el núcleo al que puedan incorporarse nuevos módulos que mejoren y complementen la funcionalidad del entorno.

Se trata de una herramienta de libre distribución. Con objeto de que su uso sea lo más extendido posible se ha previsto que Xfuzzy pueda ser ejecutado desde distintas plataformas Unix. Así el kit de distribución contiene distintos ficheros *Makefile* que permiten construir la aplicación en los siguientes sistemas operativos: Linux, IRIX, OSF1, Solaris, SunOS y Ultrix. Por supuesto se incluyen los ficheros fuente. Esto permite que el usuario pueda modificar y adaptar las aplicaciones a sus necesidades y añadir nuevos elementos de programa.

## Referencias


[1] A. Barriga, S. Sánchez-Solano, C. J. Jiménez, D. Galán, D. R. López, "Automatic Synthesis of Fuzzy Logic Controllers", *Mathware & Soft Computing*, vol. III, n. 3, pp. 425-434, Sep. 1996.

[2] S. E. Fahlman, "An Empirical Study of Learning Speed in Back-Propagation Networks", *CMU-CS-88-162*, Sept. 1988.

[3] D. L. Hung. "Dedicated Digital Fuzzy Hardware". *IEEE Micro*, vol. 15, n. 4, pp. 31-39, Aug. 1995.

[4] J. S. Jang, "ANFIS: Adaptative-network-based fuzzy inference systems", *IEEE Trans. on Systems, Man and Cybernetics*, 1992.

[5] C. C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller", Parts I & II, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20 n. 2, pp. 404-432, 1990.

[6] D. R. López, S. Sánchez-Solano, A. Barriga, "XFL: Un Lenguaje para la Definición de Sistemas Difusos", *VI Congreso Español Sobre Tecnologías y Lógica Fuzzy*, pp. 167-172, Oviedo, Sep. 1996.

[7] T. Munakata, Y. Jani, "Fuzzy Systems: An Overview", *Communications of the ACM*, vol. 37, n. 3, pp. 69-76, Mar. 1994.

[8] M. Riedmiller, H. Braun, "RPROP - A Fast Adaptative Learning Algorithm", *Proc. of ISCIS VII*, 1992.

# ENTORNO MULTICONTEXTO Y METACONTROL EN MICROPROCESADORES BORROSOS

**Pedro E. Bernad Silva**
Laboratorio de Hardware y
Control Avanzado. Instituto
Nacional de Técnica
Aeroespacial-INTA-28850-
Torrejón de Ardoz -Madrid-
SPAIN
bernarsp@inta.es

## Resumen

Este documento expone algunas ideas para optimizar el entorno de un microprocesador borroso. Estas modificaciones tiene por objeto incrementar la versatilidad y simplificar la programación del microprocesador. Hablaremos sobre múltiples contextos y control del hardware mediante metarreglas. Algunas de estas técnicas pueden encontrarse tímidamente en implementaciones comerciales de procesadores borrosos, pero están todavía por desarrollar. Hay otros dispositivos que pueden ser útiles dentro de un microprocesador borroso: derivadores, integradores, convertidores de escala, multiplexores de entradas, etcétera.

**Palabras Clave:** Entorno de microprocesador borroso, contexto, metarregla.

## 1 INTRODUCCION

Un microprocesador borroso puede dividirse en dos partes [12]: Núcleo de Procesamiento Borroso y Entorno de Procesamiento Borroso.
El Núcleo de Procesamiento Borroso es la parte del circuito capaz de procesar señales mediante un sistema basado en reglas y generar conclusiones, borrosas o no, implementando técnicas borrosas. Por otro lado, el Entorno de Procesamiento Borroso son todos los circuitos que ayudan al Núcleo de Procesamiento Borroso. Centraremos nuestro estudio en la parte del Entorno, con el objeto de lograr procesadores más versátiles y simplificar la programación.
La optimización del entorno de procesamiento borroso es el camino más fácil para acercar los dispositivos borrosos a la versatilidad de los procesadores nítidos, lo que constituye un primer paso para la programación borrosa. Este primer paso debe perseguir la realización de estructuras de programación complejas con modularidad y control de flujo en el propio código.
En este artículo veremos algunos dispositivos que pueden ser incluidos en futuras arquitecturas con este propósito, todos ellos con sus ventajas y desventajas. Estudiaremos un entorno multicontexto y un modelo de autogestión basado en metarreglas.

## 2 ENTORNO MULTICONTEXTO

### 2.1 DEFINICIONES

Podemos definir un contexto de borrosificación o deborrosificación como una distribución de los conjuntos borrosos en universo de discurso antecedente y consecuente, respectivamente. Definiremos un contexto de reglas como un conjunto de reglas que pueden inferirse en un estado del problema o en un momento del tiempo. Hay algunos ejemplos de microprocesadores con múltiples contextos de reglas como [16] y [7]. Un cambio de contexto se puede definir como un cambio en la base de reglas y posiblemente en la distribución de los conjuntos sobre el universo de discurso [12]. Un microprocesador borroso multicontexto es un procesador que tiene precargados un conjunto de contextos de borrosificación, deborrosificación y reglas, y puede cambiar de unos a otros sin detener el flujo de ejecución. Los contextos que no están activos pueden ser reemplazados en tiempo de ejecución sin detener el flujo normal del procesador. No debe confundirse con un sistema de aprendizaje on-line donde los contextos se

adaptan progresivamente según se recogen resultados. En un entorno multicontexto puede haber o no aprendizaje.

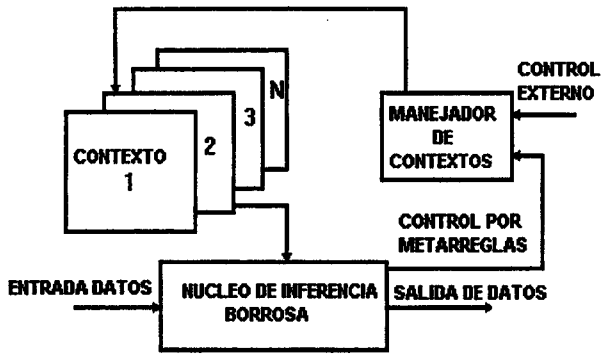



Fig1: Entorno multicontexto

## 2.2 CONTEXTO DE REGLAS

Si tenemos una regla como:

$$\textbf{IF } X_1=A_1 \wedge X_2=A_2 \wedge X_3=A_3 \wedge \ldots \wedge X_n=A_n \textbf{ THEN } Y_1=B_1 \quad (1)$$

En muchas ocasiones podemos distinguir en ella tres informaciones:

**IF informacion_contextual ∧ información_eventual THEN información_acción** (2)

**Información contextual**: Relativa al contexto donde se ejecuta una regla. Contiene las condiciones que caracterizan a dicho contexto. Pueden ser informaciones temporales, espaciales o sobre el estado del sistema. En algunos casos podemos expresar estas condiciones como conjunción de valores antecedentes, pero en otros puede tratarse de información subjetiva o independiente de las entradas del sistema. En un sistema de control la información contextual cambia más despacio que la información eventual.

**Información eventual**: Es la condición exacta o puntual que dispara una regla. En un sistema de control la información eventual cambia instantáneamente.

**Información de acción**. Esta es la información almacenada en la parte consecuente de la regla. Es la consecuencia de la conjunción de la información contextual y la información eventual.

En algunas ocasiones la información contextual y eventual poseen la misma naturaleza y las diferencias solamente pueden justificarse por la frecuencia de los cambios en el tiempo, que es muy superior en la eventual. Sin embargo la información contextual puede ser también no modelizable o subjetiva.

Para cada información contextual hay muchas eventuales, y la misma información eventual puede disparar diferentes acciones dependiendo de la información contextual. Sin embargo, dada una información contextual, todas la reglas que no la contienen no tienen efecto en el proceso de inferencia [13].

La idea de contexto de reglas está en agrupar todas las reglas que poseen la misma información contextual en un subconjunto de la base de reglas, y cuando se detecte dicha información , emplear únicamente este subconjunto. ([7] lo denomina el subconjunto de reglas activas).

Podemos almacenar un contexto como:

**información_contextual** (3)

y una regla en un contexto como:

**IF información_eventual THEN información_acción** (4)

La ventaja de esta aproximación es que el conjunto de reglas es más pequeño que la base de reglas primitiva, y el bucle de inferencias es más corto en un contexto que en toda la base (un gran número de reglas borrosas implica un incremento significativo en el tiempo de proceso global [2][3]). Otra ventaja es la inclusión de información subjetiva o no modelizable mediante reglas dentro del sistema de razonamiento. Sin embargo esta información deberá modelizarse de alguna otra manera.

## 2.3 CAMBIO DE CONTEXTO

Cuando detectamos un cambio en la información contextual en un problema programaremos un cambio de contexto hardware. Si la información contextual contiene información subjetiva o no modelizable, el cambio se detectará fuera del microprocesador borroso, y se aplicará mediante una señal externa. Si la información contextual puede ser expresada como función de los valores de entrada, el cambio de contexto puede llevarse a cabo usando metarreglas como se describe en 3.

## 2.4 VENTAJAS DE LA IMPLEMENTACION DE CONTEXTOS

La razón principal para tener varios contextos de borrosificación y deborrosificación en un microprocesador es que nos permite definir exactamente las etiquetas de las variables antecedentes y consecuentes para cada estado del problema. Al igual que en el pensamiento humano no se ven ni se actúa ante los mismos eventos bajo diferentes condiciones ambientales, un procesador borroso no tiene que actuar igual frente a los mismos eventos en diferentes contextos. Usando un entorno multicontexto de borrosificación y deborrosificación no es preciso generalizar sistemas borrosos para adaptarlos a múltiples circunstancias.

Sin embargo, un entorno con múltiples contextos de reglas puede presentar varios problemas. Imaginemos que estamos en una situación donde las condiciones contextuales cambian muchas veces en el tiempo. En un entorno multicontexto este problema implicaría continuos

cambios de contexto para tratar los eventos. Este tiempo puede ser excesivo en múltiples sistemas de control. Por otro lado un sistema multicontexto no puede reemplazar el mecanismo de interrupciones. El tiempo de cambio de contexto debe tenerse en cuenta para calcular el tiempo de respuesta. La programación multicontexto debe realizarse cuidadosamente.

Otro problema con el que podemos enfrentarnos es que los microprocesadores borrosos tienen registros internos para controlar el flujo de datos entre estados consecutivos, y un simple cambio de contexto puede acarrear inconsistencias en las salidas durante los primeros ciclos, que no son sino retrasos en la aplicación de nuevos contextos. Puede ser necesaria una estrategia de cambio, como cerrar las entradas durante la transición, cerrar las salidas, si no es un sistema crítico en tiempo (nótese que estas estrategias contradicen al principio de la no detención del procesador durante el cambio). El problema se agudiza para arquitecturas sistólicas y en pipeline [9][1][12]. En estos casos será preciso evaluar el coste de un entorno multicontexto en la eficiencia del procesador.

### 2.5 IMPLEMENTACION

Si queremos incluir un entorno multicontexto, necesitamos dispositivos de almacenamiento para guardar los contextos, y un manejador para pasar de uno a otro. El almacenamiento puede llevarse a cabo en memorias [14][1] o en registros de desplazamiento [12]. En el primer caso, el almacenamiento puede ser dentro del procesador [1], o en un dispositivo externo [14]. Un dispositivo externo siempre es más flexible, pero el acceso es más lento, y esto puede suponer una degradación de las prestaciones del procesador.

El acceso a un dispositivo interno puede ser más rápido, pero el espacio necesario para almacenar las funciones de pertenencia de los contextos de borrosificación de deborrosificación debe ser considerado. El almacenamiento de los contextos de reglas debe prever tanto el acceso (serie o paralelo) como la geometría (reglas ortogonales de igual longitud o no ortogonales). La organización del dispositivo de almacenamiento dependerá del formato de las reglas, del formato de los conjuntos borrosos [10], y de las posibilidades de manejo.

El manejador debe ser capaz de acceder y cambiar los contextos rápidamente. Un entorno multicontexto puede permitir que un contexto se esté actualizando, mediante aprendizaje o desde otro dispositivo de almacenamiento mientras otro se está ejecutando. Por ejemplo, una RAM de doble puerta [Ike 94] permite realizar simultáneamente operaciones de lectura y escritura. En este caso es importante preservar la consistencia. Un manejador multicontexto sencillo puede permitir solamente cambiar entre dos contextos de reglas, con un contexto de borrosificación y deborrosificación asociado a cada uno. Pueden ser útiles modelos más complejos. Un manejador que permita usar libremente todas las combinaciones de contextos de borrosificación, deborrosificación y reglas puede ser muy interesante en un microprocesador borroso de propósito general. En un sistema de múltiples contextos almacenados en memoria, el manejador puede ser solamente un conjunto de tres registros con tres direcciones: dirección del contexto de borrosificación en uso, dirección del contexto de deborrosificación en uso, y dirección del contexto de reglas en uso. Este direccionamiento indirecto ofrece mucha flexibilidad, pero su lentitud lo hace inapropiado para muchos sistemas de control.

### 2.6 APLICACIONES

Hay muchos ejemplos de aplicaciones que pueden incluir más de un contexto en un procesador. En primer lugar, cuando un sistema puede atravesar varios estados a lo largo del tiempo. En este caso necesitamos un modelo complejo de reglas, o un mecanismo para cambiar qué reglas van a evaluarse en cada momento. También necesitaremos cambiar los conjuntos sobre los universos de discurso. Otro ejemplo es cuando tenemos un control de baja resolución para manejar un sistema. (por ejemplo, el piloto automático de un avión), y donde se necesite más detalle (cerca del aeropuerto), cambiar a alta resolución. El tercer ejemplo en un entorno multitarea, donde un único procesador ejecuta varios trabajos completamente distintos. En este caso necesitamos un mecanismo para cambiar también las variables de entradas y de salida, resultando un cambio de contexto completo.

## 3 AUTOORGANIZACION: METARREGLAS

### 3.1 DEFINICIONES

Un controlador borroso autogestionado (SOC) es un tipo de controlador basado en reglas organizadas en dos niveles. Es capaz de modificar las reglas existentes y crear otras nuevas, con el objeto de mejorar el proceso de respuesta [6]. En el nivel inferior encontramos las reglas borrosas tradicionales, operando sobre el sistema a controlar, mientras que el segundo nivel de reglas (metarreglas) está formado por reglas borrosas que manejan el propio controlador. En este documento no veremos la manera de implementar técnicas de aprendizaje automático sobre bases de reglas. Lo que queremos es controlar los dispositivos del sistema así como los contextos (véase 2) mediante metacontrol, obteniendo programas autosuficientes.

Las metarreglas son una manera de mezclar tareas de metacontrol con tareas de inferencia. El metanivel de razonamiento y el metanivel de inferencia (inferencia realizada en un nivel que se encuentra sobre otro nivel) [4] son conceptos que aparecieron hace mucho tiempo en

sistemas expertos. Una metarregla borrosa puede definirse como un regla borrosa donde las variables antecedentes son entradas del sistema, y las variables consecuentes (o subconjuntos de éstas) son acciones para controlar los dispositivos.

### 3.2 VENTAJAS

Hay que manejar muchas cosas en un microprocesador borroso: selección de variables antecedentes y consecuentes, activación de contextos de borrosificación, deborrosificación y reglas, filtros externos de entradas y salidas, etcétera. Todos estos sistemas precisan valores nítidos para su control: una señal on/off, un número de contexto seleccionado, la habilitación de un modo de interrupciones, u otros. Podemos tener una unidad de control por señales externas, donde hay entradas y salidas de control en el chip. En estos casos, el proceso de inferencia y el proceso de control son independientes. Esta independencia puede entenderse como ventaja, pero la idea de disponer de una programación conjunta que cubra ambos aspectos como sucede en programación clásica resulta interesante. Si necesitamos hacer cambios en la configuración del sistema en tiempo de ejecución nos gustaría poder describirlos a la vez que se describe el proceso de los datos. Sería posible cargar un conjunto completo de reglas en un sistema de control borroso y ejecutarlas como un programa en un ordenador personal. Sin embargo la inferencia de las metarreglas exige un tiempo, y su almacenamiento un espacio. Asimismo para su tratamiento necesitaremos circuitería adicional, y puede que una etapa interna de filtrado para distinguir reglas de metarreglas. Por esta razón la gestión por metarreglas no es apta para todos los microprocesadores borrosos, y en muchos casos tendrá que convivir con el control externo. Un sistema crítico en el tiempo no se puede implementar usando un control por metarreglas.

### 3.3 IMPLEMENTACION

Los consecuentes de las metarreglas son normalmente crisp, porque las acciones de control no suelen tener niveles borrosos. Por esta razón no tiene sentido deborrosificar etiquetas usando media ponderada u otros procedimientos similares. Si en un mismo contexto tenemos varias reglas que actúan sobre un mismo metaconsecuente se puede tomar como conclusión el máximo de los valores de las etiquetas.

Hay algunos problemas en un procesador borroso controlado por metarreglas. En primer lugar, en los sistemas actuales el número de variables antecedentes y sobre todo consecuentes está muy limitado. Si hay tareas de control que precisan conocer el estado del sistema en el momento actual, éste debe aparecer como función de las variables de entrada o como contextos. Si la tarea de control se puede expresar como función lógica de las variables consecuentes, no se necesitan otras nuevas.

En otro caso, necesitaremos variables especiales para manejar los dispositivos o al menos algunas etiquetas de una variable consecuente. Se debe evitar que estas etiquetas sean consideradas en la deborrosificación, por ejemplo mediante en un simple bit que indique si una variable debe deborrosificarse completa o no. De esta manera es posible implementar metarreglas sobre un sistema que no tenga sin restarle potencia. Las etiquetas no señaladas (o variables no marcadas) se deborrosificarán empleando un procedimiento clásico, y las señaladas lo harán mediante el máximo. Un filtro a la salida se activará cuando se opere con metarreglas interpretando éstas y poniendo en funcionamiento los dispositivos asociados.

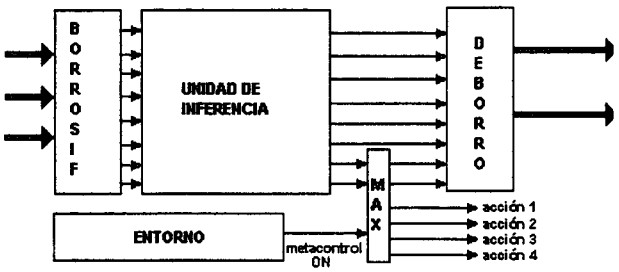



Figura2: posible implementación de manejador de metarreglas

El segundo problema es que la mayoría de los sistemas borrosos no tienen las reglas ordenadas en su dispositivo de almacenamiento. Podemos encontrar procesadores borrosos paralelos donde varias reglas se infieren al mismo tiempo [15]. Esta concurrencia no es válida para algunas tareas de control que deben ejecutarse secuencialmente. El problema puede resolverse usando un entorno multicontexto, donde dos metarreglas ordenadas deben estar en contextos diferentes, un tercera metarregla efectúa la transición. Otra solución puede ser una realimentación del estado del sistema como variable de entrada nítida. La mezcla de un sistema de control por metarreglas con un entorno multicontexto puede ser muy potente. Podemos tener metarreglas que activan diferentes contextos, y cambian de unos a otros, así como varios contextos para ofrecer una iteración a las metarreglas. En un futuro será posible desarrollar programas complejos que comiencen por una selección de entradas salidas y contextos, y atraviesen múltiples estados.

Hay un tercer problema. En muchos procesadores borrosos (especialmente en procesadores sistólicos borrosos [9]) todos las variables antecedentes están presentes en todas las reglas [11]. Es lo que se conoce por ortogonalidad de las reglas borrosas. Si tenemos una regla que no usa todas las variables antecedentes, el sistema

ejecutará una regla o conjunto de reglas conteniendo todos los valores posibles de las variables no usadas. Esto no es correcto en lógica borrosa [5], y puede ser un problema si el metacontrol solo depende de un subconjunto de las variables de entrada. En este caso se necesitará más de una regla para llevar a cabo una acción de control, lo que resulta caro en tiempo y espacio.
Una interrupción es una señal hardware que dispara un evento [8]. Las excepciones son eventos que interrumpen el flujo normal de ejecución en un procesador necesitando un cambio de modo de operación o una respuesta inmediata. Pueden ser generadas por el microprocesador o fuera de él. Algunas interrupciones son críticas en el tiempo, y no pueden incluirse en programas. Pero hay muchas señales de control que sí pueden esperar algunos ciclos para ser atendidas. El manejo de estas señales puede incluirse en un proceso global de reglas.

### 3.4 APLICACIONES

Hay muchos usos para el metacontrol usando metarreglas. Primero, un manejador de contextos podría controlarse mediante un sistema de metarreglas. Cuando las condiciones especiales de un problema recomiendan un cambio de estrategia, una metarregla activa un cambio de contexto.
Otro ejemplo se presenta ser cuando una variable antecedente comienza a mostrar valores extraños, sugiriendo que debe ser ignorada o filtrada. Una metarregla detecta el problema y cierra la entrada. En una tarjeta de control hay muchos circuitos que requieren una configuración. Podemos emplear un procesador o unidad de control nítida manejándolos, o un sistema de control por metarreglas, necesitando solamente el microprocesador borroso.

## 4 OTROS DISPOSITIVOS EN EL ENTORNO DE PROCESAMIENTO BORROSO

Hay otros dispositivos que pueden ser útiles en el entorno de un procesador borroso. En primer lugar, multiplexores de entradas y salidas. Un microprocesador borroso multipropósito precisa varias entradas y salidas diferentes cuando lleva a cabo diferentes tareas. Si las entradas y las salidas son fijas no podemos seleccionar cuales vamos a utilizar en cada problema. Hace falta un mecanismo de selección, que puede ser externo o interno. Un pequeño multiplexor interno (dentro del chip) permite seleccionar las entradas deseadas. El multiplexor se controlará mediante señales externas o con un control por metarreglas como se describe en 3.
Otros dispositivos que aplicables al entorno de un procesador son integradores y derivadores de entradas y salidas. En algunos problemas (control PID) es necesaria una entrada, su derivada y su valor integrado (es decir, su tendencia futura, su valor presente y su historia pasada). Situando estos dispositivos en el entorno simplificamos los circuitos de apoyo. El control se lleva a cabo también por metarreglas o mediante señales externas. Además de estos circuitos pueden ser interesantes para un microprocesador borroso de propósito general los convertidores analógico-digitales, registros, sistemas de realimentación para datos, convertidores de escala, etcétera.

## 5 CONCLUSIONES

Las conclusiones de este artículo son que un entorno multicontexto y el control por metarreglas son interesantes para optimizar los microprocesadores borrosos. Las ventajas de un entorno multicontexto son un mejor uso del espacio de almacenamiento, mayor versatilidad, y una simplificación del trabajo de programación.
Los juegos de reglas están más estructurados, y permiten una secuencialidad en acciones de control. Los inconvenientes son la necesidad de un hardware adicional para manejo de contextos, y la respuesta en el tiempo que se deteriora si el contexto activo está cambiando continuamente.
Un microprocesador borroso controlado por metarreglas permite programar simultáneamente control y datos con conjunto de reglas. El procesador borroso puede ser el maestro en un sistema de control sin un controlador nítido. Las interrupciones se atienden fuera del flujo de reglas porque precisan atención inmediata. Un control por metarreglas no siempre se puede implementar, porque es muy dependiente de la arquitectura del hardware.
La combinación de un entorno multicontexto con un control por metarreglas, resulta muy potente en microprocesadores borrosos, pues permite atacar problemas con varios estados y configuraciones desde un solo flujo de reglas.
En el futuro habrá dos líneas diferentes en el desarrollo de microprocesadores borrosos: procesadores de múltiples propósitos, donde las principales características sean la versatilidad y la facilidad de programación, y los procesadores de propósito específico, menos flexibles, pero de altas prestaciones.

**Agradecimientos**

Este trabajo ha sido realizado dentro del marco del proyecto Procesador Borroso de Altas Prestaciones, financiado por la CICYT con el TIC 96-1393-C06.

**Referencias**

[1] Ansgar P.(1993) Architecture of a PDM VLSI fuzzy logic controller with pipelining and optimized chip area *Second IEEE International Conference on Fuzzy Systems.* Vol I.

[2] Francesco Boschetti, Alessandro Gabrielli, Enzo Gandolfi, Massimo Masetti..(1995) Fuzzy Logic oriented to Active Rule selector and Membership Function Generator for High Speed Digital Fuzzy μ-Processor. *World Congress on Neural Network 1995. Washington* Vol 3. Pag 93-97.

[3] J.L. Castro, J.M. Zurita.(1992). Controlador Difuso Basado En Contextos. *II Congreso Español sobre Tecnologías y Lógica Fuzzy.*

[4] Z. Chen.(1996). Using meta-rules for fuzzy inference control. *Fuzzy Sets and Systems 79.* Pag 163-173.

[5] D. Dubois, H. Prade.(1980). *Fuzzy Sets and Systems. Theory and Applications. Academic Press Inc.*

[6] Ricardo Gutierres, Ricardo Tanscheit. (1993). Performance of a Fuzzy Self-Organising Controller regarding Scaling Factors, Linguistic Variables and quantization Levels of the Variables.*EUFIT'93.*

[7] H. Ikeda , Y Hiramoto y N. Kisu.(1991). A fuzzy inference processor with an 'active_rule_driven' architecture. *Proc. of Symposium on VLSI circuits.* Pags 25-26.

[8] Phillip A. Laplante.(1992). Real-Time Systems. Design and analysis. *IEEE Press.*

[9] Mahmoud A. Manzoul, S. Tayal.(1990) Sistolic VLSI array of multi-variable fuzzy control systems. *Cybernetics and Systems: An International Journal 21.* Pags 27-42.

[10] A. Ruiz, J. Gutiérrez (1991). Sistema de codificación de datos fuzzy para su eficaz procesamiento en hardware. *Informe Técnico dentro del Proyecto Orbe*, Ministerio de Industria.

[11] L. de Salvador, M. García, J. Gutiérrez (1994). Systolic architectures for fuzzy processing and their simulation. *Mathware and Soft Computing*, n° 3.

[12] L. de Salvador (1996). Modelo de un Microprocesador Borroso de Altas Prestaciones y su Diseño. *Tesis doctoral. Facultad de Informática de Madrid.*

[13] S. Sanchez Solano, C. J. Jimenez, A. Barriga, J.L.Huertas. Realizaciones hardware de sistemas basados en logica difusa.

[14] SGS Thompson Microelectronics.(1995) WARP 1.1 *Datasheet.*

[15] Hiroyuki Watanabe, Wayned Dettloff, Katty E. Yount. (1990). A VLSI fuzzy logic controller with reconfigurable, cascadable architecture. *IEEE Journal of solid state circuits Vol 25 N°2.*

[16] Jianwei Zhang , Jörg Raczowsky y Michael Webler.(1994). VLSI-design, specification and simulation of a fuzzy co-processor. *EUFIT'94.*

# XFUZZY 2.0: HERRAMIENTAS DE SÍNTESIS DE CONTROLADORES DIFUSOS

**C. J. Jiménez*, D. López**, E. Lago*, M. A. Hinojosa*, A. Barriga*, S. Sánchez-Solano***

* Instituto de Microelectrónica de Sevilla - Centro Nacional de Microelectrónica.
** Centro Informático Científico de Andalucía
Edificio CICA, Avda. Reina Mercedes s/n, 41012-Sevilla.
email: cjesus@imse.cnm.es

## Resumen

En esta comunicación se describen las herramientas de síntesis de controladores difusos incluidas en Xfuzzy 2.0, un entorno de descripción, verificación y síntesis de sistemas basados en lógica difusa. Estas herramientas permiten dos estrategias de diseño (tabla *look-up* o arquitectura específica del controlador) y dos técnicas de implementación (FPGAs o ASICs). Se evalúan las herramientas mediante ejemplos de diseños que nos permiten extraer conclusiones sobre la aplicabilidad de cada solución.

**Palabras Clave:** Controladores Difusos, Herramientas de CAD, Síntesis Automática.

## 1 INTRODUCCIÓN

En los últimos años se ha incrementado considerablemente el número de aplicaciones que utilizan técnicas de lógica difusa para resolver problemas de control o toma de decisiones y, como consecuencia, se han reportado múltiples soluciones para implementar algoritmos difusos mediante software y hardware [8]. La aplicabilidad de estas soluciones depende tanto de la complejidad del problema como de sus restricciones temporales. Las soluciones software proporcionan una gran flexibilidad para definir la base de conocimientos, seleccionar los operadores difusos y escoger los algoritmos de inferencia. Sin embargo, aquellos problemas que demandan una alta velocidad de operación requieren la adopción de soluciones hardware.

Las realizaciones hardware de controladores difusos pueden estar basadas en dos estrategias genéricas. En la estrategia *off-line* las salidas del controlador son precalculadas para todas las posibles combinaciones de las entradas. El proceso de inferencia se lleva a cabo mediante una tabla de búsqueda implementada en una RAM o mediante circuitos combinacionales [4],[6]. Si bien esta estrategia permite una flexibilidad completa en la definición del controlador difuso, su mayor inconveniente es el crecimiento exponencial de la memoria requerida cuando el número de entradas o el número de elementos en los universos de discurso aumenta. Para eliminar este problema la mayoría de las realizaciones de controladores difusos siguen una estrategia *on-line*, donde un hardware dedicado realiza la inferencia concurrentemente con los cambios en las entradas [2].

Otro aspecto importante relacionado con la realización hardware de controladores difusos es la técnica de implementación seleccionada. La implementación mediante FPGAs (*Field Programmable Gate Arrays*) permite prototipar rápidamente sistemas que más tarde pueden ser realizados como circuitos integrados. Además, los sistemas construidos con FPGAs exhiben una programabilidad intrínseca, proporcionando así un mecanismo simple para cambiar o adaptar su función durante la etapa de desarrollo. Por el contrario, para aplicaciones que requieren la fabricación de un elevado número de componentes, la implementación mediante ASICs (*Application Specific Integrated Circuits*) es más eficiente en términos de área y velocidad.

En esta comunicación se presentan las herramientas de síntesis incluidas en el entorno Xfuzzy 2.0, que permiten llevar a silicio las especificaciones de sistemas de control difuso. De acuerdo con las consideraciones anteriores se han analizado dos estrategias de realización: basada en tabla y basada en hardware específico, y dos técnicas de implementación: FPGAs y ASICs. La aplicabilidad de las diferentes soluciones es analizada al explorar el espacio de diseño mediante algunos ejemplos.

## 2 FLUJO DE DISEÑO DE CONTROLADORES DIFUSOS

El flujo de diseño de controladores difusos utilizando el entorno Xfuzzy se ilustra en la figura 1. El punto de partida del proceso de síntesis es una descripción del comportamiento del controlador basada en el lenguaje de especificación XFL [5]. La descripción del controlador contiene la

base de conocimientos inicial (funciones de pertenencia de los antecedentes y consecuentes y la base de reglas), así como el mecanismo de inferencia y el método de defuzzificación que se desea implementar. Dicha especificación puede simularse y ajustarse dentro del entorno Xfuzzy [1]. El proceso de ajuste se realiza con la ayuda de una herramienta de aprendizaje basada en algoritmos de retropropagación de errores. Esta herramienta nos permite identificar la base de reglas del sistema y ajustar los parámetros que definen las funciones de pertenencia [7].

Cuando las especificaciones del controlador han sido validadas el diseñador puede escoger entre tres tipos de implementaciones. La rama de la izquierda del flujo mostrado en la figura 1 muestra los pasos para obtener una realización basada en minimización lógica. **Xftl** recibe como entrada la descripción XFL del controlador y genera la correspondiente tabla entrada/salida con formato PLA de Berkeley. La salida tipo PLA puede ser minimizada y se pueden extraer las ecuaciones booleanas mediante cualesquiera de las herramientas de síntesis lógica compatibles con dicho formato. En nuestro caso, Xftl genera un fichero de comandos para Synopsys que selecciona las opciones de síntesis. La salida de Synopsys es un fichero con formato XNF (Xilinx Netlist Format) llamado *'fuzzy.sxnf'*. Opcionalmente se puede generar un fichero de información con datos sobre los requerimientos de rutado (número de CLBs e IOBs) y las restricciones temporales. Finalmente, *fuzzy.sxnf* se utiliza como fichero de entrada para la herramienta de Xilinx de mapeo y rutado del FPGA. Como resultado final se obtienen tres ficheros: *'fuzzy.lca'*, *'fuzzy.bit'* y la información sobre la implementación (*'fuzzy.rpt'*). El último paso consiste en programar el FPGA haciendo uso del fichero *'fuzzy.bit'*.

Dos implementaciones alternativas basadas en hardware dedicado pueden realizarse siguiendo la rama de la derecha del flujo mostrado en la figura 1. En este caso, **Xfvhdl** lee un fichero XFL y genera una descripción sintetizable en el lenguaje VHDL basada en una arquitectura específica de controladores difusos (ver [3] para una descripción detallada de la arquitectura). Las opciones arquitecturales y el número de bits de precisión son definidos por el usuario al ejecutar Xfvhdl.

Xfvhdl utiliza una librería de celdas parametrizadas que contiene las descripciones VDHL de los bloques básicos de la arquitectura del controlador. En la librería existen dos tipos de bloques: bloques que permiten construir la ruta de datos (implementan el mecanismo de inferencia) y bloques

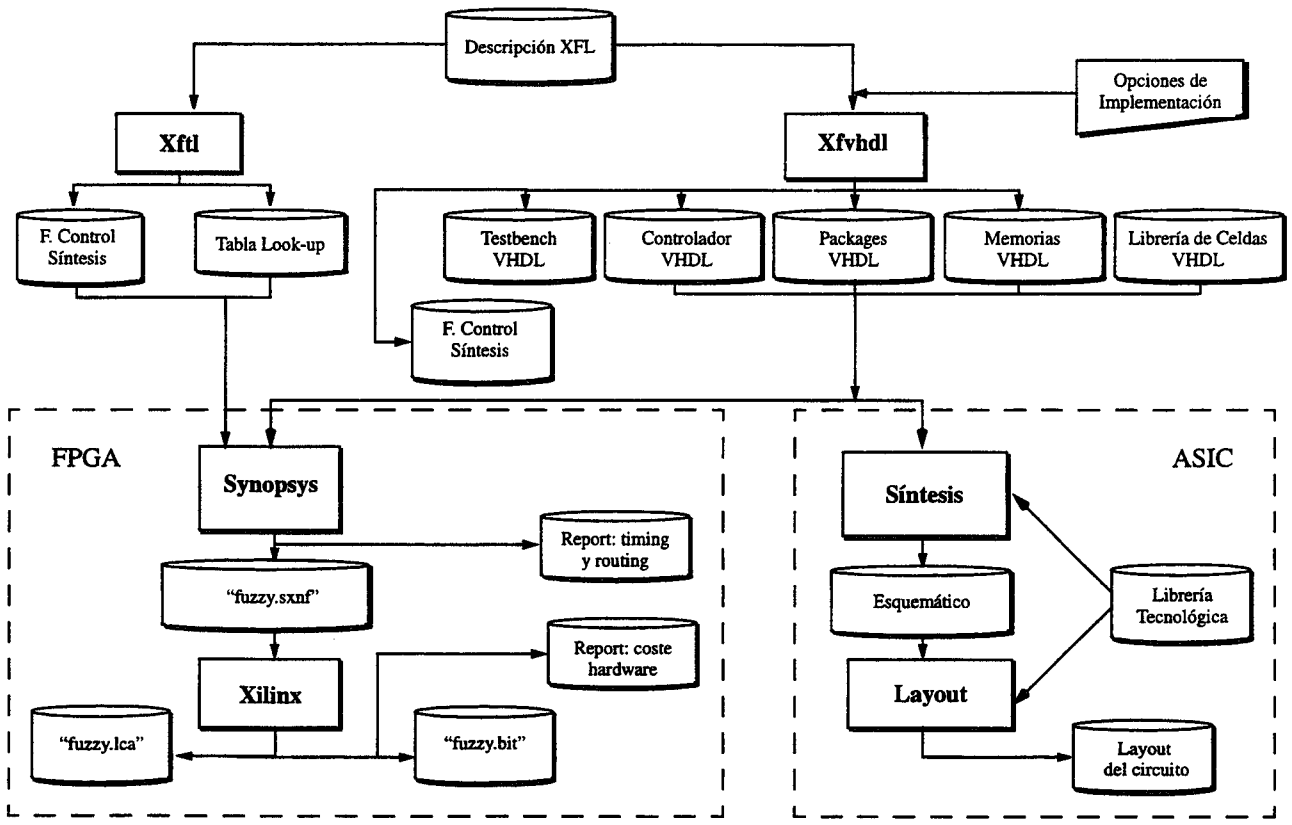



Fig. 1: Flujo de diseño para la implementación de controladores difusos desde Xfuzzy 2.0.

de control (dirigen las operaciones de escritura/lectura de las memorias y las señales que secuencian las tareas). El código usado en la descripción de la librería de celdas es compatible con las restricciones de síntesis que para VHDL imponen las herramientas de Synopsys y Mentor Graphics. Xfvhdl genera como salida los siguientes ficheros que describen el controlador:

*Ficheros de empaquetamiento*: Se generan dos ficheros de empaquetamiento VHDL. El fichero de *constantes* contiene las declaraciones de las constantes utilizadas en la descripción VHDL. Algunas de esas constantes se obtienen directamente de analizar la descripción XFL. Otras son especificadas por el usuario y, finalmente, un tercer conjunto de constantes se calculan a partir de los dos conjuntos anteriores. Por otro lado, el fichero de *entidades* contiene la declaración de todos los bloques que constituyen el controlador. La colocación de esos bloques depende de las opciones arquitecturales seleccionadas.

*Ficheros de la base de conocimientos*: La información sobre los antecedentes, reglas y consecuentes son codificadas en un conjunto de ficheros. Las definiciones están basadas en tablas de valores mediante instrucciones 'case' en VHDL. Ello permite realizar minimización booleana cuando los bloques se realizan como lógica combinacional.

*Fichero del controlador*: Este fichero contiene la descripción VHDL estructural del controlador difuso. El controlador se construye concadenando los bloques básicos de acuerdo con la descripción XFL y las opciones arquitecturales seleccionadas.

*Fichero de test*: Además de los ficheros requeridos para la síntesis se genera un fichero de test que facilita la verificación del controlador. La descripción del test contiene la colocación del controlador como un componente, un proceso que genera una señal de reloj y otro proceso que activa la señal de reset inicial así como las señales que actúan de entradas en la simulación del controlador.

Los ficheros generados por Xfvhdl pueden ser usados como entradas a las herramientas de síntesis de ASICs o FPGAs. La selección del estilo de implementación dependerá de las especificaciones o bien estará basada en aspectos económicos. Los distintos pasos del proceso de diseño se ven afectados fuertemente por dicha decisión.

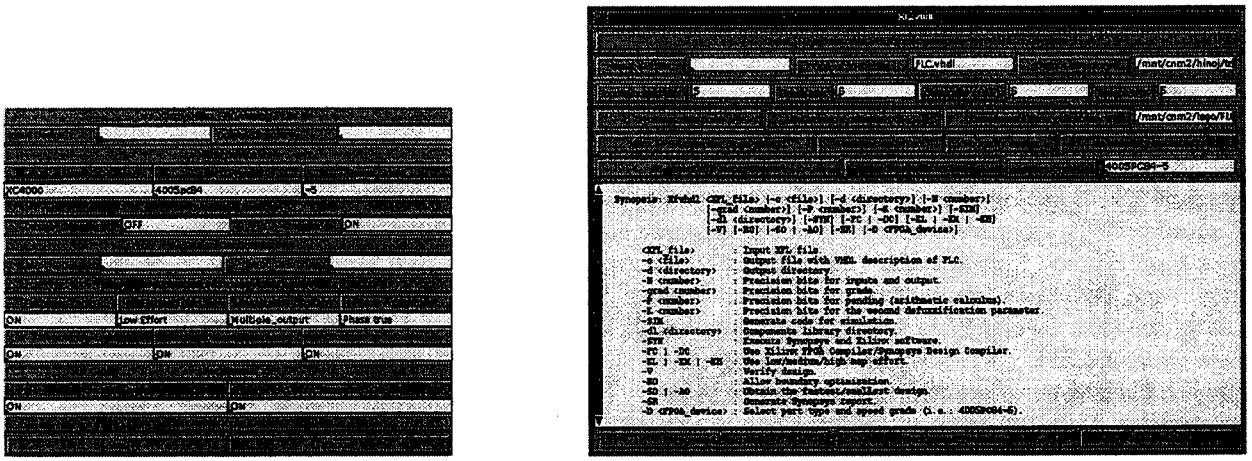
La realización de ASICs se basa en criterios de mapeo tecnológico. En este sentido, la implementación de controladores se reduce a la selección de celdas estándar y macroceldas de la tecnología objetivo. Los ficheros VHDL que provee Xfvhdl son utilizados por la herramienta de síntesis para generar el esquemático del controlador de acuerdo con la librería de la tecnología seleccionada. Este esquemático es usado por las herramientas de colocación y rutado para obtener el layout final.

Xfvhdl también genera un fichero de comandos que controla el proceso de síntesis con Synopsys. Los pasos requeridos para obtener el FPGA son similares a los descritos previamente.

Tanto Xftl como Xfvhdl pueden ejecutarse directamente, introduciendo las opciones de síntesis como parámetros del comando que ejecuta la herramienta, o a través de las interfaces gráficas proporcionadas por Xfuzzy. En la figura 2 se muestra la apariencia de las ventanas correspondientes.

## 3 EJEMPLOS DE DISEÑO

Con objeto de ilustrar la versatilidad de las herramientas desarrolladas y analizar el dominio de aplicación de las di-



a) b)

Fig. 2: Interfaces gráficas de Xfuzzy. a) Xftl. b) Xfvhdl

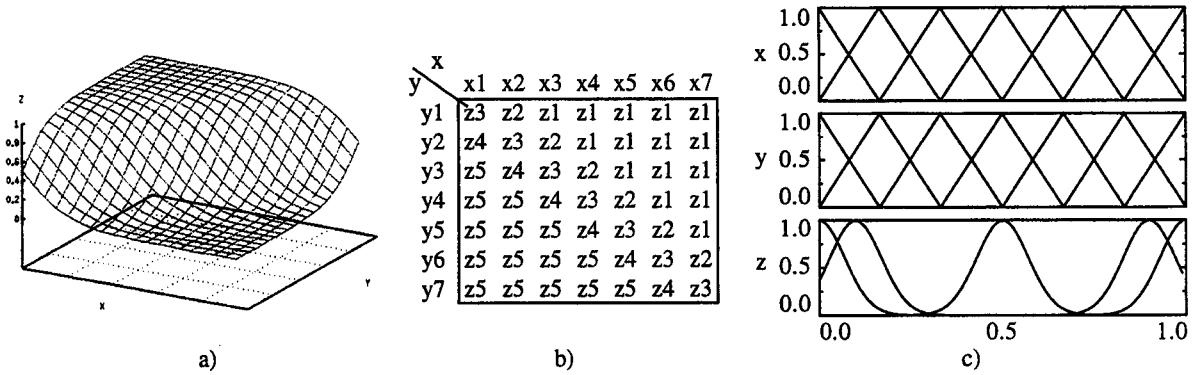



Fig. 3: Función F1: z=1/(1+exp(10*(x-y))). a) Superficie de la función. b) Base de reglas. c) Funciones de pertenencia de los antecedentes y consecuentes.

ferentes técnicas de implementación vamos a describir en esta sección la realización de controladores que actúan como aproximadores de funciones. La función objetivo elegida se muestra en la figura 3a. Para el caso de realizaciones sobre FPGAs, exploraremos el espacio de diseño considerando tres estrategias de implementación (una de ellas basada en la utilización de tablas entrada/salida y las otras dos basadas en una arquitectura específica que usa circuitos de funciones de pertenencia construidos mediante memorias o por cálculo aritmético [9]). Se analizará un rango de resolución de cuatro a catorce bits en todos los casos. En lo que se refiere al diseño de ASICs vamos a describir la implementación de dos prototipos de controlador con tres entradas y una salida, siete funciones de pertenencia y 6 bits de resolución. Los MFCs de uno de los prototipos están basados en memoria, mientras que en el otro utilizan cálculo aritmético. En todos los casos el método de defuzzificación empleado es la Media Difusa.

Usando las facilidades de Xfuzzy hemos identificado la base de reglas (figura 3b) y se han ajustado los parámetros para los controladores. Se han usado siete funciones de pertenencia triangulares para cubrir los universos de discurso de las entradas, mientras que sólo ha sido preciso utilizar cinco valores diferentes para los consecuentes (z1, ..., z5). Si bien en la figura 3c los consecuentes se han representado como funciones tipo campana, los métodos de defuzzificación simplificados que calculan las salidas del controlador como una media ponderada sólo requieren uno o dos parámetros significativos. Para el método de la Media Difusa (Fuzzy Mean, FM) el parámetro de ponderación es el centro de la campana. En el método de la Media Difusa Ponderada (Weighted Fuzzy Mean, WFM) el segundo parámetro de ponderación correspondería a la anchura de la base de la campana.

Todos los controladores han sido diseñados usando la familia de FPGAs XC4000 de Xilinx. Los prototipos ASICs se fabricaron en tecnología CMOS de 1.0 μm.

# 4 RESULTADOS EXPERIMENTALES

Para evaluar las prestaciones de las diferentes realizaciones con FPGAs hemos considerado tres características: 1) coste de la implementación en término del número de CLBs; 2) precisión de la aproximación medida por el error cuadrático medio (RMSE); y 3) velocidad de operación en función de los ciclos de reloj necesarios para realizar la inferencia.

La figura 4 muestra la evolución del coste para las tres técnicas de implementación sobre FPGAs cuando aumenta el número de bits. Para realizaciones de baja resolución la técnica de tablas proporciona los mejores resultados. Por el contrario, a medida que aumenta la resolución (más de 5 bits) las técnicas que utilizan hardware específico resultan más eficientes. Comparando las dos opciones arquitecturales, para implementaciones de alta resolución, la opción aritmética es mucho mejor que la opción de memoria.

En lo que se refiere a la precisión de la aproximación, ésta vendrá limitada por los siguientes factores: 1) el error de la aproximación difusa; 2) el error de truncamiento debido a

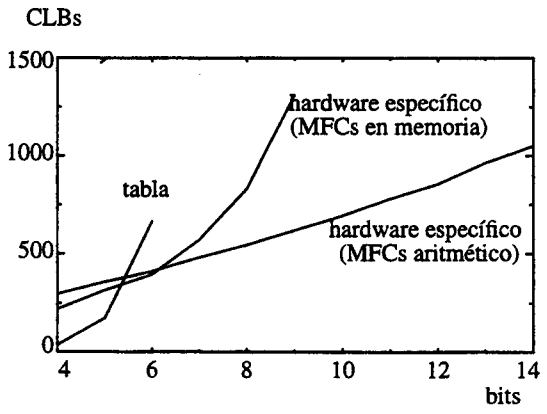



Fig. 4: Comparación de coste para FPGAs.

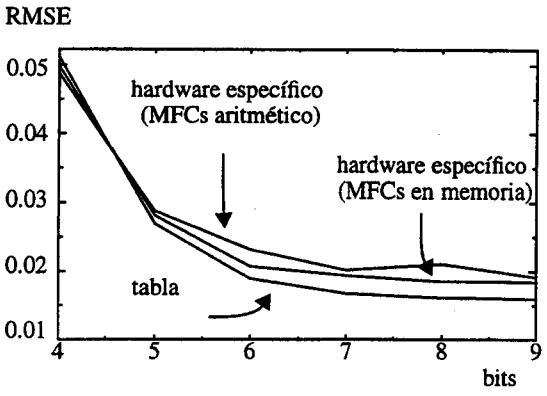



Fig. 5: Comparación del RMSE para FPGAs.

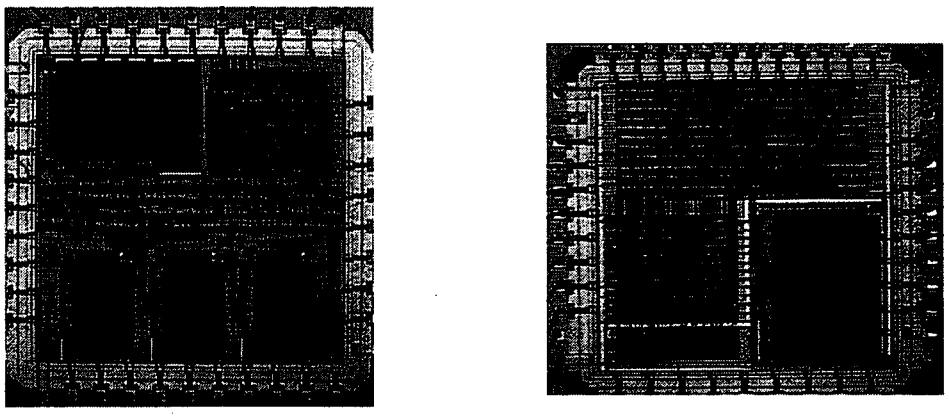
la anchura limitada de los buses; y 3) el error de redondeo introducido por la implementación hardware. El primer error es inherente al mecanismo de razonamiento aproximado usado por los sistemas que operan en lógica difusa. Este error puede ser minimizado incrementando el número de funciones de pertenencia para los antecedentes y consecuentes. La reducción del error de truncamiento se consigue aumentando la anchura de los buses. Finalmente, el error de implementación es una consecuencia de la aritmética de punto fijo utilizada en el controlador. En determinadas partes del circuito, concretamente en el defuzzificador, este error es acumulativo. La figura 5 representa el RMSE en función del número de bits. El comportamiento es similar para los tres tipos de implementación. Para bajas resoluciones (hasta 6 bits) el factor dominante es el error de truncamiento. Cuando la resolución aumenta, las principales fuentes de errores son debidos a la aproximación y al hardware. Comparando las tres técnicas de implementación, los errores de implementación tienen mayor influencia en los casos que usan hardware específico.

Un último aspecto en nuestro estudio se focaliza en la evaluación de la velocidad de inferencia de las diferentes técnicas. Las implementaciones basadas en tabla utilizan circuitos combinacionales. En este caso la velocidad de inferencia está limitada sólo por el tiempo de propagación del camino crítico. La operación del controlador es más lenta cuando se usa una arquitectura específica debido a que se requieren varios ciclos de reloj para realizar una inferencia. El throughput del sistema depende de la anchura del los buses. Por otro lado, la latencia también depende del número de etapas de pipeline (dos para el caso de MFCs basados en memoria y tres para el caso de MFCs basados en cálculo aritmético). Los resultados experimentales para controladores de cinco y seis bits implementados usando el FPGA 4013PQ160-5 de Xilinx muestran que una frecuencia de reloj de 5.5 MHz. es suficiente para garantizar la operación correcta. Esto supone una velocidad de inferencia sobre 5 MFLIPS para controladores basados en tablas y cercana a 1 MFLIPS para controladores basados en hardware específico.

En lo que se refiere al diseño de ASICs, se observa un comportamiento, en cuanto a coste, similar al obtenido en el caso de FPGAs. Para baja resolución el tamaño es similar en las opciones de MFCs en memoria o por cálculo aritmético, mientras que para altas resoluciones el coste es significativamente menor en la segunda opción.

La figura 6 muestra las microfotografías de dos prototipos ASICs que implementan controladores de resolución intermedia (6 bits). El área ocupada por el prototipo con MFCs en memoria (figura 6a) es 14.36 $mm^2$, mientras que la opción de cálculo aritmético (figura 6b) requiere 12.53 $mm^2$.

El RMSE medido para el segundo prototipo es 0.11. El error máximo obtenido está por debajo del 7%. En lo que



(a) (b)

Fig. 6: Prototipos de controladores difusos con MFCs basados en memoria (a), y MFCs aritméticos (b).

se refiere a la velocidad de inferencia ambos prototipos requieren 9 ciclos de reloj para realizar una inferencia. En el caso de MFCs almacenados en memoria dicho reloj opera a una frecuencia de 27MHz, dando lugar a una velocidad de inferencia superior a 3 MFLIPS. El prototipo con MFCs por cálculo aritmético requiere una frecuencia de reloj de 25 MHz, por lo que su velocidad de inferencia es de 2.77 MFLIPS.

De acuerdo con los resultados anteriores es necesario establecer un compromiso entre coste, precisión y velocidad durante el proceso de diseño de controladores difusos. La técnica de tabla es la mejor opción en términos de precisión y velocidad, pero los criterios de coste la hacen impracticable cuando la resolución aumenta. Las técnicas basadas en hardware específico son viables para controladores de mayor resolución. En particular, las arquitecturas que usan MFCs aritméticos pueden emplearse para la construcción de controladores de alta resolución.

## 5 CONCLUSIONES

Se han presentado dos herramientas de síntesis para la implementación hardware de controladores difusos mediante FPGAs o ASICs. Estas herramientas generan el controlador partiendo de una descripción de comportamiento de alto nivel. Una de las ventajas de esta aproximación es que las realizaciones están libres de errores de diseño (por la propia definición de síntesis automática). Otra ventaja es que las herramientas proporcionan un mecanismo para explorar de una manera sencilla y rápida el espacio de diseño, con lo que el diseñador puede escoger la mejor solución. Por último, Xfuzzy 2.0 es un entorno abierto. Esto nos permite introducir nuevos elementos en el entorno (otras arquitecturas de controladores, otros estilos de síntesis, etc).

## Referencias


[1] A. Barriga, S. Sánchez-Solano, C. J. Jiménez, D. Galán and D. R. López: "Automatic Synthesis of Fuzzy Logic Controllers", *Mathware & Soft Computing*, Vol. III, n. 3, pp. 425-434. Sept. 1996.

[2] D. L. Hung. "Dedicated Digital Fuzzy Hardware". *IEEE Micro*, vol. 15, n. 4, pp. 31-39, Aug. 1995.

[3] C. J. Jiménez, S. Sánchez-Solano, A. Barriga, "Hardware Implementation of a General Purpose Fuzzy Controller", *Proc. IFSA World Congress,* vol. 2, pp. 185-188, Sao Paulo, 1995.

[4] J. Y. Leong, M.H. Lim, K. T. Lau, "A General Approach to Encoding Heuristics on Programmable Logic Devices", *Proc. fifth IFSA World Congress,* pp. 917-920, Seul, 1993.

[5] D. R. López, S. Sánchez-Solano, A. Barriga: "XFL: a fuzzy logic systems language." *Proc. IEEE International Conference on Fuzzy Systems*, Barcelona, 1997.

[6] M. A. Manzoul, D. Jayabharathi, "FPGA for Fuzzy Controllers", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, n. 1, Jan. 1995.

[7] F. J. Moreno, D. R. López, A. Barriga, S. Sánchez-Solano, "XFBPA: A Tool for Automatic Fuzzy System Learning", *Proc. EUFIT*, 1997.

[8] T. Munakata, Y. Jani, "Fuzzy Systems: An Overview", *Communications of the ACM*, vol. 37, n. 3, pp. 69-76, Mar. 1994.

[9] S. Sánchez-Solano, A. Barriga, C. J. Jiménez, J. L. Huertas, "Design and Application of Digital Fuzzy Controllers", *Proc. IEEE International Conference on Fuzzy Systems*, Barcelona, 1997.

# OPERADORES DE CONSECUENCIAS BORROSOS, SU COHERENCIA Y GENERACION DE RELACIONES

**Jorge Elorza***
Dep. de Matemática e Informática
Universidad Pública de Navarra
Campus de Arrosadía s/n
31006 Pamplona. España
e-mail: elorza@si.upna.es
Tel: 948169556 Fax: 948169521
* *Correpondencia de autores*

**Pedro Burillo**
Dep. de Automática y Computación
Universidad Pública de Navarra
Campus de Arrosadía s/n
31006 Pamplona. España
e-mail: pburillo@si.upna.es
Tel: 948169252 Fax: 948169281

## Resumen

El propósito del presente artículo es analizar el concepto de operador coherente en lógica borrosa y el cumplimiento de los axiomas de preorden por parte de la relación inducida de un operador de consecuencias. Se presentan familias infinitas de operadores de consecuencias coherentes y no coherentes. Además dentro de los operadores que no son coherentes se muestran familias, también infinitas, de operadores de consecuencias tales que su relación inducida resulta o no preorden, poniendo de manifiesto que la coherencia de un operador es condición suficiente para obtener el cumplimiento de los axiomas de preorden de su relación inducida pero que no es necesaria.

**Palabras clave:** Fuzzy consequence operator, coherent operator, clausure operator, fuzzy preorder, fuzzy logic.

## 1 INTRODUCCION Y NOTACIONES

Es bien conocida la utilización de los operadores de consecuencia en logica clásica, introducidos por A. Tarski ([7]) en 1956. J.L. Castro y E. Trillas demuestran en 1.989 ([2]) la equivalencia entre el concepto de operador de consecuencias en el sentido de Tarski, de naturaleza puramente lógica y el concepto de preorden, de naturaleza algebraica.

Desde que L. Zadeh introduce en 1.965 ([11]) la teoría de conjuntos difusos ( Fuzzy Set Theory) son numerosos los estudios realizados en lógicas multivaloradas y lógicas fuzzy, tanto en el ámbito teórico como en el de las aplicaciones a diferentes campos científicos. En tales lógicas resulta relevante el concepto de operador de consecuencias difuso (Fuzzy Consequence Operator), abreviadamente FCO. Dado $X$ un conjunto referencial no vacío que representará un conjunto de proposiciones y un retículo completo $L$, que será el rango de las funciones características de los subconjuntos fuzzy de $X$, J. Pavelka introduce el concepto de FCO sobre $X$ ([6]) , extendiendo, de manera natural el concepto de operador de consecuencias de Tarski.

Un FCO sobre $X$ es una aplicación $C : L^X \longrightarrow L^X$, verificando :
(C1) $\mu \subset C(\mu)\ \forall\ \mu \in L^X$ (inclusión)
(C2) Si $\mu_1 \subset \mu_2 \Longrightarrow C(\mu_1) \subset C(\mu_2)\ \forall\ \mu_1,\ \mu_2 \in L^X$ (monotonía)
(C3) $C(C(\mu)) = C(\mu)\ \forall\ \mu \in L^X$ (idempotencia)

Bajo el punto de vista lógico, la interpretación de los axiomas que definen el operador $C$ puede ser la siguiente : si $X$ es un conjunto de proposiciones y $\mu$ representa un subconjunto de axiomas y $C(\mu)$ la teoría que generan, entonces :
(C1) indica que los axiomas quedan recogidos en la teoría.
(C2) indica que si crece el conjunto de axiomas la teoría es mayor.
(C3) indica que la teoría se estaciona en el primer paso, es decir, es completa en el sentido de que si se toma toda ella como nuevo conjunto de axiomas para construir una nueva teoría, ésta no aumenta.

En la última década, estos operadores, en ocasiones llamados operadores clausura, han sido estudiados ([1]), ([3]), ([4]), ([5]), ([9]), ([10]) en el contexto de la lógica borrosa, tomando como caso particular, la cadena $L = [0, 1]$.

En concreto, es conocido ([1]), ([10]), que si $R$ es un preorden borroso en $X$, entonces el operador de $[0,1]^X$ en $[0,1]^X$ definido mediante la composición de $\mu$ con $R$, es decir, $C_R(\mu) = \mu \circ R$, donde

$$\mu \circ R(x) = \sup_{w \in X} \Big\{ min\Big(\mu(w), R(w,x)\Big)\Big\} \quad (1)$$

es un operador de consecuencias en $X$.

En 1.991, E. Trillas y J.L. Castro ([1]), ([10]), añaden el axioma de coherencia a la definición de operador de consecuencias. Así un FCO es coherente si

$$min\big(\mu(a), C(\varphi_a)(x)\big) \leq C(\mu)(x) \quad (2)$$

$$\forall\ \mu \in [0,1]^X \quad \forall\ (a,x) \in X \times X$$

donde $\varphi_a(t) = \varphi_{\{a\}}(t) = \begin{cases} 1 & t = a \\ 0 & t \neq a \end{cases}$ es la función característica ordinaria del singulete $\{a\}$.

La importancia de los FCO coherentes radica en el siguiente teorema :

**Teorema.1**.- ([1]), ([10]) *Si $C$ es un FCO coherente, entonces la relación definida por $R_C(x,y) = C(\varphi_x)(y)$ es un preorden borroso (inducido por $C$).*

E. Trillas y J.L. Castro ([1]), ([10]), prueban además que el operador $C_R$ es coherente, cualquiera que sea el preorden $R$.

Por otro lado, es claro que todo operador de consecuencias clásico es coherente, lo que proporciona la equivalencia entre dicho concepto y el de preorden clásico. Sin embargo, la condición de coherencia, en general, se pierde en el caso difuso.

A pesar de ello, no se observa en la literatura estudiada un ejemplo de operador FCO no coherente. En el párrafo 3, se presenta dicho ejemplo y se proporcionan familias de operadores FCO no coherentes sobre un universo cualquiera $X$ con card $X \geq 2$.

Sobre el referencial $X$, consideraremos las siguientes familias :
$\Gamma$ el conjunto de todos los preórdenes en $X$.
$\Omega$ el conjunto de todos los operadores de consecuencias en $X$.
$\tilde{\Omega}$ la subfamilia de $\Omega$ de los operadores de consecuencias coherentes en $X$.

Obviamente quedan definidas dos aplicaciones :

$$\theta : \Gamma \to \tilde{\Omega} \quad \text{dada por} \quad \theta(R) = C_R \quad \text{y}$$

$$\tilde{\theta} : \tilde{\Omega} \to \Gamma \quad \text{dada por} \quad \tilde{\theta}(C) = R_C \ .$$

Sea, además, $\Omega_{rt}$ el conjunto de FCO, $C$, tales que su relación inducida, $R_C$, resulta un preorden en $X$. Donde los subíndices $r$, $t$ hacen referencia al cumplimiento de la propiedad reflexiva y transitiva, respectivamente, de $R_C$.

Por último, sea $\Omega_p = \Im m(\theta)$. $\Omega_p$ será llamado conjunto de operadores preordenados.

Con ésto, se tiene la siguiente cadena de contenidos :

$$\Omega_p \subset \tilde{\Omega} \subset \Omega_{rt} \subset \Omega \quad (3)$$

Nótese que el primer contenido lo proporciona el hecho de que $C_R$ es siempre un FCO coherente, el segundo es cierto gracias al teorema 1 y el tercero es obvio.

No aparecen, sin embargo, en la literatura, ejemplos que justifiquen que todos los contenidos de la cadena anterior son estrictos. El objetivo del presente artículo es proporcionar estos ejemplos mediante familias de operadores de consecuencias.

En el siguiente párrafo se estudian propiedades de la familia de operadores de consecuencias que se ofrece en [6].

## 2 UNA REPRESENTACION DE LOS OPERADORES DE CONSECUENCIAS

Para cada familia $S$ de subconjuntos difusos de $X$, J. Pavelka ([6]), introduce el operador $C_S$ tal que a cada difuso $\mu$ le asigna otro difuso $C_S(\mu)$ definido por :

$$C_S(\mu)(x) = \inf_{\nu \in S\,,\, \nu \supset \mu} \Big\{\nu(x)\Big\} \quad (4)$$

donde debe entenderse, naturalmente, que si el conjunto $\{\nu \in S \mid \nu \supset \mu\} = \emptyset$ y en particular si $S = \emptyset$, $C_S(\mu) = X$, donde $X$ representa el difuso unidad : $X(x) = 1\ \forall x \in X$

Dado un operador $C$ de $[0,1]^X$ en $[0,1]^X$, $S_C$ denotará la familia de los puntos fijos del operador $C$.

Las primeras propiedades de esta familia de operadores quedan resumidas en el siguiente teorema :

**Teorema.2**.- ([6]) (i) *$C_S$ es un FCO, cualquiera que sea la familia de subconjuntos difusos $S$.*
(ii) *Si $\mu \in S$, entonces $C_S(\mu) = \mu$.*
(iii) *Si $C$ es un FCO y $S$ es un sistema clausura, es decir, una familia de difusos cerrada para la intersección, entonces $C_{S_C} = C$ y $S_{C_S} = S$.*

Hagamos notar que como consecuencia del teorema anterior, dado un operador de consecuencias $C$, existe una familia $S$ de difusos tal que $C = C_S$, a saber $S = S_C$, obteniendo así una representación de los operadores de consecuencias sobre un universo $X$. Sin embargo la familia $S$ que representa al operador $C$ no está unívocamente determinada.

En el siguiente párrafo nos será útil observar que si la familia $S$ está formada por un único difuso, pongamos $S = \{\mu\}$, entonces

$$C_S(\rho) = \begin{cases} \mu & \rho \subset \mu \\ X & \text{otro caso} \end{cases}$$

Veamos dos condiciones suficientes, sobre la familia $S$ elegida, para obtener la coherencia del correspondiente operador. Para ello, introducimos la siguiente notación :

$\Phi_1$ denotará la familia de todas las funciones características ordinarias de los subconjuntos unipuntuales de X, es decir, $\Phi_1 = \{\varphi_x \mid x \in X\}$.

Asimismo $\Phi$ representará la familia de los subconjuntos ordinarios (crisp) de $X$, excluyendo el vacío, es decir, $\Phi = \{\varphi_A \mid A \subset X\} \setminus \{\emptyset\}$

**Teorema.3**.- *Si $S \supset \Phi_1$, entonces el operador $C_S$ es coherente.*

Demostración.- En efecto, sean $\mu \in [0,1]^X$ y $a, x \in X$ cualesquiera, como $\varphi_a \in \Phi_1$, por el teorema 2 (ii), se tiene :

$$min\big(\mu(a)\,,\, C_S(\varphi_a)(x) = min(\mu(a), \varphi_a(x)) =$$

$$= \begin{cases} 0 & \text{si} \quad x \neq a \\ \mu(a) & \text{si} \quad x = a \end{cases} \leq \mu(x) \leq C_S(\mu)(x)$$

Por tanto $C_S$ es coherente. □

**Teorema.4**.- *Si $S \supset \Phi^c$, entonces el operador $C_S$ resulta el operador identidad en $[0,1]^X$.*

Demostración.- Consideremos la familia $S = \Phi^c$ y sea $\mu \in [0,1]^X$. Veamos que $C_S(\mu) = \mu$ :

Si $\mu \in S$, es obvio aplicando el teorema 2 (ii).

Si $\mu \notin S$, es decir, $\mu \in \Phi$, entonces, $\mu$ es normal, es decir, $\mu$ toma el valor 1 sobre algún elemento de $X$.

Sea $x \in X$ tal que $\mu(x) = 1$, entonces claramente $C_S(\mu)(x) \geq \mu(x) = 1$, luego $C_S(\mu)(x) = 1$.

Sea $x \in X$ tal que $\mu(x) < 1$, entonces $\mu(x) = 0$.

Para cada $\epsilon \in (0,1)$, consideramos el difuso $\nu_{x,\epsilon}$ dado por

$$\nu_{x,\epsilon}(t) = \begin{cases} 1 & \text{si} \quad \mu(t) = 1 \\ \epsilon & \text{si} \quad t = x \\ 0 & \text{otro caso} \end{cases}$$

Desde luego $\nu_{x,\epsilon} \in S$. Así,

$$C_S(\mu)(x) = \inf_{\nu \in S\,,\, \nu \supset \mu} \Big\{\nu(x)\Big\} \leq$$

$$\leq \nu_{x,\epsilon}(x) = \epsilon \;\forall \epsilon \in (0,1)$$

Luego $C_S(\mu)(x) = 0$ y por tanto :

$$C_S(\mu)(x) = \begin{cases} 1 & \text{si} \quad \mu(x) = 1 \\ 0 & \text{si} \quad \mu(x) = 0 \end{cases} = \mu(x)$$

Basta observar ahora que si $S \supset \Phi^c$, $\mu = C_{\Phi^c}(\mu) \supset C_S(\mu)$ y por el axioma de inclusión del operador $C_S$, $\mu \subset C_S(\mu)$ y por tanto $C_S(\mu) = \mu$, cualquiera que sea el difuso $\mu$. □

Como corolario del teorema anterior, se tiene :

**Teorema.5**.- *Si $S \supset \Phi^c$, entonces el operador $C_S$ es coherente.*

Demostración.- Basta probar que el operador que a cada $\mu$ le hace corresponder $C(\mu) = \mu$ es coherente, lo cual es inmediato :

$$min\big(\mu(a)\,,\, C(\varphi_a)(x) = min(\mu(a), \varphi_a(x)) =$$

$$= \begin{cases} 0 & \text{si} \quad x \neq a \\ \mu(a) & \text{si} \quad x = a \end{cases} \leq \mu(x) = C(\mu)(x) \qquad \square$$

Veamos ahora, con un ejemplo, que los recíprocos de los teoremas anteriores no son ciertos.

**Ejemplo.1**.- Sea $S = \{X\}$, entonces claramente $C_S(\mu) = X$, para todo difuso $\mu$ y entonces $C_S$ es coherente. En efecto, dados $a, x \in X$ y $\mu \in [0,1]^X$, $C_S(\mu)(x) = X(x) = 1 \geq min\big(\mu(a)\,,\, C_S(\varphi_a)(x)\big)$. Sin embargo $S$ no contiene a $\Phi_1$ ni a $\Phi^c$.

## 3 ESTUDIO DE LOS CONTENIDOS ESTRICTOS

El objetivo de este párrafo es probar, mediante la representación de los FCO introducida en el párrafo anterior, que los contenidos de la cadena (3) son todos estrictos.

Hagamos notar ahora los casos triviales en los que, según la cardinalidad del universo $X$, la cadena de contenidos (3) degenera.

**Observación.1**.- Sea $C$ un FCO sobre $X$, entonces :

$$min\big(\mu(x), C(\varphi_x)(x)\big) = min(\mu(x), 1) =$$

$$= \mu(x) \leq C(\mu)(x)$$

Por tanto, si el cardinal de $X$ es 1 se tiene :

$$\tilde{\Omega} = \Omega_{rt} = \Omega$$

**Observación.2**.- Sea $C$ un FCO sobre $X$ y sean $x$, $y$, $z \in X$ tales que no son distintos dos a dos, entonces :

$$C(\varphi_x)(z) \geq min\Big(C(\varphi_x)(y), C(\varphi_y)(z)\Big) \qquad (5)$$

es decir

$$R_C(x,z) \geq min\Big(R_C(x,y), R_C(y,z)\Big) \qquad (6)$$

En efecto :
Si $x = y$, $C(\varphi_x)(z) = C(\varphi_y)(z)$ y se tiene (5).
Si $y = z$, $C(\varphi_x)(z) = C(\varphi_x)(y)$ y se tiene (5).
Si $x = z$, basta hacer uso de la inclusión de $C$ para obtener $C(\varphi_x)(x) \geq \varphi_x(x) = 1$ y se tiene (5).

En particular, si el cardinal de $X$ es 2 se tiene

$$\Omega_{rt} = \Omega$$

En efecto, por lo anterior $R_C$ es transitiva. Además, si $C$ verifica la inclusión, cualquiera que sea el cardinal de $X$, $R_C$ es reflexiva :

$$R_C(x,x) = C(\varphi_x)(x) \geq \varphi_x(x) = 1$$

### 3.1. $\boxed{\Omega_p \neq \tilde{\Omega}}$

Veamos en primer lugar un resultado :

**Teorema.6**.-$\Im m(\theta) = \Big\{C \in \tilde{\Omega} \mid C_{R_C} = C\Big\}$, *es decir,* $\Im m(\theta) = \Big\{C \in \tilde{\Omega} \mid \theta \circ \tilde{\theta}(C) = C\Big\}$.

Demostración.- Si $C \in \Im m(\theta)$ , existe un único preorden $R$ tal que $\theta(R) = C$. Por otro lado, sabemos que $\tilde{\theta} \circ \theta(R) = R$. Así :

$$\theta \circ \tilde{\theta}(C) = (\theta \circ \tilde{\theta}) \circ \theta(R) = \theta \circ (\tilde{\theta} \circ \theta)(R) = \theta(R) = C$$

Reciprocamente si $\theta \circ \tilde{\theta}(C) = C$, C es trivialmente imagen por $\theta$ del preorden $\tilde{\theta}(C)$ y $C \in \Im m(\theta)$ □

La siguiente familia de operadores prueba que $\Omega_p \neq \tilde{\Omega}$.

**Teorema.7**.-*Sea $X$ un conjunto universal cualquiera. Para cada familia de difusos $S$ tal que $S \supset \Phi_1$ y $S \neq [0,1]^X$, $C_S \in \tilde{\Omega} \setminus \Omega_p$.*

Demostración.- En efecto, por el teorema 3, $C_S \in \tilde{\Omega}$, bastará entonces probar que $C_S \notin \Omega_p$ y por el teorema 6, equivale a probar $\theta \circ \tilde{\theta}(C_S) \neq C_S$

$$\theta \circ \tilde{\theta}(C_S)(\mu)(x) = \sup_{w \in X}\Big\{min\Big(\mu(w), C_S(\varphi_w)(x)\Big)\Big\} =$$

$$= \sup_{w \in X}\Big\{min\Big(\mu(w), \varphi_w(x)\Big)\Big\}$$

Observemos que este supremo se alcanza en $w = x$, así

$$\theta \circ \tilde{\theta}(C_S)(\mu)(x) = \mu(x) \neq C_S(\mu) \text{ en general.}$$

Luego $\theta \circ \tilde{\theta}(C_S) \neq C_S$ y por tanto $C_S \in \tilde{\Omega} \setminus \Omega_p$. □

Hagamos notar que, como consecuencia, hemos probado que la aplicación $\theta$ definida anteriormente entre el conjunto $\Gamma$ de preordenes borrosos y el conjunto $\tilde{\Omega}$ de los operadores de consecuencias coherentes no es suprayectiva.

### 3.2. $\boxed{\tilde{\Omega} \neq \Omega_{rt}}$

La siguiente familia de operadores resuelve el contenido $\tilde{\Omega} \neq \Omega_{rt}$, lo que pone de relieve que la coherencia de un operador $C$ es condición suficiente para que su relación inducida $R_C$ resulte un preorden, pero no necesaria y que por tanto el recíproco del teorema 1 no es cierto.

**Teorema.8**.-*Sea $X$ un conjunto cualquiera con card $X \geq 2$. Distingamos en $X$ dos elementos distintos $a$ y $b$ y un difuso $\mu$ no normal $(\mu(x) < 1 \; \forall x \in X)$ y tal que $\mu(b) < \mu(a)$. Tomemos $S = \{\mu\}$. Entonces, $C_S \in \Omega_{rt}$ pero $C_S$ no es coherente.*

Demostración.- Desde luego, por ser $C_S$ verificando el axioma de inclusión, $\tilde{\theta}(C_S)$ es reflexiva. Por ser $\mu$ un difuso no normal, no contiene a ninguna función característica $\varphi_x$, luego :

$$\tilde{\theta}(C_S)(x,z) = C_S(\varphi_x)(z) = 1 \geq$$

$$\geq min\big(C_S(\varphi_x)(y), C_S(\varphi_y)(z)\big) =$$

$$= min\big(\tilde{\theta}(C_S)(x,y), \tilde{\theta}(C_S)(y,z)\big) \; \forall x,y,z \in X$$

Por tanto $\tilde{\theta}(C_S)$ es un preorden.

Sin embargo $C_S$ no es coherente :

$$min\big(\mu(a), C_S(\varphi_a)(b)\big) = min(\mu(a), 1) =$$

$$= \mu(a) > \mu(b) = C_S(\mu)(b) \quad \square$$

### 3.3. $\boxed{\Omega_{rt} \neq \Omega}$

Observemos que si un operador $C$ verifica el axioma de inclusión, necesariamente la relación inducida $R_C$ es reflexiva :

$$R_C(x,x) = C\big(\varphi_x(x)\big) \geq \varphi_x(x) = 1 \; \forall x \in X$$

En particular todo FCO, C, en $\Omega$ verifica que su relación inducida es reflexiva. Por tanto el incumplimiento de la condición de preorden por parte de la relación asociada de un FCO debe radicar en la no verificación del axioma de transitividad.

Veamos una familia de operadores de consecuencias cuya relación asociada no es un preorden. En particular no serán coherentes.

Teorema.9.-*Sea $X$ un conjunto cualquiera, con card $X \geq 3$. Distingamos en $X$ tres elementos distintos dos a dos $x$, $y$, $z$ y un difuso $\mu$ tal que $\mu(x) = 1 \neq \mu(y) > \mu(z)$. Tomemos $S = \{\mu\}$. Entonces, $C_S \in \Omega \setminus \Omega_{rt}$.*

Demostración.- Sabemos que $C_S \in \Omega$. Observemos que $\mu$ contiene a $\varphi_x$ y no contiene a $\varphi_y$. Así,

$$\tilde{\theta}(C_S)(x,z) = C_S(\varphi_x)(z) = \mu(z) < \mu(y) =$$

$$= min\big(\mu(y), 1\big) = min\big(C_S(\varphi_x)(y), C_S(\varphi_y)(z)\big) =$$

$$= min\big(\tilde{\theta}(C_S)(x,y), \tilde{\theta}(C_S)(y,z)\big)$$

luego $\tilde{\theta}(C_S)$ no es tansitiva y por tanto no es un preorden sobre $X$. $\square$

## 4 CONCLUSIONES Y LINEAS FUTURAS

En el presente trabajo se ha puesto de manifiesto, a través de la representación de los operadores de consecuencias que se nos ofrece en J. Pavelka ([6]), la existencia de familias amplias de FCO coherentes y no coherentes y se han estudiado sus vinculaciones con el cumplimiento o no de los axiomas de preorden borroso por parte de sus relaciones inducidas. De hecho se han construído familias de operadores de consecuencias no coherentes tales que sus relaciones inducidas han resultado ser un preorden y familias de la misma naturaleza cuyas relaciones inducidas no son preordenes.

Respecto de la aplicación $\theta$ que a cada preorden $R$ le asocia el operador $C_R$, se ha caracerizado (teorema 6) el conjunto imagen de dicha aplicación, poniendo de manifiesto que dicha aplicación no es suprayectiva.

Será interesante analizar distintas clasificaciones de los operadores de consecuencias en atención a su coherencia, a las relaciones inducidas, a sus puntos fijos, así como la incidencia de estas propiedades en inferencia borrosa.

También será una vía de interés el análisis de propiedades de la aplicación $\theta$, dando especial importancia a aquellas propiedades relativas a la conservación de operaciones definidas en el conjunto de los preordenes borrosos.


### Agradecimientos

Agradecemos los comentarios y sugerencias aportados por el profesor Enric Trillas.


### Referencias


[1] Castro, J.L. - Trillas, E. (1991) *Tarski's fuzzy consequences*, submit to IJAR (Proc. of the Internat. Fuzzy Engineering Symp'91. Vol.1, pg. 70-81.)

[2] Castro, J.L. - Trillas, E. (1989) *Sobre preórdenes y operadores de consecuencias de Tarski.* Theoria, Segunda época, Curso 1.988-1.989, N.11, pg. 419-425.

[3] Castro, J.L. - Trillas, E. - Cubillo, S. (1994) *On consequence in approximate reasoning.* Journal of Applied Non-Classical Logics. Volume 4 n.1/1.994, pg. 91-103.

[4] Gerla, G. (1994) *An Extension Principle for Fuzzy Logics.* Mathematical Logic Quartely, 40, pg. 357-380.

[5] Gerla, G. (1994) *Comparing fuzzy and crisp deduction systems.* Fuzzy sets and systems 67, pg. 317-328.

[6] Pavelka, J. (1979) *On Fuzzy Logic I.* Zeitschr. f. Math. Logik und Grundlagen d. Math. Bd.25, pg. 45-52.

[7] Tarski, A. (1956) *Logic, semantics, metamathematics.* Clarendon Press, Oxford.

[8] Trillas, E. - Alsina, C. (1993) *Logic : Going farther from Tarski?* Fuzzy sets and systems 53, pg. 1-13.

[9] Trillas, E. - Alsina, C. (1992) *Shyntesizing implications.* International Journal of Artificial Intelligence, 7, pg. 705-713.

[10] Trillas, E. - Alsina, C. - Terricabras, J.M. (1995) *Introducción a la lógica borrosa.* Ed.Ariel.

[11] Zadeh, L.A. (1965) *Fuzzy set.* Information and control, 8, pg. 338-353.

# μ-T-Condicionalización de una relación difusa. Medida de μ-T- Condicionalidad.

**Luis Garmendia Salvador**
Univ. Alfonso X el Sabio
e-mail:luisgar@uax.es

## Resumen

Dada una relación difusa que no es T-condicional, se define la relación T-condicionalizada, dos medidas de μ-T-condicionalidad, asi como un método fácil para estudiar si la relación es T-condicional, encontrando, en su caso, las regiones de T-incondicionalidad para cada norma triangular T.

**Palabras Clave:** Relación difusa, T-Condicional, T-Condicionalización, Medida de T-Condicionalidad.

## 1 Introducción

Es bien sabida la importancia de la propiedad de condicionalidad de relaciones difusas para que se verifique la propiedad del MODUS PONENS y de este modo obtener consecuencias lógicas al aplicar la regla composicional de inferencia.

Sea T una norma triangular y μ un conjunto difuso sobre $E_1 \cup E_2$. Una relación R: $E_1 \times E_2 \rightarrow [0,1]$ es μ-T-condicional si $T(\mu(a),R(a,b)) \leq \mu(b)$ para todo (a,b) perteneciente a $E_1 \times E_2$. Para las relaciones R que no verifican esta propiedad, se propone definir una relación que denominaremos $R^{T\text{-cond}}$ que sea condicional, esté contenida en la primera y que diste de ella una distancia mínima; es decir se contruye una relación, que reduciendo a R lo mínimo posible, pase a ser condicional. Dada una relación R y una norma triangular T, cada relación condicionada $R^{T\text{-cond}}$ definida es única. Se define una **distancia de incondicionalidad** entre la relación R dada y la relación $R^{T\text{-cond}}$ construida; lo que supone tener una medida de lo que se tiene que reducir la relación para que sea T-condicional. Dicha distancia puede usarse como algoritmo reconocedor de relaciones condicionales, ya que si R es T-condicional entonces esta distancia es cero. Además se define una **medida de incondicionalidad sobre la verificación del Modus Ponens** analizando la diferencia entre $T(\mu(x),R(\mu(x), \mu(y)))$ y $\mu(y)$ cuando $T(\mu(x),R(\mu(x), \mu(y))) > \mu(y)$.

Se estudian dos casos, cuando la relación esta definida en un universo $[0,1]\times[0,1]$ y es continua pudiendo entonces representar la distancia de incondicionalidad y la medida de incondicionalidad por un volumen, mientras que cuando la relacion esta definida sobre dos universos discretos, la distancia de incondicionalidad y la medida de incondicionalidad son una suma de diferencias. Este último es caso particular del primero.

## 2 Preliminares

Sea R: $[0,1]\times[0,1] \rightarrow [0,1]$ una relación difusa. Se denomina $T_R$ a la relación definida por:

$T_R: [0,1]\times[0,1] \rightarrow [0,1]$ tal que $T_R(x,y)=T(x, R(x,y))$

Sea $J^T$: $[0,1]\times[0,1] \rightarrow [0,1]$ tal que $J^T(x,y)= \text{Sup}\{z \mid T(x,z)\leq y\}$.

## 3 T-Condicionalización de una relación difusa

**Definición 1:**

Dada una relación difusa R y una norma triangular T se define la relación T-condicionalizada $\mathbf{R^{T\text{-cond}}}$ como:

$$\mathbf{R^{T\text{-cond}}}(x, y)=\begin{cases} R(x,y) & \text{si } T_R(x,y)) \leq y \\ J^T(x,y) & \text{si } T_R(x,y) > y \end{cases}$$

**Proposición 1:**

***$R^{T\text{-cond}}(x, y) = Min(R(x,y), J^T(x,y))$***

Demostración

Si $R(x,y) \leq J^T(x,y)$ entonces $T_R(x,y)=T(x, R(x,y)) \leq y$, luego $R^{T\text{-cond}}(x, y) = R(x,y)$ por definición.

Si $R(x,y) > J^T(x,y)$ entonces $T_R(x,y)=T(x,R(x,y)) > y$, luego $R^{T\text{-cond}}(x, y) = J^T(x,y)$ por definición. Por tanto $R^{T\text{-cond}}(x, y) = \text{Min}(R(x,y), J^T(x,y))$

**Proposición 2:**

***La relación $R^{T\text{-cond}}$ es T-condicional.***

Demostración

Se debe probar que verifican que $T(x, R^{T\text{-cond}}(x,y)) \leq y$ para todo (x,y).

Si $T_R(x,y)=T(x,R(x,y)) \leq y$ la relación R ya es condicional y por tanto tambien lo es $R^{Tcond}$. Si $T_R(x,y)>y$ por definición de $J^T$ como el supremo de las relaciones que verifican la condición de condicionalidad entoces $R^{Tcond}$ tambien es T-condicional.

**Proposición 3:**

***Si T'' y T' son normas triangulares tales que $T' \leq T''$ entonces $R^{T''\text{-cond}} \subseteq R^{T'\text{-cond}}$***

Demostración

En estos casos se definen:

$$\mathbf{R}^{\mathbf{T'cond}}(x,y)=\begin{cases} R(x,y) & \text{si } T'_R(x,y)) \le y \\ J^{T}(x,y) & \text{si } T'_R(x,y) > y \end{cases}$$

y

$$\mathbf{R}^{\mathbf{T''cond}}(x,y)=\begin{cases} R(x,y) & \text{si } T''_R(x,y)) \le y \\ J^{T'}(x,y) & \text{si } T''_R(x,y) > y \end{cases}$$

Debemos estudiar tres regiones:
a) Si $T''_R(x,y) \le y$ entonces $T'(x,y) \le T''(x,y) \le y$, luego en esa región coinciden las relaciones R, $R^{T'\text{-cond}}$ y $R^{T''cond}$.
b) Si $T'_R(x,y) \le y$ pero $T''(x,y) > y$ entonces $R^{T'\text{-cond}}(x,y) = R(x,y)$, mientras que
$R^{T''\text{-cond}}(x,y) = J^{T''}(x,y) = \text{Min}\{R(x,y), J^{T''}(x,y)\}$ por la proposición 1.
c) Si $T'_R(x,y)>y$ entonces $R^{T''\text{-cond}}(x,y) = J^{T''}(x,y) = \text{Sup}\{z \mid T''(x,z)\le y\}$ y
$R^{T'\text{-cond}}(x,y) = J^{T'}(x,y) = \text{Sup}\{z \mid T'(x,z)\le y\}$. Al ser $T' \le T''$ sabemos que $J^{T''}(x,y) \le J^{T'}(x,y)$.

Como consecuencia inmediata se tiene que:
$\mathbf{R^{Min\text{-}cond} \subseteq R^{T\text{-}cond} \subseteq R}$. Por tanto la relación $R^{Min\text{-}cond}$ es condicional para cualquier norma triangular T, pero si T es distinto del mínimo se debe buscar la $R^{T\text{-}cond}$ para hacer mínima la distancia.
Tambien tenemos que
$\mathbf{R^{Min\text{-}cond} \subseteq R^{Prod\text{-}cond} \subseteq R^{Lukasw\text{-}cond} \subseteq R}$

**Construcción de $\mathbf{R^{Min\text{-}cond}}$**

Cuando **T=Min**, en el dominio donde R no es Min-condicional ha de ser x>y, por lo que $J^{Min}(x,y)=y$. Por lo tanto se obtiene la siguiente construcción:

Dada una relación difusa R se construye la relación $\mathbf{R^{Min\text{-}cond}}$:

$$\mathbf{R}^{\mathbf{Min\text{-}cond}}(x,y)=\begin{cases} R(x,y) & \text{si } \text{Min}_R(x,y) \le y \\ y & \text{si } \text{Min}_R(x,y) > y \end{cases}$$

**Construcción de $\mathbf{R^{Prod\text{-}cond}}$**
Si T es igual al producto se busca cuando Prod(x, R(x,y))es mayor que y para entonces construir $J^{Prod}(x,y)$ como el Sup $\{z \mid \text{Prod}(x,z)\le y\}$. La igualdad se verifica cuando z=y/x. Por lo tanto se obtiene la siguiente construcción:

$$\mathbf{R}^{\mathbf{Prod\text{-}cond}}(x,y)=\begin{cases} R(x,y) \text{ si Prod}(x,R(x,y)) \le y \\ y/x \text{ si Prod}(x,R(x,y)) > y \end{cases}$$

**Construcción de $\mathbf{R^{Lukasw\text{-}cond}}$**
Si la norma triangular elegida es la Lukasiewcz se analiza en que region Lukasw(x, R(x,y)) (=W(x, R(x,y)) es mayor que y para construir $J^{Lukasw}(x,y)$ como el Sup $\{z \mid \text{Lukasw}(x,z)\le y\}$. La igualdad se verifica cuando z=y-x+1. Por lo tanto se obtiene la siguiente construcción:

$$\mathbf{R}^{\mathbf{lukasw\text{-}cond}}(x,y)=\begin{cases} R(x,y) & \text{si } W(x,R(x,y)) \le y \\ y-x+1 & \text{si } W(x,R(x,y)) > y \end{cases}$$

## 4 Distancia de incondicionalidad

Se define la siguiente **distancia de incondicionalidad** entre la relación R dada y la relación $R^{T\text{-cond}}$ construida. Esto supone definir una medida de incondicionalidad sobre la diferencia entre la relación dada y la relación condicionalizada construida:

**Definición 2:**
Dada una relación difusa R, una norma triangular T construimos su relación condicional asociada $R^{T\text{-cond}}$. Se denomina **distancia de T-incondicionalidad de distancia mínima** $M^{no\text{-}T\text{-}cond}$ a la distancia definida por:

$$\mathbf{M^{no\text{-}T\text{-}cond}}=\int \int_{T_R(x,y)\ge y} (R(x,y)-J^{T}(x,y))dxdy$$

En particular si la norma triangular es el mínimo se tiene la siguiente
**Definición 3:**
Dada una relación difusa R, se denomina **distancia de Min-incondicionalidad** $\mathbf{M^{no\text{-}Min\text{-}cond}}$ a la distancia definida por:

$$\mathbf{M^{no\text{-}Min\text{-}cond}}=\int \int_{T_R(x,y)\ge y} (R(x,y)-y)\,dx\,dy$$

**Definición 4:**
Se definen la siguiente **distancia de condicionalidad**:
$\mathbf{M_R^{T\text{-}cond} = 1 - M_R^{no\text{-}T\text{-}cond}}$

**Observaciones:**
Si $x\le y$ entonces $T_R(x,y) = T(x, R(x,y)) \le T(y,R(x,y)) \le y$, por lo tanto la posible región de integración debe estar incluida en la región x>y. Por consiguiente se observa que:
$\mathbf{M_R^{no\text{-}T\text{-}cond} \le M_R^{no\text{-}Min\text{-}cond} \le 0.5}$ luego $\mathbf{0.5 \le M_R^{Min\text{-}cond} \le M_R^{T\text{-}cond}}$

## 5 Medida de incondicionalidad sobre la verificación del Modus Ponens

Se definen las siguientes medidas de incondicionalidad sobre las diferencias entre T(x,R(x,y)) e y cuando T(x,R(x,y))≥ y:

**Definición 5:**
Dada una relación difusa R y una norma triangular T, se denomina **medida de T-incondicionalidad sobre la verificación del Modus Ponens** a:

$$M_P = \int_{T_R(x,y)>y} \int_{0\le y<x\le 1} T_R(x,y) - T(x, J^T(x,y))\,dydx$$

$$= \int_{T_R(x,y)>y} \int_{0\le y<x\le 1} T_R(x,y) - y \quad dydx$$

Observación:
Sea $z=T_R(x,y)$ una superficie. Sabemos que $T_R(x,y)=T(x,R(x,y))\le x$, luego la integración de esta medida están entre los regiones $z \ge y$ y $z \le x$. Por tanto una cota superior de $M_P$ será

$$\int_0^1 \int_0^x (x-y)\,dx\,dy = \frac{1}{6}$$

Por esta razón para normalizar la medida se define **la medida de incondicionalidad normalizada sobre la verificación del Modus Ponens** asi como la medida de condicionalidad dual (para que el intervalo de definición sea [0, 1]).

**Definición 6:**

$$M_{MP}^{\text{Tincond}} = 6 \int_{T_R(x,y)>y} \int_{0\le y<x\le 1} T_R(x,y) - T(x, J^T(x,y))$$

$$=6 \int_{T_R(x,y)>y} \int_{0\le y<x\le 1} T_R(x,by) - y$$

$$M_{MP}^{\text{T-cond}} = 1 - M_{MP}^{\text{T-incond}}$$

Ambas medidas están en el intervalo [0,1].

## 6 Ejemplos

Se estudian algunas relaciones, buscando cuales son las regiones de su dominio donde hay incondicionalidad para integrar en dicha region. Al mismo tiempo se estudia gráficamente la condicionalidad.

### 6.1 LUKASIEWICZ(x,y)=Min(1,1-x+y)

Se representa gráficamente el dominio de la relación (fig 1.1), $T_{Min}$ (fig 1.2), $T_{Prod}$ (fig 1.3).
Las regiones coloreadas son los lugares donde $T_R$ es mayor que y, es decir, las regiones de incondicionalidad.

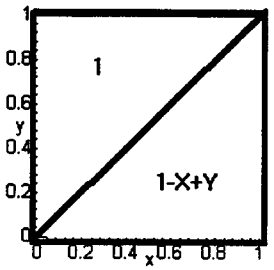



Fig 1.1

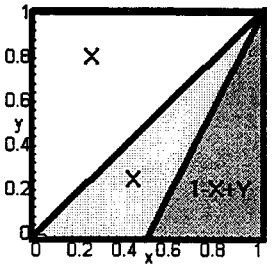



Fig 1.2

$$M^{\text{no-Min-Cond}} = \int_0^1 \int_0^x (1-x)dydx = \frac{1}{6} = 0.16666$$

La medida $M_p^{Min}$ es

$$\int_0^1 \int_y^{\frac{1+y}{2}} (x-y)dxdy + \int_{\frac{1}{2}}^1 \int_0^{2x-1} (1-x)dydx = \frac{1}{24} = 0.04166$$

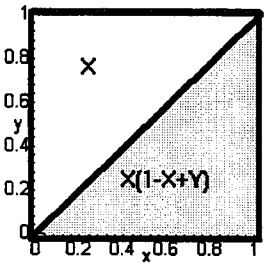



Fig 1.3

$$M^{\text{no-Prod-Cond}} = \int_0^1 \int_0^x (1-x+y-y/x)dydx =$$

$$= \frac{1}{12} = 0.08333$$

$$\mathbf{M_p^{Prod}} = \int_0^1 \int_0^x (x - x^2 + xy - y)dydx = \frac{1}{24} = \mathbf{0.04166}$$

**6.2 KLEENE-DIENES(x,y)=Max(1-x,y)**

La relación sobre su dominio viene definida por la figura 2.1.1

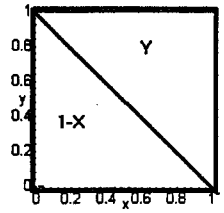



Fig 2.1.1

La relación se visualiza en el espacio en la figura 2.1.2

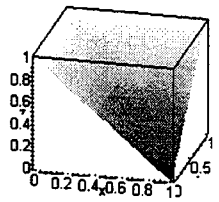



Fig 2.1.2

$T_{Min}$ viene representado sobre su dominio por la figura 2.2.1, en la que encontramos dos regiones de incondicionalidad distintas en las que $T_{Min}$ es mayor que y.

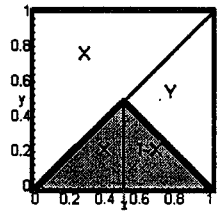



Fig 2.2.1

$T_{Min}$ viene representada en el espacio en la figura 2.2.2.

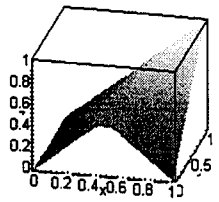



Fig 2.2.2

Se aprecia visualmente que $T_{min}$ no esta siempre por debajo del plano z=y.Es facil encontrar analíticamente la región de Min-incondicionalidad.

Las medidas de Min-incondicionalidad son

$$\mathbf{M^{no\text{-}Min\text{-}Cond}} = \int_0^{\frac{1}{2}} \int_y^{1-y} (1 - x - y)dxdy = \mathbf{0.08333}$$

$$\mathbf{M_p^{Min}} = \int_0^{\frac{1}{2}} \int_0^x (x - y)dydx + \int_{\frac{1}{2}}^1 \int_0^{1-x} (1 - x - y)dydx =$$

$$= \mathbf{0.05666}$$

$T_{Prod}$ viene representado sobre su dominio por la figura 2.3.1.

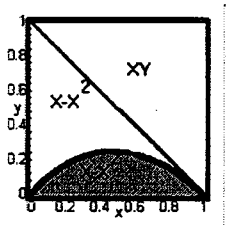



Fig 2.3.1

$T_{Prod}$ viene representada en el espacio en la figura 2.3.2.

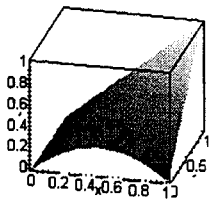



Fig 2.3.2

Se aprecia visualmente que $T_{Prod}$ no esta siempre por debajo del plano z=y.Es facil encontrar analíticamente la región de Min-incondicionalidad.

Las medidas de Prod-incondicionalidad son

$$\mathbf{M^{no\text{-}ProdCond}} = \int_0^1 \int_0^{x-x^2} (1 - x - y / x)dydx = \frac{1}{24} = \mathbf{0.041666}$$

$$\mathbf{M_p^{Prod}} = \int_0^1 \int_0^{x-x^2} (x - x^2 - y)dydx = \frac{1}{60} = \mathbf{0.01666}$$

Kleenes-Dienes es Lukasiewicz-condicional, pues $T_W$ siempre esta por debajo del plano z=y.

Observación:

Las regiones de incondicionalidad y las medidas de incondicionalidad de Kleenes-Dienes coincide con las de las siguientes relaciones:

**EARLY-ZADEH(x,y)=Max(1-x,Min(x,y))**

**WILLMOTT(x,y)=Min[Max(1-x,y),Max(x,1-x),Max(y,1-y)]**

$$\mathbf{KLIR\&YUAN2(x,y)=}\begin{cases} y & x = 1 \\ 1 - x & x \neq 1, y \neq 1 \\ 1 & x \neq 1, y = 1 \end{cases}$$

6.3 **GOUGIN(x,y)**= $\begin{cases} 1 & \text{si } x \leq y \\ y/x & \text{si } x > y \end{cases}$

La relación sobre su dominio viene definida por la figura 3.1

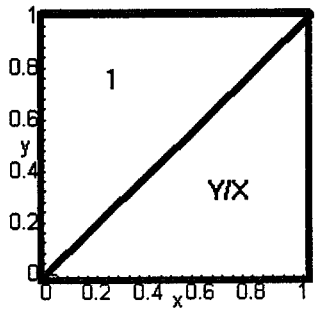



Fig 3.1.

$T_{Min}$ viene representado sobre su dominio por la figura 3.2., en la que encontramos dos regiones de incondicionalidad distintas en las que $T_{Min}$ es mayor que y.

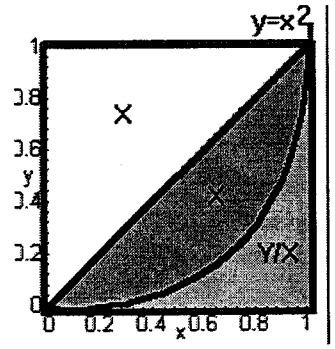



Fig 3.2.

Las medidas de Min-incondicionalidad son

$$\mathbf{M^{no\text{-}Min\text{-}cond}} = \int_0^1 \int_0^x \left(\frac{y}{x} - y\right) dy dx = \frac{1}{12} = \mathbf{0.08333}$$

$$\mathbf{M_p^{Min}} = \int_0^1 \int_{x^2}^x (x - y) dy dx + \int_0^1 \int_0^{x^2} \left(\frac{y}{x} - y\right) dy dx =$$

$$= \frac{1}{60} + \frac{1}{40} = \mathbf{0.041666}$$

Es fácil comprobar que $T_{Prod}$ es siempre menor que y luego Gougin es Prod-condicional, y por lo tanto, también es Lukasiewicz-condicional.

**Agradecimientos**

Agradezco la ayuda y colaboracion del equipo de investigación de Enric Trillas, de la Facultad de Informática, U.P. de Madrid.

**Referencias**

[1] E. Trillas. On Logic and Fuzzy Logic. International Journal of Uncertainty, Fuzzyness and Knoledge-Based Systems, 1/2:107-137,1993.

[2] D.Dubois and H. Prade. Fuzzy Sets and Systems Theory and its Applications. Academic Press, New York,1980.

[3] E. Trillas and S. Cubillo. Modus Ponens on Boolean Algebras Revisited. Mathware & Soft Computing, III-1/2: 105-112, 1996.

[4] G.J. Klir and Bo Yuan. Fuzzy Sets and Fuzzy Logic.Prentice Hall, NJ,1995.

[5] E. Trillas and C. Alsina. Logic: Going furder from Tarski?. Fuzzy Sets and Systems 53 (1993), 1-13.

# LA $\vec{\lambda}$-DESVIACIÓN CUADRÁTICA MEDIA PARA VARIABLES ALEATORIAS DIFUSAS EN POBLACIONES FINITAS

**Asunción Lubiano**
Dpto. de Estadística, I.O. y D.M.
Universidad de Oviedo
33071 Oviedo
lubiano@pinon.ccu.uniovi.es

**Ana Colubi**
Dpto. de Estadística, I.O. y D.M.
Universidad de Oviedo
33071 Oviedo
colubi@pinon.ccu.uniovi.es

**María Angeles Gil**
Dpto. de Estadística, I.O. y D.M.
Universidad de Oviedo
33071 Oviedo
angeles@pinon.ccu.uniovi.es

## Resumen

En un trabajo reciente sobre la definición de distancias entre números difusos, Bertoluzza y otros han presentado una métrica que consiste en una media ponderada de las distancias entre los $\alpha$-cortes de esos números y se basa en los extremos y puntos medios de tales $\alpha$-cortes. En este trabajo va a introducirse una medida con valores reales de la desviación asociada a una variable aleatoria difusa respecto a un número difuso prefijado en una población finita. Se analizarán las propiedades fundamentales de esta medida y se ilustrará su cálculo mediante varios ejemplos.

**Palabras Clave:** distancia entre números difusos, distancia entre intervalos acotados reales, variable aleatoria difusa.

## 1 INTRODUCCIÓN

En ciertos estudios sobre el análisis de regresión difusa, Bertoluzza *et al.* [1] (ver también Bertoluzza *et al.* [2]) han tratado el problema de encontrar una distancia entre números difusos que reemplace la distancia euclídea en $\mathbb{R}$, $d(x,y) = \sqrt{(x-y)^2}$. Para ello, se ha partido de una distancia entre intervalos acotados reales, que evita posibles inconvenientes de las métricas más usuales y generaliza algunas de éstas. Sobre la base de esta distancia, se puede definir para cada par de números difusos la función medible en $(0,1]$ que a cada $\alpha \in (0,1]$ le asocia la distancia entre sus $\alpha$-cortes. La norma en $L^2$ de esta última función será la distancia entre los números difusos considerados.

Si se tiene una población finita y una variable con valores difusos que puede formalizarse por medio de una variable aleatoria difusa (Puri and Ralescu [4]), es posible construir una medida de desviación asociada a esa variable que tome a su vez valores difusos (como la medida definida por Kruse and Meyer [3]).

Sin embargo, las medidas de desviación no siempre tienen por objetivo servir de medida descriptiva resumen de una variable. A menudo interesa considerar medidas de desviación en estudios comparativos entre variables o poblaciones, especialmente cuando se emplean para cuantificar el error asociado a la estimación de un valor difuso mediante otro, y en tales casos la valoración de la desviación debe ser real si quiere garantizarse la viabilidad de dicha comparación.

En este trabajo, y a partir de la distancia entre números difusos de Bertoluzza y otros, va a establecerse una medida de desviación real entre una variable aleatoria difusa y un número difuso prefijado. Esta medida será una extensión de la desviación cuadrática media de una variable aleatoria clásica respecto a un valor real prefijado, y se comprobará que las propiedades de esta última van a conservarse, así como buena parte de las aplicaciones de la misma.

## 2 LA $\vec{\lambda}$-DESVIACIÓN CUADRÁTICA MEDIA: DEFINICIÓN Y PROPIEDADES

Supongamos que se considera una población finita con $N$ individuos (que se suponen equiprobables), $\Omega = \{\omega_1, \ldots, \omega_N\}$, y un espacio medible $(\Omega, \mathcal{A})$ asociado a ella. Sea $\mathcal{X} : \Omega \to \mathcal{F}_c(\mathbb{R})$ una variable aleatoria difusa, donde $\mathcal{F}_c(\mathbb{R})$ representa la clase de los conjuntos difusos $\widetilde{V}$ de $\mathbb{R}$ tal que para todo $\alpha \in (0,1]$ el $\alpha$-corte, $\widetilde{V}_\alpha \in \mathcal{K}_c(\mathbb{R})$ y $\widetilde{V_0} = cl\left[co\left(sop\,\widetilde{V}\right)\right] \in \mathcal{K}_c(\mathbb{R})$, con $\mathcal{K}_c(\mathbb{R})$ la clase de los subconjuntos compactos convexos y no vacíos de $\mathbb{R}$.

**Definición 2.1** *La $\vec{\lambda}$-desviación cuadrática media asociada a la variable aleatoria difusa $\mathcal{X}$ en la población con respecto al valor $\widetilde{A} \in \mathcal{F}_c(\mathbb{R})$ se define*

*como el valor real:*

$$DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)=\frac{1}{N}\sum_{j=1}^{N}\left[D_{\vec{\lambda}}\left(\mathcal{X}(\omega_j),\widetilde{A}\right)\right]^2,$$

*donde* $D_{\vec{\lambda}}\left(\widetilde{B},\widetilde{A}\right)$ *es la métrica en* $\mathcal{F}_c(\mathbb{R})$ *introducida por Bertoluzza et al.* [1], *es decir:*

$$D_{\vec{\lambda}}\left(\widetilde{B},\widetilde{A}\right)=\sqrt{\int_{(0,1]}\left[d_{\vec{\lambda}}\left(\widetilde{B}_\alpha,\widetilde{A}_\alpha\right)\right]^2 d\alpha}$$

*y*

$$d_{\vec{\lambda}}\left(\widetilde{B}_\alpha,\widetilde{A}\alpha\right)=\Big[\lambda_1\left(\sup\widetilde{B}_\alpha-\sup\widetilde{A}_\alpha\right)^2+$$

$$\lambda_2\left(\operatorname{med}\widetilde{B}_\alpha-\operatorname{med}\widetilde{A}_\alpha\right)^2+\lambda_3\left(\inf\widetilde{B}_\alpha-\inf\widetilde{A}_\alpha\right)^2\Big]^{1/2}$$

*con* $\vec{\lambda}=(\lambda_1,\lambda_2,\lambda_3)$, $\lambda_i\in[0,1)$, $i=1,2,3$, *y* $\lambda_1+\lambda_2+\lambda_3=1$ *y* med *el punto medio del intervalo correspondiente.*

A continuación van a examinarse algunas de las propiedades fundamentales de la medida de desviación anterior, supuesto fijado el vector $\vec{\lambda}$.

En primer lugar, va a verificarse que $DCM_{\vec{\lambda}}$ extiende al caso difuso el conocido momento central de segundo orden de una variable aleatoria clásica con respecto a un valor real, para lo cual a lo largo del trabajo se identificará cada valor $x\in\mathbb{R}$ con la función indicador del conjunto $\{x\}$.

**Proposición 2.1** *Si* $\mathcal{X}(\Omega)\subset\mathbb{R}$ *y* $\widetilde{A}\in\mathbb{R}$, *se cumple que:*

$$DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)=E\left[\left(\mathcal{X}-\widetilde{A}\right)^2\right],$$

*donde E representa el valor esperado de una variable aleatoria clásica.*

*Demostración:* En efecto, en las condiciones supuestas se tiene que:

$$D_{\vec{\lambda}}\left(\mathcal{X}(\omega_j),\widetilde{A}\right)=d\left(\mathcal{X}(\omega_j),\widetilde{A}\right),\ j=1,\ldots,N,$$

$$E\left[\left(\mathcal{X}-\widetilde{A}\right)^2\right]=\frac{1}{N}\sum_{j=1}^{N}\left[d\left(\mathcal{X}(\omega_j),\widetilde{A}\right)\right]^2.\quad\square$$

La proposición siguiente confirma que la no negatividad de los momentos de segundo orden se conserva en el caso difuso, y que la anulación de $DCM_{\vec{\lambda}}$ equivale a que la variable aleatoria difusa tome el mismo valor difuso sobre todos los individuos de la población.

**Proposición 2.2** *Cualquiera que sea* $\widetilde{A}\in\mathcal{F}_c(\mathbb{R})$, *se cumple siempre que* $DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)\geq 0$, *con* $DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)=0$ *si, y sólo si,* $\mathcal{X}$ *es una variable aleatoria difusa degenerada en* $\widetilde{A}$.

*Demostración:* En efecto, al estar $DCM_{\vec{\lambda}}$ definida como un promedio de cuadrados de valores reales, $DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)\geq 0$. Además, $DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)=0$ si, y sólo si, $D_{\vec{\lambda}}\left(\mathcal{X}(\omega_j),\widetilde{A}\right)=0$, $j=1,\ldots,N$, debido a la no negatividad de estos valores por ser $D_{\vec{\lambda}}$ una distancia en $\mathcal{F}_c(\mathbb{R})$.

Por otro lado, la última afirmación también garantiza que cualquiera que sea $j\in\{1,\ldots,N\}$, $D_{\vec{\lambda}}\left(\mathcal{X}(\omega_j),\widetilde{A}\right)=0$ si, y sólo si, $\mathcal{X}(\omega_j)=\widetilde{A}$, lo que equivale a que $\mathcal{X}$ sea una variable aleatoria difusa degenerada en $\widetilde{A}$. $\square$

La propiedad que se presenta a continuación, pone de manifiesto que $DCM_{\vec{\lambda}}$ no sólo está relacionada con el momento central de segundo orden para variables aleatorias clásicas respecto a valores reales por ser una extensión del mismo (ver Proposición 2.1), sino también porque alternativamente puede calcularse a partir de los momentos centrales de segundo orden de ciertas variables y valores reales.

**Proposición 2.3** *Cualquiera que sea* $\widetilde{A}\in\mathcal{F}_c(\mathbb{R})$, *se cumple que:*

$$DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)=\int_{(0,1]}\Big\{\lambda_1E\left[\left(\sup\mathcal{X}_\alpha-\sup\widetilde{A}_\alpha\right)^2\right]$$
$$+\lambda_2E\left[\left(\operatorname{med}\mathcal{X}_\alpha-\operatorname{med}\widetilde{A}_\alpha\right)^2\right]$$
$$+\lambda_3E\left[\left(\inf\mathcal{X}_\alpha-\inf\widetilde{A}_\alpha\right)^2\right]\Big\}\,d\alpha$$

*con* $\mathcal{X}_\alpha:\Omega\to\mathcal{K}_c(\mathbb{R})$ *tal que* $\mathcal{X}_\alpha(\omega_j)=(\mathcal{X}(\omega_j))_\alpha$, $j=1,\ldots,N$, *para todo* $\alpha\in[0,1]$.

*Demostración:* En efecto:

$$DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)$$
$$=\frac{1}{N}\sum_{j=1}^{N}\int_{(0,1]}\Big[\lambda_1\left(\sup(\mathcal{X}(\omega_j))_\alpha-\sup\widetilde{A}_\alpha\right)^2$$
$$+\lambda_2\left(\operatorname{med}(\mathcal{X}(\omega_j))_\alpha-\operatorname{med}\widetilde{A}_\alpha\right)^2$$
$$+\lambda_3\left(\inf(\mathcal{X}(\omega_j))_\alpha-\inf\widetilde{A}_\alpha\right)^2\Big]\,d\alpha$$
$$=\int_{(0,1]}\Big[\lambda_1\frac{1}{N}\sum_{j=1}^{N}\left(\sup(\mathcal{X}(\omega_j))_\alpha-\sup\widetilde{A}_\alpha\right)^2$$

$$+\lambda_2 \frac{1}{N}\sum_{j=1}^{N}\left(\operatorname{med}\left(\mathcal{X}(\omega_j)\right)_\alpha - \operatorname{med}\widetilde{A}_\alpha\right)^2$$

$$+\lambda_3 \frac{1}{N}\sum_{j=1}^{N}\left(\inf\left(\mathcal{X}(\omega_j)\right)_\alpha - \inf\widetilde{A}_\alpha\right)^2\Bigg]\, d\alpha. \quad \square$$

La propiedad siguiente establece una conexión entre la $\vec{\lambda}$-desviación cuadrática media de $\mathcal{X}$ respecto a un valor cualquiera en $\mathcal{F}_c(\mathbb{R})$ y la $\vec{\lambda}$-desviación cuadrática media respecto al valor esperado difuso de $\mathcal{X}$ (definido por Puri and Ralescu [4]), que vendría dado en este caso por:

$$\widetilde{E}(\mathcal{X}) = \frac{1}{N}\odot\left[\mathcal{X}(\omega_1)\oplus\ldots\oplus\mathcal{X}(\omega_N)\right],$$

es decir, para todo $\alpha\in[0,1]$:

$$\left(\widetilde{E}(\mathcal{X})\right)_\alpha$$

$$=\left[\frac{1}{N}\sum_{j=1}^{N}\inf\left(\mathcal{X}(\omega_j)\right)_\alpha, \frac{1}{N}\sum_{j=1}^{N}\sup\left(\mathcal{X}(\omega_j)\right)_\alpha\right].$$

La conexión mencionada indica que:

**Proposición 2.4** *Cualquiera que sea $\widetilde{A}\in\mathcal{F}_c(\mathbb{R})$, se cumple que:*

$$DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right) = DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{E}(\mathcal{X})\right) + \left[D_{\vec{\lambda}}\left(\widetilde{E}(\mathcal{X}),\widetilde{A}\right)\right]^2.$$

*Demostración:* En efecto:

$$DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)$$

$$=\frac{1}{N}\sum_{j=1}^{N}\int_{(0,1]}\Bigg[\lambda_1\left(\sup\left(\mathcal{X}(\omega_j)\right)_\alpha - \sup\widetilde{A}_\alpha\right)^2$$

$$+\lambda_2\left(\operatorname{med}\left(\mathcal{X}(\omega_j)\right)_\alpha - \operatorname{med}\widetilde{A}_\alpha\right)^2$$

$$+\lambda_3\left(\inf\left(\mathcal{X}(\omega_j)\right)_\alpha - \inf\widetilde{A}_\alpha\right)^2\Bigg]\, d\alpha$$

$$=\frac{1}{N}\sum_{j=1}^{N}\int_{(0,1]}\Bigg[\lambda_1\left(\sup\left(\ calX(\omega_j)\right)_\alpha - \sup\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)^2$$

$$-2\lambda_1\left(\sup\left(\mathcal{X}(\omega_j)\right)_\alpha - \sup\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)\cdot\left(\sup\left(\widetilde{E}(\mathcal{X})\right)_\alpha - \sup\widetilde{A}_\alpha\right)$$

$$+\lambda_1\left(\sup\left(\widetilde{E}(\mathcal{X})\right)_\alpha - \sup\widetilde{A}_\alpha\right)^2$$

$$+\lambda_2\left(\operatorname{med}\left(\mathcal{X}(\omega_j)\right)_\alpha - \operatorname{med}\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)^2$$

$$-2\lambda_2\left(\operatorname{med}\left(\mathcal{X}(\omega_j)\right)_\alpha - \operatorname{med}\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)\cdot\left(\operatorname{med}\left(\widetilde{E}(\mathcal{X})\right)_\alpha - \operatorname{med}\widetilde{A}_\alpha\right)$$

$$+\lambda_2\left(\operatorname{med}\left(\widetilde{E}(\mathcal{X})\right)_\alpha - \operatorname{med}\widetilde{A}_\alpha\right)^2$$

$$+\lambda_3\left(\inf\left(\mathcal{X}(\omega_j)\right)_\alpha - \inf\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)^2$$

$$-2\lambda_3\left(\inf\left(\mathcal{X}(\omega_j)\right)_\alpha - \inf\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)\cdot\left(\inf\left(\widetilde{E}(\mathcal{X})\right)_\alpha - \inf\widetilde{A}_\alpha\right)$$

$$+\lambda_3\left(\inf\left(\widetilde{E}(\mathcal{X})\right)_\alpha - \inf\widetilde{A}_\alpha\right)^2\Bigg]\, d\alpha$$

$$=\frac{1}{N}\sum_{j=1}^{N}\int_{(0,1]}\Bigg[\lambda_1\left(\sup\left(\mathcal{X}(\omega_j)\right)_\alpha - \sup\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)^2$$

$$+\lambda_2\left(\operatorname{med}\left(\mathcal{X}(\omega_j)\right)_\alpha - \operatorname{med}\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)^2$$

$$+\lambda_3\left(\inf\left(\mathcal{X}(\omega_j)\right)_\alpha - \inf\left(\widetilde{E}(\mathcal{X})\right)_\alpha\right)^2\Bigg]\, d\alpha$$

$$+\left[D_{\vec{\lambda}}\left(\widetilde{E}(\mathcal{X}),\widetilde{A}\right)\right]^2. \quad \square$$

A partir de la Proposición 2.4, se deriva como consecuencia inmediata que $DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)$ alcanza su valor mínimo con respecto a $\widetilde{A}\in\mathcal{F}_c(\mathbb{R})$ en el valor esperado difuso de $\mathcal{X}$. Así:

**Proposición 2.5** *La función $G_{\mathcal{X}}:\mathcal{F}_c(\mathbb{R})\to[0,+\infty)$ tal que*

$$G_{\mathcal{X}}(\widetilde{A}) = DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)$$

*alcanza su valor mínimo en $\mathcal{F}_c(\mathbb{R})$ para $\widetilde{A}=\widetilde{E}(\mathcal{X})$.*

*Demostración:* En efecto, de acuerdo con la Proposición 2.4:

$$DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{A}\right)\geq DCM_{\vec{\lambda}}\left(\mathcal{X},\widetilde{E}(\mathcal{X})\right),$$

con igualdad si, y sólo si, $D_{\vec{\lambda}}\left(\widetilde{E}(\mathcal{X}),\widetilde{A}\right)=0$ que, al ser $D_{\vec{\lambda}}$ distancia en $\mathcal{F}_c(\mathbb{R})$, se cumple si, y sólo si, $\widetilde{A}=\widetilde{E}(\mathcal{X})$. $\square$

## 3 EJEMPLOS ILUSTRATIVOS

Cuando se quiere calificar un día de acuerdo con su temperatura, es habitual recurrir a adjetivos como FRIO, FRESCO, NORMAL, CALIDO o CALIENTE. Según esa calificación, el tipo de día podría considerarse como una variable aleatoria difusa, $\mathcal{X}$, cuyos valores serían los cinco adjetivos precedentes, y una posible aproximación de los mismos a lo largo del verano podría llevarse a cabo por medio de números difusos triangulares y trapezoidales con soportes contenidos en $[8, 40]$ (en $^\circ$ C), como los siguientes: FRIO $= Tra(8, 8, 14, 19)$, FRESCO $= Tri(14, 19, 24)$, NORMAL $= Tri(19, 24, 29)$, CALIDO $= Tri(24, 29, 34)$ y CALIENTE $= Tra(29, 34, 40, 40)$ (Figura 1).

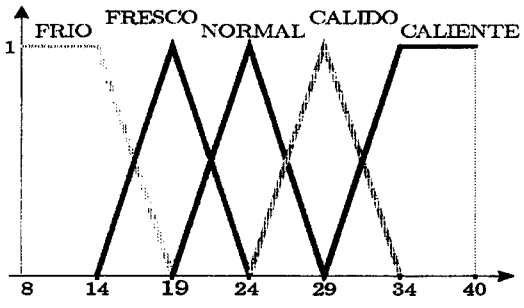



Figura 1: Valores difusos del tipo de día.

En los ejemplos siguientes se analiza la $\vec{\lambda_0}$-desviación cuadrática media asociada a $\mathcal{X}$ respecto a distintos valores y en diferentes poblaciones, con $\vec{\lambda_0} = (0.5, 0, 0.5)$.

**Ejemplo 3.1.** En este primer ejemplo la población considerada es la de los 31 días del mes de Julio de cierto año y en cierta zona ($\Omega_1$), supuesto que a lo largo de ese mes 2 días han sido FRIOS, 4 han sido FRESCOS, 7 NORMALES, 17 CALIDOS y 1 CALIENTE. Sea desea estudiar la $\vec{\lambda_0}$-desviación cuadrática media asociada a $\mathcal{X}$ con respecto a $\widetilde{A}_1 = \widetilde{E}(\mathcal{X})$, $\widetilde{A}_2 = 24$ y $\widetilde{A}_3 =$ NORMAL.

Como $\left(\widetilde{E}(\mathcal{X})\right)_\alpha = [20.71 + 4.68\alpha, 30.81 - 4.84\alpha]$ para todo $\alpha \in [0, 1]$, es decir, $\widetilde{E}(\mathcal{X}) = Tra(20.71, 25.39, 25.97, 30.81)$, se tiene que:

$$\left[D_{\vec{\lambda_0}}\left(\text{FRIO}, \widetilde{A}_1\right)\right]^2 = 184.85,$$

$$\left[D_{\vec{\lambda_0}}\left(\text{FRESCO}, \widetilde{A}_1\right)\right]^2 = 45.19,$$

$$\left[D_{\vec{\lambda_0}}\left(\text{NORMAL}, \widetilde{A}_1\right)\right]^2 = 2.99,$$

$$\left[D_{\vec{\lambda_0}}\left(\text{CALIDO}, \widetilde{A}_1\right)\right]^2 = 10.79,$$

$$\left[D_{\vec{\lambda_0}}\left(\text{CALIENTE}, \widetilde{A}_1\right)\right]^2 = 104.08.$$

En consecuencia:

$$DCM_{\vec{\lambda_0}}\left(\mathcal{X}, \widetilde{A}_1\right) = 27.71.$$

Si se aplica la Proposición 2.4, las otras dos desviaciones pueden hallarse sin más que calcular

$$\left[D_{\vec{\lambda_0}}\left(\widetilde{A}_1, \widetilde{A}_2\right)\right]^2 = 11.54, \quad \left[D_{\vec{\lambda_0}}\left(\widetilde{A}_1, \widetilde{A}_3\right)\right]^2 = 2.99,$$

puesto que:

$$DCM_{\vec{\lambda_0}}\left(\mathcal{X}, \widetilde{A}_2\right) = DCM_{\vec{\lambda_0}}\left(\mathcal{X}, \widetilde{A}_1\right) + \left[D_{\vec{\lambda_0}}\left(\widetilde{A}_1, \widetilde{A}_2\right)\right]^2 = 39.22,$$

$$DCM_{\vec{\lambda_0}}\left(\mathcal{X}, \widetilde{A}_3\right) = DCM_{\vec{\lambda_0}}\left(\mathcal{X}, \widetilde{A}_1\right) + \left[D_{\vec{\lambda_0}}\left(\widetilde{A}_1, \widetilde{A}_3\right)\right]^2 = 30.7.$$

**Ejemplo 3.2.** En este segundo ejemplo se examina la comparación, a través de la $\vec{\lambda_0}$-desviación cuadrática media asociada a $\mathcal{X}$ con respecto a $\widetilde{E}(\mathcal{X})$, de la distribución de los tipos de días en el mes de Julio del año y en la zona considerados en el Ejemplo 3.1 con una distribución hipotética "uniforme" en un mes de 30 días (población que denotaremos por $\Omega_2$), de forma que 6 días de ese mes habrían sido FRIOS, 6 FRESCOS, 6 NORMALES, 6 CALIDOS y 6 CALIENTES.

En este caso, procediendo como en el Ejemplo 3.1, se obtendría que en $\Omega_2$ el valor esperado de $\mathcal{X}$ viene dado por $\widetilde{E}(\mathcal{X}) = Tra(18.8, 22.8, 25.2, 29.2)$ y que:

$$DCM_{\vec{\lambda_0}}\left(\mathcal{X}, \widetilde{E}(\mathcal{X})\right) = 66.29,$$

de modo que $\mathcal{X}$ resultaría sensiblemente menos desviada respecto a su valor esperado en $\Omega_1$ que en $\Omega_2$.

**Ejemplo 3.3.** En este tercer ejemplo se estudia la comparación, a través de la $\vec{\lambda_0}$-desviación cuadrática media asociada a $\mathcal{X}$ con respecto a $\widetilde{E}(\mathcal{X})$, de la distribución de los tipos de días en el mes de Julio del año y en la zona considerados en el Ejemplo 3.1 con la distribución de los tipos de días en el mes de Agosto del año y en la zona del Ejemplo 3.1 (población $\Omega_3$), supuesto que a lo largo de ese Agosto ningún día ha sido FRIO, 3 han sido FRESCOS, 6 NORMALES, 12 CALIDOS y 10 CALIENTES.

Siguiendo un desarrollo similar al del Ejemplo 3.1, se concluiría que en $\Omega_3$ el valor esperado de $\mathcal{X}$ viene dado por $\widetilde{E}(\mathcal{X}) = Tra(23.68, 26.68, 30.61, 34)$ y que:

$$DCM_{\vec{\lambda_0}}\left(\mathcal{X}, \widetilde{E}(\mathcal{X})\right) = 30.05,$$

de modo que $\mathcal{X}$ resultaría ligeramente menos desviada respecto a su valor esperado en Julio que en Agosto.

## 4 CONCLUSIONES FINALES

La $\vec{\lambda}$-desviación cuadrática media ha sido empleada en estudios de regresión difusa (ver Bertoluzza *et al.* [2]), aunque las propiedades de esta medida no se habían examinado con anterioridad.

Existen varios problemas abiertos cuya investigación está en curso. Entre ellos cabe destacar: su estudio para variables aleatorias difusas en poblaciones generales; la definición de la extensión (con valores reales) de la varianza para variables aleatorias difusas; la definición de una extensión (también con valores reales) del error cuadrático medio asociado a un estimador difuso de un parámetro difuso.

**Agradecimientos**

La investigación en este trabajo se ha financiado en parte mediante el Proyecto DGES No. PB95-1049 del Ministerio de Educación y Cultura, al que deseamos poner de manifiesto nuestro agradecimiento.

## Referencias

[1] C. Bertoluzza, N. Corral and A. Salas (1995). On a new class of distances between fuzzy numbers. *Mathware & Soft Computing* **2** 71-84.

[2] C. Bertoluzza, N. Corral and A. Salas (1995). Polinomial regression in a fuzzy context. The least squares method, *Proc. 6th IFSA Conference*, 431-434.

[3] R. Kruse and K. Meyer (1987). *Statistics with Vague Data.* Reidel Publ. Co., Dordrecht.

[4] M.L. Puri and D.A. Ralescu (1986). Fuzzy random variables. *J. Math. Anal. Appl.* **114**, 409-422.

# COMPLETITUD Y DIRECCION EN CIERTOS SISTEMAS RELACIONALES DIFUSOS

**P. Burillo, R. Fuentes-González, L. González**
Departamento de Automática y Computación. Universidad Pública de Navarra
Campus de Arrosadía s/n. 31006. Pamplona. Navarra.
E-mail: *rfuentes@si.upna.es*

## Resumen

Los conceptos de acotación, completitud y conjunto dirigido, habitualmente asociados a conjuntos parcialmente ordenados (X, ≤), se extienden a clases de sistemas relacionales difusos $(X, \tilde{R})$ de modo coherente. Se estudian propiedades de estas generalizaciones y se presentan ejemplos.

**Palabras Clave:** Grados de completitud y dirección en sistemas relacionales difusos. Implicación material difusa. Orden difuso.

## 1 INTRODUCCIÓN

Dado un referencial X y una relación difusa reflexiva $\tilde{R} \in [0,1]^{X\times X}$, en [2] se extendieron a clases de subconjuntos difusos ciertos conceptos distinguidos ligados al orden $\leq_X$ de un conjunto parcialmente ordenado $(X, \leq_X)$. Concretamente, los conceptos de *mayorante*, *maximal*, *máximo*, *supremo*, (y sus duales respectivos), de subconjuntos ordinarios $M \subseteq X$ a subconjuntos difusos $\tilde{A} \in [0,1]^X$. Las extensiones de mayorante y su dual aparecen como operadores sobre subconjuntos difusos y se basan en una versión del operador triángulo $\triangleleft$ de Kohout y Bandler [3] dado por la expresión:

*(BK)* $(\forall y \in X)\ ((\tilde{A} \triangleleft \tilde{R})(y) = inf\{\tilde{A}(x) \rightarrow \tilde{R}(x,y) / x \in X\})$,

donde "$\rightarrow$" representa una *implicación material difusa* [6], [7],[8] (como las de *Kleene-Dienes* y de *Lukasiewicz*), que como es conocido verifica:

$$1 \rightarrow 0 = 0; \qquad \alpha \rightarrow 1 = 1 \quad \forall \alpha \in [0,1].$$

En el trabajo citado [2], se asocia a un subconjunto difuso $\tilde{A} \in [0,1]^X$ los nuevos subconjuntos difusos:

$$(i)\ \tilde{A}^u = \tilde{A} \triangleleft \tilde{R} \quad (MAYORANTE)$$
$$\tilde{A}^l = \tilde{A} \triangleleft \tilde{R}^{op} \quad (MINORANTE),$$

donde $\tilde{R}^{op}(y,x) = \tilde{R}(x,y)\ \forall (x,y) \in X\times X$ representa la relación difusa opuesta de $\tilde{R}$. Después, utilizando estos operadores $^u$ y $^l$, que proporcionan *cotas difusas* en $(X, \tilde{R})$, y una *t-norma* cualquiera $\top$ [4] en [0,1], se introducen en [2] los nuevos conceptos asociados al subconjunto difuso $\tilde{A}$ dados por:

$$(ii)\quad (\top\text{-}MAX)\tilde{A} = \tilde{A} \top \tilde{A}^u \quad , \quad (\top\text{-}MIN)\tilde{A} = \tilde{A} \top \tilde{A}^l ,$$
$$(iii)\quad (\top\text{-}MAXIMAL)\tilde{A} = \tilde{A} \top (\tilde{A} \triangleleft (Clr(\tilde{R}'))^{op})$$
$$(\top\text{-}MINIMAL)\tilde{A} = \tilde{A} \top (\tilde{A} \triangleleft Clr(\tilde{R}')) .$$
$$(iv)\quad (\top\text{-}SUP)\tilde{A} = \tilde{A}^u \top (\tilde{A}^u)^l \ , \ (\top\text{-}INF)\tilde{A} = \tilde{A}^l \top (\tilde{A}^l)^u$$

Expresiones en las que $\tilde{A} \top \tilde{B}$ representa la versión generalizada de la intersección difusa asociada a la *t-norma* $\top$ [5], es decir $(\tilde{A} \top \tilde{B})(x) = \tilde{A}(x) \top \tilde{B}(x)\ \forall x \in X$, $\tilde{R}'$ es la relación asociada a $\tilde{R}$ y a la implicación "$\rightarrow$" tal que: $\tilde{R}'(x,y) = \tilde{R}(x,y) \rightarrow 0$ para $(x,y) \in X^2$ y $Clr(\tilde{R}')$ representa la *clausura reflexiva* de $\tilde{R}'$.

Se prueba en el trabajo citado que en el sistema relacional $(X, \tilde{R})$ todos estos nuevos conceptos juegan, en relación al subconjunto difuso $\tilde{A}$, un papel análogo al que desempeñan los del mismo nombre en un referencial parcialmente ordenado $(X, \leq_X)$. Se recuperan además en las definiciones dadas estos conceptos clásicos cuando $\tilde{R}$ es la función característica del orden $\leq_X$ en X y $\tilde{A}$ es un nítido.

Por otra parte, dada una relación reflexiva $\tilde{R}$, se dice que es *$(\rightarrow)$-antisimétrica* [2] si:

$$(v)\quad \tilde{R}^{op} \tilde{\subset} Clr(\tilde{R}') ,$$

($\tilde{\subset}$ representa la inclusión difusa usual definida puntualmente a partir del orden ≤ en [0,1]). Se verifica

que toda relación $\tilde{R}$ reflexiva y antisimétrica en el sentido de Zadeh [9], es ($\rightarrow$)-antisimétrica según *(v)*.

En este trabajo, considerando un sistema relacional difuso $(X, \tilde{R})$, y utilizando los operadores $^u$ , *MAX*, *SUP*, y sus duales mencionados en los párrafos anteriores, se extienden conceptos clásicos de la teoría de conjuntos parcialmente ordenados: *acotación, completitud, dirigido* y *filtrado* a ciertas clases de subconjuntos difusos.

Aunque la mayoría de los conceptos dados aquí son válidos para relaciones difusas en general, estamos especialmente interesados en el caso en que $\tilde{R}$ es una relación difusa reflexiva y tiene algún grado de "antisimetría". Por estas razones en lo que sigue consideraremos:

*(a)* Una implicación material difusa "$\rightarrow$" en [0,1].
*(b)* Una *t*-norma difusa $\top$ en [0,1].
*(c)* Un referencial ordinario $X \neq \emptyset$ y subconjuntos difusos $\tilde{A}$ de X.
*(d)* Una relación binaria difusa $\tilde{R}$ en X que sea reflexiva y ($\rightarrow$)-antisimétrica, (en el sentido de la definición *(v)*).
*(e)* Los conceptos *MAYORANTE MAX, SUP*, etc. dados por las expresiones *(i)*, *(ii)* y *(iv)* en función de los operadores $\triangleleft$, $\top$ asociados a $\rightarrow$ y $\top$.

## 2 EJEMPLOS

Veamos algunos ejemplos para centrar la discusión.

**2.1**- En [2] se utiliza el siguiente ejemplo de cálculo de $\tilde{A}^u$, *MAX*$\tilde{A}$, *MAXIMAL*$\tilde{A}$, *SUP*$\tilde{A}$ y sus duales.
En el referencial $X = \{x_1,x_2,x_3,x_4,x_5,x_6\}$ consideramos la relación binaria difusa $\tilde{R}$ determinada por la tabla:

Tabla 1

| $\tilde{R}$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ |
|---|---|---|---|---|---|---|
| $x_1$ | 1 | 0.8 | 0.2 | 0.6 | 0.6 | 0.4 |
| $x_2$ | 0 | 1 | 0 | 0 | 0.6 | 0 |
| $x_3$ | 0 | 0 | 1 | 0 | 0.5 | 0 |
| $x_4$ | 0 | 0 | 0 | 1 | 0.6 | 0.4 |
| $x_5$ | 0 | 0 | 0 | 0 | 1 | 0 |
| $x_6$ | 0 | 0 | 0 | 0 | 0 | 1 |

relación que además de reflexiva resulta ser transitiva y antisimétrica perfecta (por tanto ($\rightarrow$)-antisimétrica para toda implicación "$\rightarrow$") , y que Zadeh propone como ejemplo de *orden difuso* [9]. Se puede representar esta relación mediante el grafo:

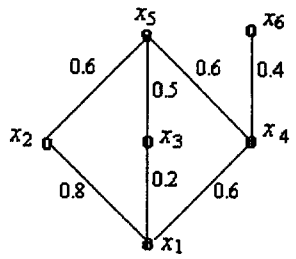



Figura 1

Consideramos:
***2.1.1***.- La implicación de *Kleene-Dienes*: $\alpha \rightarrow_K \beta = max(1-\alpha, \beta)$ y la *t-norma* $\top_1 = min$ en [0,1]. Dado el subconjunto difuso $\tilde{A} = 0.4/x_1+1/x_2+0.8/x_4$ de X, y utilizando las expresiones *(BK)*, *(i)* , *(ii)*, *(iii)* y *(iv)* calculamos en [2]:

$$\tilde{A}^u = 0.2/x_2+0.6/x_5 \ , \ \tilde{A}^l = 0.6/x_1+0.2/x_2 \ ,$$
$$(\top_1\text{-}MAX)\tilde{A} = 0.2/x_2 \ , \ (\top_1\text{-}MIN)\tilde{A} = 0.4/x_1+0.2/x_2 \ ,$$
$$(\top_1\text{-}MAXIMAL)\tilde{A} = 0.2/x_1+1/x_2+0.8/x_4 \ ,$$
$$(\top_1\text{-}MINIMAL)\tilde{A} = 0.4/x_1+0.6/x_2+0.6/x_4 \ ,$$
$$(\top_1\text{-}SUP)\tilde{A} = 0.2/x_2+0.6/x_5 \ ,$$
$$(\top_1\text{-}INF)\tilde{A} = 0.6/x_1+0.2/x_2.$$

Es decir, utilizando la relación difusa $\tilde{R}$ que expresa un orden vago en X, y una implicación difusa con la que expresamos "como deducir", se asocia a representaciones de enunciados del tipo: "$x_2$ *es* $\tilde{A}$ , *muy posiblemente* $x_4$ *es* $\tilde{A}$ *y es poco posible que* $x_1$ *sea* $\tilde{A}$ "; otras representaciones análogas de enunciados como "*es muy poco posible que* $x_2$ *sea el máximo de* $\tilde{A}$ " o como "*es posible que* $x_5$ *sea supremo de* $\tilde{A}$ *y muy poco posible que lo sea* $x_2$".

En el caso del nítido $\tilde{1} \in [0,1]^X$, (función característica de X), obtenemos:

$$\tilde{1}^u = \tilde{0} \ , \ \tilde{1}^l = 0.2/x_1 \ ,$$
$$(\top_1\text{-}MAX)\,\tilde{1} = (\top_1\text{-}SUP)\,\tilde{1} = \tilde{0} \ ,$$
$$(\top_1\text{-}MIN)\,\tilde{1} = (\top_1\text{-}INF)\,\tilde{1} = 0.2/x_1 \ ,$$
$$(\top_1\text{-}MAXIMAL)\,\tilde{1} =$$
$$0.2/x_1+0.4/x_2+0.5/x_3+0.4/x_4+1/x_5+1/x_6 \ ,$$
$$(\top_1\text{-}MINIMAL)\,\tilde{1} =$$
$$1/x_1+0.2/x_2+0.8/x_3+0.4/x_4+0.4/x_5+0.6/x_6 \ .$$

Veamos a continuación que estos conceptos varían con la implicación y con la *t*-norma elegidas:

***2.1.2***.- Para la misma relación difusa $\tilde{R}$ de la tabla 1, consideramos ahora la implicación material difusa de *Lukasiewicz*: $\alpha \rightarrow_L \beta = min(1, (1-\alpha+\beta))$ y la *t*-norma en [0,1] definida por $\alpha \top_2 \beta = max(0, (\alpha+\beta-1))$.

Tomando de nuevo el mismo subconjunto difuso $\tilde{A} = 0.4/x_1+1/x_2+0.8/x_4$ y utilizando las definiciones mencionadas en *(i)*, *(ii)*, *(iii)* y *(iv)*, obtenemos los subconjuntos difusos asociados:

$$\tilde{A}^u = 0.2/x_2+0.6/x_5 \ , \ \tilde{A}^l = 0.8/x_1+0.2/x_2 \ ,$$
$$(\top_2\text{-}MAX)\tilde{A} = 0.2/x_2 \ , \ (\top_2\text{-}MIN)\tilde{A} = 0.2/x_1+0.2/x_2 \ ,$$
$$(\top_2\text{-}MAXIMAL)\tilde{A} = 1/x_2+0.8/x_4 \ ,$$
$$(\top_2\text{-}MINIMAL)\tilde{A} = 0.4/x_1+0.8/x_2+0.8/x_4 \ ,$$
$$(\top_2\text{-}SUP)\tilde{A} = 0.2/x_2+0.4/x_5, \ (\top_2\text{-}INF)\tilde{A} = 0.6/x_1+0.2/x_2.$$

Puede observarse que para el nítido $\tilde{1}$, los resultados que se obtienen coinciden con los del apartado *2.1.1.*

**2.2**- Sea el referencial $X = \{x_1,x_2,x_3,x_4,x_5\}$ y consideremos en [0,1] la implicación de Kleene-Dienes y la t-norma $\top_1 = min$. Sea $\tilde{R}_1$ la relación:

Tabla 2

| $\tilde{R}_1$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ |
|---|---|---|---|---|
| $x_1$ | 1 | 0.8 | 0.9 | 1 |
| $x_2$ | 0.2 | 1 | 0.9 | 1 |
| $x_3$ | 0.1 | 0.1 | 1 | 1 |
| $x_4$ | 0 | 0 | 0 | 1 |

que es reflexiva, transitiva y $(\rightarrow_K)$-antisimétrica. Para $\tilde{A} = 0.1/x_1+0.1/x_2+0.9/x_3+1/x_4$ se obtiene:

$$(\top_1\text{-}MAX)\tilde{A} = (\top_1\text{-}MAXIMAL)\tilde{A} = (\top_1\text{-}SUP)\tilde{A} = 1/x_4 \ ,$$
$$(\top_1\text{-}MIN)\tilde{A} = (\top_1\text{-}MINIMAL)\tilde{A} = (\top_1\text{-}INF)\tilde{A} =$$
$$0.1/x_1+0.1/x_2+0.9/x_3+0.1/x_4.$$

# 3 GRADO DE ACOTACIÓN Y GRADO DE COMPLETITUD

Sean X un referencial no vacío, "$\rightarrow$" una implicación material difusa y sea $\tilde{R}$ una relación difusa reflexiva y $(\rightarrow)$-antisimétrica. Las definiciones que proponemos a continuación extienden coherentemente la noción usual de acotación a subconjuntos difusos en el *sistema relacional* $(X, \tilde{R})$.

### 3.1 Definiciones

Sean $\tilde{A} \in [0,1]^X$, $\tilde{A}^u = \tilde{A} \triangleleft \tilde{R}$ su *mayorante* y sea $h(\tilde{A}^u) = sup\{ \tilde{A}^u(x) / x \in X \} \in [0,1]$ la *altura* de este último. Si existe $\alpha \in [0,1]$ tal que $0 < \alpha \leq h(\tilde{A}^u)$, diremos que $\tilde{A}$ es *$\alpha$-acotado superiormente*. Si $h(\tilde{A}^u) = 0$, diremos que $\tilde{A}$ no es acotado superiormente. En el primer caso, $h(\tilde{A}^u)$ es el *grado de acotación superior* de $\tilde{A}$.

Si $\tilde{A}^l = \tilde{A} \triangleleft \tilde{R}^{op}$ es el *minorante* de $\tilde{A}$ y si existe $\beta \in [0,1]$ tal que $0 < \beta \leq h(\tilde{A}^l)$, diremos que $\tilde{A}$ es *$\beta$-acotado inferiormente*. Si $h(\tilde{A}^l) = 0$, diremos que $\tilde{A}$ no es acotado inferiormente. El valor $h(\tilde{A}^l)$ es el *grado de acotación inferior* de $\tilde{A}$.

Si existe $\gamma \in [0,1]$ tal que $0 < \gamma \leq \min(h(\tilde{A}^u), h(\tilde{A}^l))$, diremos que $\tilde{A}$ es *$\gamma$-acotado*. El elemento $GB(\tilde{A}) = min(h(\tilde{A}^u), h(\tilde{A}^l))$ es el *grado de acotación* de $\tilde{A}$.

Como ejemplo, considerando la implicación y la *t*-norma mencionadas en el apartado *2.1.1.* del ejemplo *2.1*, obtenemos que el subconjunto difuso $\tilde{A} = 0.4/x_1+1/x_2+0.8/x_4$ es 0.6-acotado superior e inferiormente y por tanto es 0.6-acotado.
Por otra parte el nítido $\tilde{1}$ es 0.2-acotado inferiormente, pero no es acotado superiormente y en consecuencia no es acotado.

Si consideramos estos mismos subconjuntos difusos $\tilde{A}$ y $\tilde{1}$ pero con la implicación y la t-norma del apartado *2.1.2.* del ejemplo citado, se obtienen los mismos resultados excepto que $\tilde{A}$ es 0.8-acotado inferiormente.

En [2] se demuestra que los operadores $^u$ y $^l$ son antítonos en $([0,1]^X, \tilde{\subset})$, es decir:

*(vi)* $(\tilde{A} \tilde{\subset} \tilde{B}) \Rightarrow (\tilde{B}^u \tilde{\subset} \tilde{A}^u \ \& \ \tilde{B}^l \tilde{\subset} \tilde{A}^l)$,

Como consecuencia de *(vi)*, se obtiene el resultado:

### 3.2 Proposición

*Si $\tilde{A}$ es un subconjunto difuso $\alpha$-acotado (superiormente, inferiormente), entonces todo subconjunto difuso $\tilde{B}$ tal que $\tilde{B} \tilde{\subset} \tilde{A}$ es también $\alpha$-acotado (superiormente, inferiormente).*

Por otra parte, es inmediato comprobar que si $\tilde{R}$ es la función característica de un orden $\leq_X$ en X y $\tilde{A}$ es nítido, entonces estos conceptos de acotación aquí definidos recuperan los usuales en $(X, \leq_X)$.

Otros conceptos distinguidos en la teoría de conjuntos ordenados son los de *retículo* y *retículo completo*. Proponemos la extensión de este último a estructuras relacionales difusas del modo siguiente:

### 3.3 Definición

Fijadas la implicación "$\rightarrow$", la *t*-norma $\top$ y la relación $\tilde{R}$, consideremos los elementos de la cadena [0,1]:

$$s = inf\{ h( (\top\text{-}SUP)\tilde{A} ) / \tilde{A} \in [0,1]^X \},$$
$$t = inf\{ h( (\top\text{-}INF)\tilde{A} ) / \tilde{A} \in [0,1]^X \},$$

donde $h(\tilde{B})$ es la altura del subconjunto difuso $\tilde{B}$.
Si existe un $\alpha$ tal que $0 < \alpha \leq s$, diremos que $(X, \tilde{R})$ es *$\alpha$-completo superiormente*. Si existe un $\beta$ tal que $0 < \beta \leq t$, diremos que $(X, \tilde{R})$ es *$\beta$-completo inferiormente*. Si existe un $\gamma$ tal que que $0 < \gamma \leq min(s, t)$ diremos que

$(X, \tilde{R})$ es *γ-completo*. Finalmente, llamaremos *grado de completitud* de $(X, \tilde{R})$ al elemento $GC(X, \tilde{R}) \in [0,1]$:

$$GC(X, \tilde{R}) = min(s, t) .$$

En el ejemplo 2.1, tanto en el caso del subapartado *2.1.1* como en el caso del subapartado *2.1.2*, resulta que el sistema relacional $(X, \tilde{R})$ no es γ-completo para ningún γ, ya que en ambos casos $(\top\text{-}SUP)\,\tilde{1} = \tilde{0}$ y por tanto $s = 0$.

En el ejemplo 2.2, consideremos la restricción $S$ de la relación de la tabla 2 al subconjunto $\{x_1,x_2\}$ de X:

Tabla 3

| $S$ | $x_1$ | $x_2$ |
|---|---|---|
| $x_1$ | 1 | 0.8 |
| $x_2$ | 0.2 | 1 |

Para un subconjunto difuso genérico $\tilde{A} = \alpha/x_1+\beta/x_2$ en el sistema relacional $(\{x_1,x_2\},S)$, se prueba que:

$$(\top_1\text{-}SUP)\tilde{A} = max(0.2, min(0.8, 1-\beta))/x_1 + max(0.2, min(0.8, \beta))/x_2 .$$

$$(\top_1\text{-}INF)\tilde{A} = max(0.2, min(0.8, \alpha))/x_1 + max(0.2, min(0.8, 1-\alpha))/x_2 ,$$

de lo que se deduce

$$h((\top_1\text{-}SUP)\tilde{A}) = min(0.8, max(\beta, 1-\beta)) \geq 0.5;$$

$$h((\top_1\text{-}INF)\tilde{A}) = min(0.8, max(\alpha, 1-\alpha)) \geq 0.5.$$

Como para $\tilde{A} = 0.5/x_1+0.5/x_2$ se verifica $h((\top_1\text{-}SUP)\tilde{A}) = h((\top_1\text{-}INF)\tilde{A}) = 0.5$, obtenemos $s = t = 0.5$, y en consecuencia el grado de completitud tiene el valor:

$$GC(\{x_1,x_2\},S) = 0.5 .$$

Es inmediato comprobar que si $\tilde{R}$ es la función característica de un orden $\leq_X$ en X y $\tilde{A}$ es nítido, entonces la definición de completitud dada recupera la usual en conjuntos parcialmente ordenados $(X, \leq_X)$.

# 4 SUBCONJUNTOS DIFUSOS CONSISTENTES Y DIRIGIDOS

En la teoría de conjuntos parcialmente ordenados se contempla un concepto más débil que el de retículo completo: el concepto de *CPO* [1], (acrónimo de *complete partially ordered set*), que está basado en la definición de conjunto dirigido:

Dado $X \neq \emptyset$ con un orden parcial $\leq_X$ en X, se dice que X es *dirigido* si y solo si

$$\forall (x,y) \in X \times X \quad \exists z \in X \text{ tal que } x \leq_X z,\ y \leq_X z.$$

Y $M \neq \emptyset$, subconjunto de un conjunto parcialmente ordenado $(X,\leq_X)$, se dice un *subconjunto dirigido* de X si M, con la restricción del orden $\leq_X$, resulta dirigido.

Se dice que $(X,\leq_X)$ es un *CPO* si tiene elemento mínimo 0 y si para todo subconjunto dirigido $D \subseteq X$ existe el supremo $\vee D \in X$.

Este concepto de conjunto dirigido y su dual, (conjunto ordenado *filtrado*), tienen un papel relevante en muchas de las aplicaciones de la teoría de conjuntos ordenados y en la topología, lo que hace interesante la extensión de estos conceptos a sistemas relacionales difusos $(X, \tilde{R})$.

Sean "$\rightarrow$", $\top$, X, $\tilde{R}$ y $\tilde{A}$ objetos que verifican lo señalado en *(a)*, *(b)*, *(c)* y *(d)* de la introducción. Previo al concepto de dirección difusa, daremos una versión difusa de una definición más general, la de consistencia [1].

### 4.1 Definiciones

Si $sop\,\tilde{B} = \{x \in X / \tilde{B}(x) > 0\}$ representa el soporte de $\tilde{B}$ y si $\alpha \neq 0$ es un elemento de [0,1], diremos que $\tilde{A}$ es α-*consistente* en el sistema relacional $(X, \tilde{R})$ si:

$$h(\tilde{B}^u) \geq \alpha \quad \forall \tilde{B} \tilde{\subset} \tilde{A} \text{ tal que } sop\,\tilde{B} \text{ es finito.}$$

Llamaremos *grado de consistencia* de $\tilde{A}$ en $(X, \tilde{R})$:

$$gr_{con}(\tilde{A}) = sup\{\alpha / \tilde{A} \text{ es } \alpha\text{-consistente}\} \in [0,1].$$

Si $\alpha \neq 0$, diremos que $\tilde{A}$ es α-*dirigido* en el sistema relacional $(X, \tilde{R})$ si y solo si:

$$h(\tilde{A} \intercal \tilde{B}^u) \geq \alpha \quad \forall \tilde{B} \tilde{\subset} \tilde{A} \text{ tal que } sop\,\tilde{B} \text{ es finito.}$$

Y llamaremos *grado de dirección* de $\tilde{A}$ en $(X, \tilde{R})$:

$$gr_{dir}(\tilde{A}) = sup\{\alpha / \tilde{A} \text{ es } \alpha\text{-dirigido}\} \in [0,1].$$

Evidentemente, si $\tilde{A}$ es α-dirigido entonces $sop\tilde{A} \neq \emptyset$. Además, todo α-dirigido es α-consistente y por tanto:

$$gr_{dir}(\tilde{A}) \leq gr_{con}(\tilde{A}) \quad \forall \tilde{A}.$$

Se dice que $\tilde{A}$ es α-*filtrado* en el sistema relacional $(X, \tilde{R})$ si es α-dirigido en $(X, \tilde{R}^{op})$, es decir:

$h(\tilde{A} \intercal \tilde{B}^l) \geq \alpha \quad \forall \tilde{B} \tilde{\subset} \tilde{A}$ tal que $sop\,\tilde{B}$ es finito. Y el *grado de filtración* de $\tilde{A}$ en $(X, \tilde{R})$ viene dado por

$$gr_{fil}(\tilde{A}) = sup\{\alpha / \tilde{A} \text{ es } \alpha\text{-filtrado}\} \in [0,1].$$

En [2] se demuestra que para todo punto difuso $x_\alpha$ se verifica $(x_\alpha)^u(x) = (x_\alpha)^l(x) = 1$, luego el punto difuso $x_\alpha$ es 1-consistente, α-dirigido y α-filtrado.

Demostraremos que para todo $\tilde{A}$ con soporte finito, los grados de consistencia, dirección y filtración se expresan en función de los operadores $^u$, *MAX* y *MIN*.

**4.2 Proposición**
*Si $\tilde{A}$ tiene soporte finito, entonces:*

$$gr_{con}(\tilde{A}) = h(\tilde{A}^u),$$
$$gr_{dir}(\tilde{A}) = h((\mathsf{T}\text{-}MAX)\tilde{A}\ ),$$
$$gr_{fil}(\tilde{A}) = h((\mathsf{T}\text{-}MIN)\tilde{A}\ ).$$

Demostración.
Si $\tilde{A}$ tiene soporte finito y es $\alpha$-consistente entonces $h(\tilde{A}^u) \geq \alpha$, que demuestra que $h(\tilde{A}^u)$ es un mayorante del conjunto $\{\alpha/\ \tilde{A}$ es $\alpha$-consistente$\}$. Por otra parte, para todo $\tilde{B}$ tal que $\tilde{B} \tilde{\subset} \tilde{A}$ se verifica $\tilde{A}^u \tilde{\subset} \tilde{B}^u$ y en consecuencia $h(\tilde{A}^u) \leq h(\tilde{B}^u)$, que prueba que $\tilde{A}$ es $h(\tilde{A}^u)$-consistente, luego $h(\tilde{A}^u) \in \{\alpha/\ \tilde{A}$ es $\alpha$-consistente$\}$ y por lo tanto

$$h(\tilde{A}^u) = sup\{\alpha/\ \tilde{A} \text{ es } \alpha\text{-consistente}\} = gr_{con}(\tilde{A}).$$

Supongamos que $\tilde{A}$ tiene soporte finito y es $\alpha$-dirigido. Se verifica en este caso que $h((\mathsf{T}\text{-}MAX)\tilde{A}\ ) = h(\tilde{A}\ \mathsf{T}\ \tilde{A}^u) \geq \alpha$, luego $h(\tilde{A}\ \mathsf{T}\ \tilde{A}^u)$ es un mayorante del subconjunto $\{\alpha/\ \tilde{A}$ es $\alpha$-dirigido$\}$. Como $\forall\, \tilde{B}$ tal que $\tilde{B} \tilde{\subset} \tilde{A}$ se verifica $\tilde{A}\ \mathsf{T}\ \tilde{A}^u \tilde{\subset} \tilde{A}\ \mathsf{T}\ \tilde{B}^u$ y en consecuencia $h(\tilde{A}\ \mathsf{T}\ \tilde{A}^u) \leq h(\tilde{A}\ \mathsf{T}\ \tilde{B}^u)$, resulta que $\tilde{A}$ es $h(\tilde{A}\ \mathsf{T}\ \tilde{A}^u)$-dirigido. Es decir, $h(\tilde{A}\ \mathsf{T}\ \tilde{A}^u) = h((\mathsf{T}\text{-}MAX)\tilde{A}\ )$ es el máximo del subconjunto $\{\alpha/\ \tilde{A}$ es $\alpha$-dirigido$\}$, luego concluimos finalmente la igualdad

$$gr_{dir}(\tilde{A}) = h((\mathsf{T}\text{-}MAX)\tilde{A}\ ).$$

La demostración de la igualdad $gr_{fil}(\tilde{A}) = h((\mathsf{T}\text{-}MIN)\tilde{A}\ )$ es análoga a la del párrafo anterior. ■

**4.3 Ejemplos.**
En el apartado 2.1.1, para el subconjunto difuso $\tilde{A} = 0.4/x_1 + 1/x_2 + 0.8/x_4$ se verifica que $gr_{con}(\tilde{A}) = 0.6$, $gr_{dir}(\tilde{A}) = 0.2$ y $gr_{fil}(\tilde{A}) = 0.4$, luego es como máximo, 0.6-consistente, 0.2-dirigido y 0.4-filtrado en $(X, \tilde{R})$.

El nítido $\tilde{1}$ es tal que $gr_{con}(\tilde{1}) = gr_{dir}(\tilde{1}) = 0$ y $gr_{fil}(\tilde{1}) = 0.2$, luego no es consistente ni dirigido y es, como máximo, 0.2-filtrado.

En el ejemplo 2.1.2, ( implicación de Lukasiewicz y la t-norma $\alpha \mathsf{T}_2 \beta = \max(0,(\alpha+\beta-1))$, obtenemos los mismos resultados del párrafo anterior para el nítido $\tilde{1}$. En cuanto al subconjunto difuso $\tilde{A}$, el único valor que cambia es $gr_{fil}(\tilde{A})$, que toma ahora el valor 0.2, y por lo tanto $\tilde{A}$ es como máximo 0.2-filtrado.

# 5 CONCLUSIÓN

Mediante las anteriores nociones, se hace posible la extensión de los conceptos relativos a acotación, completitud y dirección usuales, de modo que puedan aplicarse a sistemas relacionales difusos cualesquiera.

En futuros trabajos se estudiará más detalladamente la incidencia de las *t*-normas e implicaciones que intervienen en las definiciones, así como la aplicación de estos desarrollos en el análisis de las preferencias difusas y la decisión.

**Agradecimientos**
Los autores agradecen las observaciones e indicaciones aportadas por el anónimo revisor de este trabajo.

**Referencias**

[1] B.A. Davey and H.A. Priestley (1990) *Introduction to lattices and order*. Cambridge University Press.
[2] R. Fuentes-González (1997). Cotas y extremos asociados a implicaciones difusas y a t-normas. En *CIMAF'97*. (24-28 de marzo). La Habana. Cuba.
[3] L.J. Kohout y W. Bandler (1992). Use of fuzzy relations in knowledge representation, adquisition and processing. En *Fuzzy logic for the management of uncertainty*. John Wiley & Sons. 415-435.
[4] B. Schweizer y A. Sklar (1983). *Probabilistic Metric Spaces*. Elsevier North-Holland.
[5] E. Trillas, C. Alsina y J.M. Terricabras (1995). *Introducción a la lógica borrosa*. Ariel, Barcelona.
[6] E. Trillas y L. Valverde (1984). An inquiry into indistinguishability operators. En *Aspects of Vagueness*. H. Scala et al. eds. Reidel 231-256.
[7] E. Trillas y L.Valverde (1985). On implications and indistinguishability in the setting of fuzzy logic. En *Management Decision Support Systems Using Fuzzy Sets and Possibility Theory*. J. Kaprzyk, R. Yager eds. Verlag TÜV Rheinland, Köln 198-212.
[8] T. Whalen y B. Schott (1992). Presumption, prejudice and regularity in fuzzy material implication. En *Fuzzy logic for the management of uncertainty*. John Wiley & Sons, Inc. 265-280.
[9] L.A. Zadeh (1971) Similarity Relations and Fuzzy Orderings. *Inform. Sci.* vol. **3** . 177-200.

# DIVERGENCIAS ENTRE CONJUNTOS Y ENTRE PARTICIONES DIFUSAS

**Susana Montes**
Dept. Estadística
Universidad de Oviedo
e-mail: smr@pinon.ccu.uniovi.es

**Pedro Gil**
Dept. Estadística
Universidad de Oviedo
e-mail: pedro@pinon.ccu.uniovi.es

## Resumen

El propósito de este trabajo es cuantificar el grado de diferencia entre dos subconjuntos difusos dentro de un referencial finito, a través de unas medidas, a las que se llamará divergencias. Una vez hecho ésto, se extendió la idea de borrosidad de un conjunto a la de borrosidad de toda una partición, dando lugar por tanto, a la definición de divergencia entre particiones.

**Palabras Clave:** Divergencias, medidas de borrosidad, $\varepsilon$-particiones y $\varepsilon$-relaciones.

## 1 MEDIDAS DE DIVERGENCIA

Como ya se ha comentado en el resumen anterior, en este trabajo se ha intentado cuantificar el grado de diferencia entre dos subconjuntos difusos. Es evidente que dos conjuntos serán más parecidos, cuanto más cercanas estén sus funciones de pertenencia en cualquier punto del referencial, y por tanto más parecidas serán sus medidas de borrosidad con la borrosidad del conjunto intermedio entre ellos dos, que se definiría como sigue:

**Definición 1.1** *Dados dos subconjuntos difusos $\widetilde{A}$ y $\widetilde{B}$ dentro de un referencial finito $X$, se define el subconjunto difuso intermedio, que se denotará por $\frac{(\widetilde{A}+\widetilde{B})}{2}$, al conjunto con función de pertenencia el valor medio de las funciones de pertenencia de ambos conjuntos en cada punto, es decir:*

$$\frac{(\widetilde{A}+\widetilde{B})}{2}(x) = \frac{\widetilde{A}(x)+\widetilde{B}(x)}{2} \quad \forall x \in X \qquad (1)$$

Puesto que la medida de la divergencia entre dos conjuntos va a estar basada en la borrosidad de esos conjuntos, se va en primer lugar a recordar la definición de este concepto, que ya ha sido estudiado por numerosos autores [4], [7], [8], [9], [15], [16], ...

**Definición 1.2** *[3] Una medida de borrosidad es una función real $f$ definida sobre $\widetilde{P}(X)$, donde $\widetilde{P}(X) = \{$ subconjuntos difusos de $X\}$, tal que:*
*Axioma 1: $f(\widetilde{A}) = 0 \Longleftrightarrow \widetilde{A}$ es un conjunto nítido, es decir, si no hay borrosidad.*
*Axioma 2: Si $\widetilde{A}, \widetilde{B} \in \widetilde{P}(X)$ y $\widetilde{B}$ es "más difuso" que $\widetilde{A} \Longrightarrow f(\widetilde{A}) \leq f(\widetilde{B})$.*
*Axioma 3: $f(\widetilde{A})$ alcanza el valor máximo $\Longleftrightarrow \widetilde{A}$ es "difuso maximal".*

Es evidente que dependiendo de lo que se esté considerando que es el grado de borrosidad, los términos "más difuso" y "difuso maximal" tendrán significado propio y único dentro de cada concepción particular.

También puede exigírsele y puede considerarse natural un cuarto axioma que asegure que $f(\widetilde{A}) = f(\widetilde{A}^c), \forall \widetilde{A} \subset X$, y con esta suposición vamos a trabajar el resto de las secciones.

Con todo esto, se define la divergencia entre dos conjuntos como sigue:

**Definición 1.3** *Dados dos subconjuntos difusos $\widetilde{A}$ y $\widetilde{B}$ dentro de un referencial finito $X$, se define el grado de divergencia de $\widetilde{A}$ respecto a $\widetilde{B}$ y se denota por $R(\widetilde{A}, \widetilde{B})$, a la función de $\widetilde{P}(X) \times \widetilde{P}(X)$ en $\mathbb{R}$ definida por:*

$$R(\widetilde{A}, \widetilde{B}) = f(\frac{(\widetilde{A}+\widetilde{B})}{2}) - \frac{f(\widetilde{A})+f(\widetilde{B})}{2} \qquad (2)$$

*donde $f$ es una medida cualquiera de borrosidad según la definición 1.1.*

Evidentemente, según la definición que se esté considerando para la medida de borrosidad, dentro de las

numerosas que han sido dadas, se estará obteniendo una clase diferente de medida de divergencia. De hecho, cuando se utilicen las medidas de borrosidad obtenidas a través de medidas de incertidumbre, que proporciona el siguiente teorema, podrá relacionarse el concepto de divergencia con él de la incertidumbre borrosa que existe sobre los puntos del referencial que verdaderamente pertenecen al conjunto.

**Teorema 1.4** *[2] Sea $X = \{x_1, \ldots, x_n\}$ un referencial finito y $\{H_1, \ldots, H_n\}$, una familia de medidas de incertidumbre que verifican el Principio de Transferencia. Si se consideran las funciones $F_i : [0,1] \longrightarrow R^+$ tales que $F_i(p) = H_i(p, 1-p)$ $\forall i = 1, \ldots, n.$, entonces la aplicación $d : \widetilde{P}(X) \longrightarrow \mathbb{R}^+$ definida por $d(A) = h[\sum_{i=1}^{n} F_i(A(x_i))]$, donde $h : \mathbb{R}^+ \longrightarrow \mathbb{R}^+$ es una función no-decreciente tal que $h(x) = 0$ sii $x = 0$, es una medida de borrosidad.*

En base a este teorema, a partir de cualquier medida de incertidumbre se obtiene una medida de borrosidad, y por tanto una medida de divergencia, quedaría definida como sigue,

$$R(\widetilde{A}, \widetilde{B}) = \frac{h\left[\sum_{x\in X} H\left(\left(\frac{\widetilde{A}+\widetilde{B}}{2}\right)(x), \left(\frac{\widetilde{A}+\widetilde{B}}{2}\right)^c(x)\right)\right] - h\left[\sum_{x\in X} H\left(\widetilde{A}(x), \widetilde{A}^c(x)\right)\right] + h\left[\sum_{x\in X} H\left(\widetilde{B}(x), \widetilde{B}^c(x)\right)\right]}{2} \quad (3)$$

expresión, que en el caso en el que h fuese la identidad, tomaría la forma,

$$R(\widetilde{A}, \widetilde{B}) = \sum_{x\in X}\left[H\left(\left(\frac{\widetilde{A}+\widetilde{B}}{2}\right)(x), \left(\frac{\widetilde{A}+\widetilde{B}}{2}\right)^c(x)\right) - \frac{H(\widetilde{A}(x), \widetilde{A}^c(x)) + H(\widetilde{B}(x), \widetilde{B}^c(x))}{2}\right] \quad (4)$$

es decir, la divergencia estaría puesta como suma en todos los puntos del referencial, de la divergencia que habría entre los sistemas $\{\widetilde{A}(x), \widetilde{A}^c(x)\}$ y $\{\widetilde{B}(x), \widetilde{B}^c(x)\}$, concepto que ya ha sido ampliamente estudiado en anteriores investigaciones sobre teoría de la información [1], [12], [14], .... Por tanto, lo que la divergencia hace es medir, para todos los elementos del referencial, la incertidembre que existe sobre si ese elemento concreto está en $\widetilde{A}$ o está en $\widetilde{A}^c$ y la compara con la misma incertidumbre para $\widetilde{B}$, dando como resultados valores pequeños si la incertidumbre para $\widetilde{A}$ es muy cercana a la incertidumbre para $\widetilde{B}$, o valores grandes cuando sean muy diferentes.

Otra clase importante puede ser la formada por aquellas medidas de borrosidad obtenidas a partir de la expresión ([6])

$$f(\widetilde{A}) = h\left(\sum_{x\in X} g_x(\widetilde{A}(x))\right) \quad (5)$$

donde $g_x(x \in X)$ son funciones $g_x : [0,1] \longrightarrow \mathbb{R}^+$ monótonas crecientes en $[0, 1/2]$, concava hacia abajo, simétrica respecto al $1/2$ y satisfaciendo que $g_x(0) = g_x(1) = 0$; $h$ es una función estrictamente creciente de $\mathbb{R}^+$ en $\mathbb{R}^+$.

De esta forma se generan muchas de las principales medidas, como por ejemplo, la medida de borrosidad dada por De Luca y Termini, de gran importancia dentro de la teoría de los conjuntos difusos. Puede ser interesante estudiar este ejemplo con más detenimiento.

**Ejemplo 1.5** *La medida de divergencia que se obtiene, al considerar la medida de borrosidad de De Luca y Termini vendría dada por:*

$$f(\widetilde{A}) = \sum_{x\in X} g(\widetilde{A}(x)) \quad \forall \widetilde{A} \subset X \quad (6)$$

*siendo $g$ la función real definida por la siguiente expresión $g(x) = -(x\log(x) + (1-x)\log(1-x))$, puesto que en todo este trabajo se está considerando como definición del complementario la inicial propuesta por Zadeh: $\widetilde{A}^c(x) = 1 - \widetilde{A}(x)$.*

*Así, en este caso, la divergencia de $\widetilde{A}$ respecto a $\widetilde{B}$, tomaría la forma:*

$$R(\widetilde{A}, \widetilde{B}) = \sum_{x\in X} h(\widetilde{A}(x), \widetilde{B}(x)) \quad (7)$$

*siendo $h$ una función en dos variables definida por*

$$h(x,y) = g\left(\frac{x+y}{2}\right) - \frac{g(x)+g(y)}{2} \quad (8)$$

*cuya representación gráfica es*

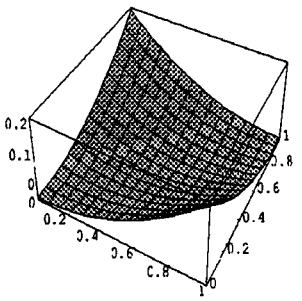



Figura 1: Representación gráfica de la función h(x,y).

*Al observar la figura anterior, es evidente como realmente $R$ toma valores grandes para puntos donde las funciones de pertenencia de $\widetilde{A}$ y $\widetilde{B}$ son muy distintas, tomando el valor máximo cuando una es cero y*

*la otra uno, y toma valores muy pequeños cuando las funciones de pertenencia son muy parecidas, siendo el mínimo cuando coincidan, es decir, cuando en ese punto no haya nada de divergencia.*

Una vez establecido y explicado el significado del concepto de divergencia, se van a ver algunas propiedades interesantes de $R$:

**Propiedades 1.6** *Supuesto que $R$ es la aplicación de $\widetilde{P}(X) \times \widetilde{P}(X)$ en $\mathbb{R}$ definida en 1.3, se verifican las siguientes propiedades:*

*(1) La divergencia de un subconjunto difuso cualquiera del referencial respecto a si mismo es siempre cero.*

$$R(\widetilde{A}, \widetilde{A}) = 0 \quad \forall \widetilde{A} \subset X \tag{9}$$

*(2) La divergencia de un conjunto $\widetilde{A}$ respecto a otro conjunto $\widetilde{B}$ coincide con la divergencia de $\widetilde{B}$ respecto a $\widetilde{A}$.*

$$R(\widetilde{A}, \widetilde{B}) = R(\widetilde{B}, \widetilde{A}) \quad \forall \widetilde{A}, \widetilde{B} \subset X \tag{10}$$

*por lo tanto se hablará en general de divergencia entre los conjuntos, sin ser necesaria la especificación del orden.*

*(3) La divergencia entre un conjunto y su complementario es una función decreciente de la borrosidad del conjunto.*

$$R(\widetilde{A}, \widetilde{A}^c) = f(\widetilde{E}) - f(\widetilde{A}) \quad \forall \widetilde{A} \subset X \tag{11}$$

*donde $\widetilde{E}$ es el conjunto difuso con función de pertenencia $\widetilde{E}(x) = 1/2$ en cualquier punto $x$ del referencial $X$, por lo tanto, cuanto más difuso es $\widetilde{A}$, mayor valor toma $f(\widetilde{A})$ (por el axioma 2 de la definición 1.2), y en consecuencia, puesto que $f(\widetilde{E})$ es constante, menos divergencia existe entre $\widetilde{A}$ y $\widetilde{A}^c$, como era de esperar.*

*(4)Dados dos subconjuntos difusos cualesquiera, la divergencia del menos difuso respecto a su complementario es mayor que la divergencia del otro respecto a su respectivo complementario,*

$$\begin{array}{c} \text{Si } \widetilde{A} \text{ es menos difuso que } \widetilde{B} \Longrightarrow \\ R(\widetilde{A}, \widetilde{A}^c) > R(\widetilde{B}, \widetilde{B}^c) \quad \forall \widetilde{A}, \widetilde{B} \subset X \end{array} \tag{12}$$

*puesto que lógicamente, cuando más difuso es un conjunto, menos divergencia debería existir entre él y su complementario.*

*(5) La divergencia entre dos conjuntos difusos siempre es menor o igual que la divergencia entre un subconjunto cualquiera del referencial y su complementario.*

$$R(\widetilde{A}, \widetilde{B}) \leq R(Z, Z^c) \quad \forall \widetilde{A}, \widetilde{B} \subset X \tag{13}$$

*donde $Z$ es un subconjunto nítido cualquiera de $X$. Es evidente que la divergencia máxima se conseguirá cuando se trabaja con un conjunto nítido y su complementario, puesto que se tendrán los dos conjuntos más diferentes que hay posibilidad de encontrar en $X$, además esta divergencia es la misma para cualquier nítido y coincide con la borrosidad del conjunto $\widetilde{E}$.*

*Como consecuencia de esta propiedad y de la (3), se puede escribir la borrosidad de un conjunto difuso como la diferencia entre la divergencia de un conjunto nítido cualquier y su complementario y la divergencia del difuso y su complementario,*

$$\begin{array}{c} f(\widetilde{A}) = f(\widetilde{E}) - R(\widetilde{A}, \widetilde{A}^c) = R(Z, Z^c) - R(\widetilde{A}, \widetilde{A}^c) \\ \forall \widetilde{A} \subset X \end{array} \tag{14}$$

*puesto que $R(Z, Z^c)$ es una cantidad fija, y así cuanto menos diverja de su complementario, más difuso es, de forma análoga a un criterio ya establecido para medir la borrosidad comparando la distancia de los elementos del conjunto nítido y del difuso con sus respectivos complementarios [5].*

*(6) La divergencia de un conjunto respecto al conjunto $\widetilde{E}$ (algunas veces llamado equilibrio ó difuso maximal), es la misma que la de su complementario.*

$$R(\widetilde{A}, \widetilde{E}) = R(\widetilde{A}^c, \widetilde{E}) \quad \forall \widetilde{A} \subset X \tag{15}$$

*Además de las propiedades anteriores, si $R$ es una medida de divergencia obtenida a partir de una de las medidas de borrosidad dadas mediante la ecuación 5 considerando que $h$ no tiene concavidad hacia arriba, se tendrá también que*

*(7) La divergencia entre dos conjuntos es siempre positiva.*

$$R(\widetilde{A}, \widetilde{B}) \geq 0 \quad \forall \widetilde{A}, \widetilde{B} \subset X \tag{16}$$

*dándose la igualdad si y sólo si $\widetilde{A} = \widetilde{B}$.*

*(8) Dados $\widetilde{A}$, $\widetilde{B}$ y $\widetilde{C}$ con funciones de pertenencia verificando que*

$$\begin{array}{l} \text{o bien, } \widetilde{A}(x) \geq \widetilde{B}(x) \geq \widetilde{C}(x), \\ \text{o bien, } \widetilde{A}(x) \leq \widetilde{B}(x) \leq \widetilde{C}(x), \end{array}$$

*en todo elemento $x$ del referencial, entonces $R(\widetilde{A}, \widetilde{B}) \leq R(\widetilde{A}, \widetilde{C})$, puesto que es evidente por las condiciones que verifican las funciones de pertenencia que $\widetilde{A}$ diverge menos de $\widetilde{B}$ que de $\widetilde{C}$.*

*En particular si se considera el criterio de "ser menos difuso que" siguiente: $\widetilde{A}$ es menos difuso que $\widetilde{B} \Longleftrightarrow$*

$$\begin{array}{l} \widetilde{A}(x) \leq \widetilde{B}(x) \quad \forall x \in X / \widetilde{B}(x) < 1/2 \\ \widetilde{A}(x) \geq \widetilde{B}(x) \quad \forall x \in X / \widetilde{B}(x) > 1/2 \end{array} \tag{17}$$

*entonces si $\widetilde{A}$ es menos difuso que $\widetilde{B}$, se tiene que $R(\widetilde{A},\widetilde{E}) \geq R(\widetilde{B},\widetilde{E})$.*

*Como última propiedad en esta sección, se va a proporcionar una cota para la divergencia de dos conjuntos $\varepsilon$-iguales, antes de lo cual se va a recordar dicha definición:*

**Definición 1.7** *[10] Sea $X$ el referencial y sean $\widetilde{A}$ y $\widetilde{B}$ subconjuntos difusos de él, se dice que $\widetilde{A}$ es $\varepsilon$-igual al subconjunto difuso $\widetilde{B}$, y se denotará por $\widetilde{A} \equiv_\varepsilon \widetilde{B}$, si y sólo si*

$$(\widetilde{A})_\delta = (\widetilde{B})_\delta \quad \forall \delta \in (\varepsilon, 1-\varepsilon) \tag{18}$$

*es decir, si son "casi" iguales, salvo en sus valores extremos.*

*Así, para las divergencia obtenidas a partir de medidas de borrosidad como la del ejemplo 1.5, es decir, en las que la función $h$ es la identidad, se puede asegurar que*

$$Si\ \widetilde{A} \equiv_\varepsilon \widetilde{B} \Longrightarrow R(\widetilde{A},\widetilde{B}) \leq f(\xi) \tag{19}$$

*donde $\xi$ es el conjunto difuso con función de pertenencia $\xi(x) = \varepsilon, \forall x \in X$, es decir, si $\varepsilon$ es pequeño, como suele ser lo habitual, la divergencia es prácticamente nula.*

## 2 DIVERGENCIA ENTRE PARTICIONES

En esta sección se va a cuantificar la divergencia entre dos particiones. Era lógico pensar que para establecer una definición análoga a la dada en la definición 1.3 para conjuntos, sería ahora necesario definir el concepto de la borrosidad que existe en una partición. Esta definición es dada de forma totalmente análoga a la dada en 1.2.

**Definición 2.1** *Dado un referencial $X$, se dice que una función real $b$ definida sobre $\widetilde{Par}(X)$ (particiones difusas en $X$) es una medida de borrosidad de una partición, si verifica los siguientes axiomas:*
*Axioma 1: $b(\Pi) = 0 \Longleftrightarrow \widetilde{A}_i$ es un conjunto nítido, $\forall \widetilde{A}_i \in \Pi$.*
*Axioma 2: Si $\Pi, \Pi' \in \widetilde{P}(X)$ con $\Pi << \Pi' \Longrightarrow b(\Pi) \leq b(\Pi')$.*
*Axioma 3: $b(\Pi)$ alcanza su valor máximo $\Longleftrightarrow \widetilde{A}_i$ es el difuso maximal $\forall \widetilde{A}_i \in \Pi$.*

Al igual que ocurría en la definición 1.3, los términos partición "menos difusa" y "difuso maximal" tomarán una u otra forma dependiendo del investidagor. Una posibilidad para la definición de partición menos difusa (el concepto de conjunto menos difuso y difuso maximal, ya han sido ampliamente estudiados), podría ser por ejemplo la dada en la siguiente definición, aunque ésta es simplemente una idea que ya se ha manejado, pero no es evidentemente la única posibilidad.

**Definición 2.2** *Sean $\Pi$ y $\Pi'$ dos particiones dentro del referencial $X$, se dice que la partición $\Pi$ es menos difusa que $\Pi'$ ($\Pi << \Pi'$) si y sólo si*

$$\forall \widetilde{A}_i \in \Pi, \exists \widetilde{B}_{j_i} \in \Pi' / \widetilde{A}_i \text{ es menos difuso que } \widetilde{B}_{j_i} \text{ y además } \widetilde{B}_{j_i} \neq \widetilde{B}_{j_k} \quad \forall i \neq k \tag{20}$$

Una vez vistos los axiomas que llevan a la definición de las medidas de borrosidad de una partición, la siguiente proposición establece como se pueden generar dichas medidas a partir de las bien estudiadas medidas de borrosidad para conjuntos.

**Proposición 2.3** *Sea $f : \widetilde{P}(X) \longrightarrow \mathbb{R}^+$ una medida de borrosidad dada según la definición 1.2. Se tiene entonces que la aplicación $b : \widetilde{Par}(X) \longrightarrow \mathbb{R}$ que a cada elemento de $\widetilde{Par}(X)$ le asigna el valor*

$$b(\Pi) = \sum_{\widetilde{A}_i \in \Pi} f(\widetilde{A}_i) \quad \forall \Pi \in \widetilde{Par}(X) \tag{21}$$

*es una medida de borrosidad de una partición.*

Una vez sentadas las bases, se puede pasar, de forma totalmente análoga a la establecida en la definición 1.3, a definir la divergencia entre particiones, con una única salvedad, el considerar el hecho de que no debe influir el orden en el que coloquemos los elementos de la partición, es decir, que la divergencia sea la misma a pesar de que haya sido hecha cualquier reordenación.

**Definición 2.4** *Sean $\Pi$ y $\Pi'$ dos particiones dentro del referencial $X$, se define una medida de divergencia entre ellas, y se denota por $R(\Pi,\Pi')$, a la aplicación de $\widetilde{Par}(X) \times \widetilde{Par}(X)$ en $\mathbb{R}$, definida por*

$$R(\Pi,\Pi') = b\left(\frac{\Pi+\Pi'}{2}\right) - \frac{mb(\Pi)+nb(\Pi')}{2} \tag{22}$$

*donde $\dfrac{\Pi+\Pi'}{2}$ representa la familia de conjuntos difusos formados por los conjuntos $\frac{\widetilde{A}_i+\widetilde{B}_j}{2}$ definidos en 1.1, y $n$ y $m$ son el número de elementos en la partición $\Pi$ y $\Pi'$ respectivamente.*

De hecho, si $b$ es una medida de borrosidad generada según la proposición 2.3, la divergencia entre dos particiones, se puede relacionar con la divergencia entre sus elementos, como muestra la siguiente proposición.

**Proposición 2.5** *Si $R$ es una medida de divergencia entre particiones definida en 2.4, donde $b$ es generada*

*según la proposición 2.3, entonces la divergencia entre dos particiones se puede calcular mediante la siguiente expresión equivalente:*

$$R(\Pi,\Pi') = \sum_{\widetilde{A}_i\in\Pi}\sum_{\widetilde{B}_j\in\Pi'} R(\widetilde{A}_i,\widetilde{B}_j). \tag{23}$$

Con esta definición es fácil probar la siguiente propiedad, consecuencia directa de la propiedad (4) de la sección anterior.

**Proposición 2.6** *Dadas dos particiones $\Pi$ y $\Pi'$ con el mismo número de elementos, si $\Pi$ es menos difusa, con el criterio que se esté considerando conveniente en ese momento, que $\Pi'$, entonces la divergencia de la primera a su complementario, es menor o igual que la misma divergencia para la segunda, es decir,*

$$\begin{array}{l} \textit{Si } \Pi \textit{ es menos difusa que } \Pi' \\ \Longrightarrow R(\Pi,\Pi^c) > R(\Pi',\Pi'^c) \end{array} \tag{24}$$

*donde el complementario de una partición se define como*

$$\Pi^c = \{\widetilde{A}_i^c / \widetilde{A}_i \in \Pi\} \tag{25}$$

Lo último que se va a hacer en este trabajo es relacionar el concepto de divergencia con él de distancia entre particiones, que ya ha sido tratado en anteriores estudios [11]. Se va en primer lugar a recordar esta distancia que venía definida por

**Definición 2.7** *Sean $\Pi = \{\widetilde{A}_1,\widetilde{A}_2,\ldots,\widetilde{A}_n\}$ y $\Pi' = \{\widetilde{B}_1,\widetilde{B}_2,\ldots,\widetilde{B}_n\}$, dos particiones dentro de un referencial finito $X$. Se define la distancia entre $\Pi$ y $\Pi'$ como:*

$$\begin{aligned} d(\Pi,\Pi') &= \min_{\sigma\in S_n}\sum_{\widetilde{A}_i\in\Pi} d_H(\widetilde{A}_i,\widetilde{B}_{\sigma(i)}) \\ &= \min_{\sigma\in S_n}\sum_{\widetilde{A}_i\in\Pi}\sum_{x\in X}|\widetilde{A}_i(x)-\widetilde{B}_i(x)| \end{aligned} \tag{26}$$

*puesto que $d_H$ representa la distancia de Hamming entre conjuntos difusos y $S_n$ denota el conjunto de todas las permutaciones de $\{1,2,\ldots,n\}$, y por tanto todas las permutaciones posibles de los elementos de $\Pi'$.*

Pues bien, una vez hecho esto, se van a relacionar ambos conceptos en el caso particular de dos tipos distintos de definiciones de partición, el dado por Ruspini [13] y él de $\varepsilon$-partición [11]. De hecho, en parte, toda esta sección ha sido desarrollada para estudiar la divergencia existente entre una partición según la definición dada por Ruspini

**Definición 2.8** *La familia formada por los subconjuntos $\{\widetilde{A}_1,\widetilde{A}_2,\ldots,\widetilde{A}_n\}$ dentro un referencial discreto $X$ es una partición de un conjunto difuso $\widetilde{A}$ si y sólo si se cumple que:*

$$\sum_{i=1}^{n}\widetilde{A}_i(x) = \widetilde{A}(x) \quad \forall x\in X$$

y su $\varepsilon$-partición más próxima, que es de entre todas las $\varepsilon$-particiones de un conjunto, definidas como

**Definición 2.9** *Dada una familia $\{\widetilde{A}_1,\widetilde{A}_2,\ldots,\widetilde{A}_n\}$ de conjuntos difusos de $X$ se dice que es una $\varepsilon$-partición de un conjunto $\widetilde{A}$ si verifica que:*

$$\begin{array}{c} (\widetilde{A}_i\cap\widetilde{A}_j) \equiv_\varepsilon \emptyset \quad \forall i,j = 1,2,\ldots,n \\ (\bigcup_{i=1}^{n}\widetilde{A}_i) \equiv_\varepsilon \widetilde{A} \end{array} \tag{27}$$

aquella que dista menos, midiendo dicha distancia según la definición 2.7, de la partición de Ruspini dada. La forma de esta $\varepsilon$-partición ya ha sido estudiada en un trabajo anterior [11]. Pues bien, con todo esto, para las divergencias definidas a partir de las familias de medidas de borrosidad dadas en 5 siendo $h$ lineal, se tiene

**Proposición 2.10** *Dada una partición $\Pi$ según la definición de Ruspini 2.8 del referencial, si $\Pi'$ es la $\varepsilon$-partición más próxima a él, entonces se tiene que*

$$R(\Pi,\Pi') \le R(\Pi,\Pi'') \tag{28}$$

*siendo $\Pi''$ cualquier otra $\varepsilon$-partición de $X$.*

Como consecuencia de esta proposición, se puede demostrar de forma análoga, que si consideramos la partición nítida generada a partir de la relación de equivalencia nítida procedente de realizar el $\varepsilon$-corte en la relación de $\varepsilon$-equivalencia que genera una $\varepsilon$-partición (no se va a profundizar en las definiciónes de $\varepsilon$-equivalencia, o de como ésta genera una $\varepsilon$-partición, por considerar que ésto daría lugar a introducir demasiados términos en este trabajo; una explicación detallada de estos resultados relativos a relaciones de $\varepsilon$-equivalencia pueden encontrarse en [10]), entonces la divergencia entre la $\varepsilon$-partición y la partición nítida resultante es menor o igual que la divergencia de la $\varepsilon$-partición respecto a cualquier otra partición nítida, es decir,

**Proposición 2.11** *Si se considera una $\varepsilon$-partición $\Pi$ del referencial $X$, esta llevará asociada una relación de $\varepsilon$-equivalencia $\widetilde{R}$, y el $\varepsilon$-corte de ésta $\widetilde{R}_\varepsilon$, será una relación de equivalencia nítida, que generará por tanto una partición nítida $\Pi_\varepsilon$, pues bien, bajo estas hipótesis,*

$$R(\Pi,\Pi_\varepsilon) \le R(\Pi,\Pi_n) \tag{29}$$

*donde $\Pi_n$ denota cualquier partición nítida de $X$.*

Como último comentario, decir que también se verifica que $\Pi_\varepsilon$ es la partición nítida que menos dista de $\Pi$, lo que lleva de nuevo a pensar que el criterio de la divergencia es menos fuerte que él de distancia.

## 3 CONCLUSIONES Y PUNTOS ABIERTOS

En este trabajo se ha intentado introducir un criterio para medir la diferencia entre dos conjuntos difusos, al que se ha llamado divergencia, extendiéndolo después para medir la divergencia entre dos particiones. La aplicación definida con este propósito, verifica, como se ha visto a lo largo del trabajo, unas muy buenas propiedades que dejan constancia de como realmente esta aplicación está midiendo lo que se desea. En todo el desarrollo se han manejado varios conceptos ya introducidos por los autores en anteriores estudios, en los que se ha intentado no profundizar demasiado, debido a las limitaciones de espacio, pero de los que posiblemente fuese interesante tener un conocimiento más detallado para la mejor compresión de este trabajo.

En cuanto a los puntos abiertos, el más inmediato sería el de buscar otros criterios para medir la divergencia que no pasasen necesariamente por las medidas de borrosidad, además de intentar axiomatizar el concepto de medida de divergencia.

**Agradecimientos**

La investigación de este trabajo está subvencionada en parte a través del Proyecto DGICYT PB94-1328. Queremos agradecer al Ministerio de Educación y Ciencia dicha ayuda económica.

## Referencias

[1] J. Burbea and C.R. Rao (1982). Entropy differential metric, distance and divergence measures in probabilistic spaces: a unified approach. *T. Multiv. Anal.* **12**:575-596.

[2] I. Couso, P. Gil and S. Montes (1996). Measures of fuzziness and Information Theory. *IPMU'96* **1**:501-505.

[3] D. Dubois and H. Prade (1980). Fuzzy Sets and Systems. Theory and Applications. *Academic Press, Inc.*

[4] Emptoz (1981). Nonprobabilistic entropies and indetermination measures in the setting of fuzzy sets theory. *Fuzzy Sets and Systems* **5**:307-317.

[5] G.J. Klir and T.A. Folger (1988). Fuzzy Sets, Uncertainty, and Information. *Prentice-Hall International (UK) Limited*

[6] G.J. Klir and B. Yuan (1988). Fuzzy Sets and Fuzzy Logic. Theory ans applications. *Prentice-Hall International (UK) Limited*

[7] J. Knopfmacher (1975). On measures of fuzziness. *J. Math. Analysis and Applications* **49**:529-534.

[8] S.G. Loo (1977). Measures of fuzziness. *Cybernetica* **20**:201-210.

[9] A. De Luca and S. Termini (1972). A definition of a Nonprobabilistic Entropy in the Setting of Fuzzy Sets Theory. *Inf. and Control* **20**:301-312.

[10] S. Montes, J. Jiménez and P. Gil (1996). Épsilon-Particiones y Alfa-Equivalencias. *ESTYLF'96* **1**:201-205.

[11] S. Montes, M.J. López-Palomo and P. Gil (1995). Las épsilon-Particiones. *ESTYLF'95* **1**:96-99.

[12] L. Pardo and al. (1992). H-fi-divergences.

[13] E.H. Ruspini (1969). A new approach to clustering. *Inf. Control* **15**:22-32.

[14] I.J. Taneja, L. Pardo, D. Morales and M.L. Menéndez (1989). On Generalized Information and Divergence Measures and their Applications: A Brief review. *Qestiió.* **13**:47-75.

[15] E. Trillas and T. Riera (1978). Entropies in Finite Fuzzy Sets. *Inf. Sc.* **15**:159-168.

[16] R.R. Yager (1995). Measures of Entropy and Fuzziness Related to Agregation Operators. *Information and Control* **82**:147-166.

# UN MODELO MULTIOBJETIVO FUZZY PARA EL CONTROL MEDIOAMBIENTAL AGRICOLA

**J. M. Cadenas[a] y J.L. Verdegay[b]**
[a]Departamento de Informatica y Sistemas. Universidad de Murcia. 30071 Espinardo, Murcia
[b]Departamento de Ciencias de la Computacion e Inteligencia Artificial. Universidad de Granada. 18071 Granada
jcadenas@dif.um.es,verdegay@goliat.ugr.es

## Resumen

En este articulo se justifica el uso de modelos de Programacion Lineal Multiobjetivo Fuzzy en el control de la calidad medioambiental, cuando los agentes involucrados en la toma de decisiones plantean objetivos de naturaleza imprecisa. Asi mismo, y como ilustracion del enfoque general propuesto, se formula un modelo multiobjetivo que muestra la adecuacion del enfoque propuesto en el caso de terrenos agricolas.

**Palabras Clave:** Problemas multiobjetivo, medioambiente, Programacion Lineal Fuzzy

## 1 INTRODUCCION

Que duda cabe que los modelos de Programacion Lineal (PL) tienen un rango de aplicaciones tan amplio que, sin temor a equivocarnos, podemos asegurar que estan presentes en cualquier area cientifica. Por otro lado, los modelos que usan los conjuntos y la logica fuzzy, tambien tienen una muy importante presencia en un sin fin de campos. Como consecuencia, la Programacion Lineal Fuzzy (PLF) se ha exhibido como una destacada metodologia, capaz de resolver situaciones inabordables con tecnicas convencionales en una enorme variedad de ambitos.

Uno de los temas en los que las tecnicas de PL, y por supuesto las de PLF, aun no han demostrado todo su potencial, y sin embargo pueden ofrecer todas las ventajas que la consolidada teoria en las que se basan ofrece, es el referente a algunos modelos de gestion y evaluacion del impacto medioambiental. En este ambito, sin embargo, los modelos que suelen aparecer, mas que uni-objetivo, involucran a varios objetivos que, de forma comprensible, representan a los diferentes agentes que pueden estar interesados en la referida gestion o evaluacion.

Con la idea de avanzar en las posibilidades que la PLF ofrece para la resolucion de problemas de esta naturaleza, en este articulo, y con la base propuesta en [2], comenzamos por describir de forma general el marco teorico que sustenta los Problemas de Programacion Multiobjetivo Fuzzy, asi como un enfoque de solucion de los mismos que no restrinja a los decisores implicados a disponer de informacion exclusivamente precisa, sino que les facilite el manejo de datos de naturaleza vaga, y consecuentemente fuzzy, como suelen estar disponibles en la mayoria de las situaciones practicas.

De las diferentes posibilidades que se pueden abordar, y basicamente por razones de espacio, aqui nos centraremos en el caso de objetivos fuzzy, aunque se indican otras direcciones para desarrollos futuros.

Con ese soporte teorico, a continuacion, presentaremos un modelo de control de la calidad medioambiental que, aunque circunscrito al caso agricola, puede servir para atender otras situaciones particulares diferentes.

## 2 PROBLEMAS DE OPTIMIZACION MULTIOBJETIVO FUZZY

En terminos muy generales un problema de Programacion Matematica (PM) se plantea en los siguientes terminos

$$\text{Min } \{f(x)/x \in X\} \quad (1)$$

donde x es un vector de dimension N de variables de decision, f es una funcion real que, usualmente, se llama funcion objetivo, y X es el conjunto de restricciones que han de cumplir las variables,

$$X = \{x/x \in \mathbf{R}^N,\ h_i(x)=0,\ i=1,2,...,p;\ g_j(x) \leq 0,\ j=1,2,...,q\}$$

en el que las $h_i$ y las $g_j$ son funciones reales definidas sobre X.

Para resolver este problema se necesita encontrar un $x^* \in X$ tal que,

$$f(x^*) \leq f(x), \forall x \in X$$

A tal $x^*$ se le llama optimo global de (1).

Sin embargo hay muchas situaciones practicas que involucran mas de un objetivo. Una de las mas importantes clases de problemas de decision multiobjetivo es el bien conocido Problema del Vector Maximo (PVM), tambien llamado Problema de Optimizacion Multiobjetivo.

Algunos trabajos claves en este tema son los de V. Pareto [10], donde se introdujo el concepto de solucion no dominada, el de H.W. Kuhn y A.W. Tucker [8], dando las condiciones necesarias y suficientes para la no dominancia, el de L.A. Zadeh [13], caracterizando una solucion de un PVM como no dominada, y el mas general de V. Chankong e Y.Y. Haimes [4] describiendo un buen numero de tecnicas y metodologias de solucion del PVM.

Abreviadamente un PVM se plantea como,

$$\text{Min } \{f_1(x), f_2(x), ..., f_n(x) / x \in X\}$$

Cuando en particular se considera el caso lineal, es decir, cuando todas las funciones objetivo y las que definen las restricciones son lineales, como sera el caso que tratemos en este articulo, el modelo se refiere como un PVM lineal, que se plantea de forma facilmente comprensible en los siguientes terminos

$$\text{Min } \{c_1x, c_2x, ..., c_nx / Ax \leq b, x \geq 0\} \quad (2)$$

siendo los $c_i$ vectores N-dimensionales de valores reales que definen los costos, A una matriz de coeficientes reales mxN, y b un vector m-dimensional que se conoce con el nombre de demanda (o recursos). En todo lo que sigue nos referiremos a este caso de PVM lineal, simplemente por PVM.

Resolver un PVM supone encontrar su conjunto de soluciones no dominadas, es decir, el conjunto de los $x^*$ tales que no existe ningun otro $x \in X$ tal que

$$f_j(x) \leq f_j(x^*)$$

para todo j = 1,...,n, dandose la desigualdad estricta el al menos un j.

Sin embargo, desde un punto de vista practico, es necesario relacionar este concepto con alguno que resulte operativo, y la forma mas y mejor conocida es la de caracterizar las soluciones no dominadas como soluciones optimales. Entre los muchos enfoques que existen para hacer esto, hay dos que son los mas utilizados: el enfoque ponderativo y el restrictivo.

Por medio del primer enfoque, [4], se define el problema ponderado, para algun vector de pesos, como

$$\text{Min } \{\sum_{j=1,n} \omega_j(c_jx) / Ax \leq b; x \geq 0\} \quad (3)$$

donde $\omega_j \geq 0$ y $\sum_{j=1,n} \omega_j = 1$. En este caso lineal, como se sabe bien, pueden encontrarse todas las soluciones no dominadas resolviendo (3).

El problema auxiliar correspondiente al segundo enfoque suele denominarse Problema $\lambda$-Restringido al k-esimo objetivo, [4], y se plantea del siguiente modo

$$\text{Min } \{c_kx / c_jx \leq \lambda_j, j=1,...,n; j \neq k; Ax \leq b, x \geq 0\} \quad (4)$$

donde $\lambda = (\lambda_1,...,\lambda b\ k\text{-}1,\lambda_{k+1},...,\lambda_n)$. En este caso, resolviendo (4), pueden encontrarse todas las soluciones no dominadas.

Adicionalmente a todo este material teorico, desde el importante articulo de R.E. Bellman y L.A. Zadeh de 1970, [1], y mas adelante por la contribucion de H.J. Zimmermann en 1978, [14], el tema del PVM en ambiente fuzzy (PVMF) ha recibido una considerable atencion por un buen numero de investigadores que, en casi todos los casos y de forma paralela a lo ocurrido con el PVM convencional, han orientado sus esfuerzos a la caracterizacion de soluciones no dominadas, aunque ahora considerando las diferentes versiones del PVMF que pueden tenerse en cuenta, segun que la imprecision aparezca en las resticciones o en los objetivos, [3,5,10,11].

En ese ultimo caso, y cuando de lo que se dispone es de informacion fuzzy sobre los coeficientes que participan en las distintas funciones objetivo, estos coeficientes pueden modelizarse mediante numeros fuzzy, y asi el problema que se podria considerar es el siguiente,

$$\text{Min } \{c_1^fx, c_2^fx, ..., c_n^fx / Ax \leq b, x \geq 0\} \quad (5)$$

donde ahora cada vector $c_j^f$, j = 1,..., n, esta definido por N numeros fuzzy que, por simplicidad, podemos suponer triangulares, aunque numeros de tipo LR mas generales tambien podrian ser asumidos sin añadir complejidad en nuestro caso, y que conforman un PVMF que, como

mostraremos en lo que sigue, tiene varias posibilidades de desarrollo.

En efecto, consideremos el modelo (5). Para resolver este problema, desde un punto de vista teorico, podemos emplear el anterior enfoque restrictivo. Asi el problema paralelo al (4), en una primera aproximacion, quedaria como

$$\text{Min } \{c_1^f x \,/\, c_j^f x \leq \lambda_j,\ j=2,...,n;\ Ax \leq b,\ x \geq 0\} \quad (6)$$

Pero ahora algunas de las restricciones que aparecen aqui, desde un punto de vista estrictamente teorico, no tienen mucho sentido ya que involucran valores fuzzy y reales en restricciones que pretenden compararlos, por lo que resulta mas correcto el siguiente modelo

$$\text{Min } \{c_1^f x \,/\, c_j^f x \leq_f \lambda_j^f,\ j=2,...,n;\ Ax \leq b,\ x \geq 0\} \quad (7)$$

donde el simbolo $\leq_f$ hace referencia a que la forma de comparar ambos miembros en la desigualdad tiene que hacerse mediante una relacion de ordenacion de numeros fuzzy.

Respecto a este problema es necesario hacer tres consideraciones,

a) Como queda claro (7) generaliza de modo trivial al problema (4), ya que cuando los numeros fuzzy que en el aparecen se particularizan a numeros reales, la relacion $\leq$ sub f pasa a ser una relacion convencional.

b) La relacion de comparacion de numeros fuzzy $\leq_f$ puede ser cualquiera de la extensa lista disponible, [12], pero lo que es mucho mas importante: podemos considerar diferentes relaciones. Concretamente, podemos suponer que los n objetivos estan asociados a n distintos agentes. Entonces, como resulta evidente que no todos esos agentes tienen que estar de acuerdo en emplear el mismo criterio de comparacion de numeros fuzzy, el problema (7) nos permite representar estas distintas formas de entender la comparacion, sin mas que introducir las diferentes funciones de ordenacion, respectivamente, en la restriccion correspondiente a cada objetivo que se adecue a esta situacion.

c) Aunque aqui, por comodidad, estamos ciñendonos al caso lineal, no hay ningun inconveniente teorico para extender y definir la generalizacion no lineal de (7), aunque desde un punto de vista practico tal generalizacion pudiera no ser facilmente resoluble, o incluso imposible de resolver con tecnicas precisas.

Por tanto resulta obvio que cuando tenemos un PVMF como (6), el modelo (7) nos facilita un marco teorico para encontrar una solucion (o un conjunto de soluciones) al mismo, pero no un unico modelo auxiliar para su solucion como mostraremos a continuacion.

En efecto, para dar una forma operativa a (7), recordemos brevemente algo sobre los diferentes indices de comparacion entre numeros difusos existentes en la literatura [2, 12]. De entre los diferentes enfoques existentes para comparar estas cantidades, aqui solo consideraremos el que se deriva del empleo de indices de comparacion.

En efecto, si

$$I: \mathbf{F(R)} \rightarrow [0,1]$$

es un indice de comparacion para numeros fuzzy, entonces

$$\forall A^f, B^f \in \mathbf{F(R)},\ A^f \leq_f B^f \Leftrightarrow I(A^f) \leq I(B^f)$$

por lo que, conforme al indice I que se use, se pueden obtener diferentes modelos auxiliares para resolver (7), que es nuestro objetivo teorico.

Asi, en general, el modelo auxiliar que sirve para resolver (7), y consecuentemente, (5), queda planteado como

$$\text{Min } \{I(c_1^f x) \,/\, I(c_j^f x) \leq I(\lambda_j^f),\ j=2,...,n;\ Ax \leq b,\ x \geq 0\} \quad (8)$$

o, mucho mas generalmente, como

$$\text{Min}\{I^1(c_1^f x) \,/\, I^j(c_j^f x) \leq I^j(\lambda_j^f),\ j=2,...,n;\ Ax \leq b,\ x \geq 0\} \quad (9)$$

donde ahora $I^j$, j = 1,...,n, es el indice con el que el j-esimo agente que interviene en el problema quiere efectuar las comparaciones referentes al objetivo que el mismo aporta.

Como ejemplo trivial de lo expuesto, si $A^f, B^f \in \mathbf{F(R)}$ son dos numeros fuzzy, que convencionalmente notamos por $A^f = (A, A_i, A_d)$ y $B^f = (B, B_i, B_d)$, y la forma de comparacion, comun en todos los casos, es la dada por el indice

$$A^f \leq_f B^f \Leftrightarrow A \leq B$$

entonces (9) toma la forma,

$$\text{Min } \{c_1 x \,/\, c_j x \leq \lambda_j,\ j=2,...,n;\ Ax \leq b,\ x \geq 0\}$$

que evidentemente reproduce (4).

## 3 UN MODELO DE CONTROL MEDIO AMBIENTAL EN CONTEXTO AGRICOLA

Afortunadamente cada vez es mayor la cantidad de recursos economicos que se destinan a la gestion y control del medio ambiente. Siendo este un objetivo que se explica en si mismo, adicionalmente esta el de realizar esas inversiones de la manera mas efectiva, en terminos de costo y rendimiento.

Como en otras areas, [6], la PL juega un importante papel en este contexto ya que, dado el caracter global que permiten los planteamientos de sus problemas, y la facilidad que tienen los algoritmos que los resuelven, junto con la garantia de obtencion de soluciones optimales, proporciona una metodologia consistente y suficientemente experimentada.

Aunque en este campo las tecnicas de PL, y mas generalmente las de PM, pueden aplicarse a una gran variedad de situaciones (planificacion, gestion de recursos, evaluacion de impactos, ...), y en diferentes ambitos (aire, agua, tierra), en casi todos los casos, y debido a la naturaleza de los datos disponibles, los modelos a emplear tienen una marcada naturaleza fuzzy que, hasta ahora ha sido muy poco explorada.

En este sentido en lo que sigue proponemos una primera aproximacion al tema, concentrandonos en el terreno agricola, y desde el punto de vista de lo anteriormente desarrollado, es decir, suponiendo la existencia de multiples objetivos, en correspondencia con distintos agentes, cuyos coeficientes estan definidos por numeros fuzzy.

El modelo generico que consideramos se basa en los propuestos por E.O. Heady y G.F. Vocke en su libro de 1992, [8], pero con la novedad de permitir la participacion de distintos agentes que, a su vez, disponen de datos fuzzy, aunque no exclusivamente.

Las caracteristicas del modelo que aqui consideramos son las siguientes:

Suponemos que hay i productores agricolas que pueden vender sus productos en j mercados. Cada agricultor puede emplear k metodos diferentes de produccion, y dispone de un conjunto de s terrenos distintos. Adicionalmente, se supone la existencia de h productos quimicos (pesticidas, fertilizantes, etc.) y que hay una disponibilidad de p cultivos. Sobre estos datos aqui no se hara ninguna hipotesis de vaguedad. Aunque desde otro punto de vista, por ejemplo, podria suponerse cierta imprecision en la diferencias que caracterizan a las superficies cultivables, en el enfoque que estamos planteando no las consideraremos.

Las variables que se consideran son $X_{ipk}$, que dara la superficie de terreno de la region o del productor i, para producir el cultivo p con el metodo k, y $T_{pij}$, para representar la cantidad del producto p que se transportaria desde la region i hasta el mercado j. Como resulta evidente, los valores que se obtengan para estas variables pueden asumir perfectamente una naturaleza fuzzy.

Con estas definiciones, aparecen dos tipos diferentes de coeficientes imprecisos para las funciones objetivo, que aqui tienen esta naturaleza por la dificultad que en el terreno agricola tiene la precision de los mismos, y que se corresponden con dos distintos agentes: Por un lado, los productores en si mismos, y por otro los transportistas de mercancias. Para los primeros, por $(cx^f)_{ipk}$ notaremos el costo total de produccion, que dependera de diferentes factores, entre los que destacan principalmente los climatologicos. Para los segundos, $(ct^f)_{pij}$ representara el beneficio del transporte de las mercancias, que finalmente dependera de las producciones que hubiera en global en la region, de modo que no pueden conocerse con precision en tanto no concluya cada campaña, siendo necesario sin embargo conocer su valor con anticipacion de cara a la optimizacion de rendimientos y beneficios.

Por tanto, los dos objetivos que habremos de tener en cuenta son

$$\text{Min: } \Sigma_{i.p.k} (cx^f)_{ipk} X_{ipk}$$

$$\text{Max: } \Sigma_{p,i,j} (ct^f)_{pij} T_{pij}$$

que por ser lineales podemos sintetizar en

$$\text{Min } \{\Sigma_{i.p.k} (cx^f)_{ipk} X_{ipk}, - \Sigma_{p,i,j} (ct^f)_{pij} T_{pij}\}$$

Por otro lado, es evidente que la superficie total de terrenos que se pueden cultivar estara limitada. Esta restriccion, aunque admitiria otras modalidades, podemos establecerla como

$$L_i = \Sigma_{p.k} X_{ipk}$$

En lo que respecta a equilibrar lo producido con lo que se quiere vender en el mercado, si $R_{ipk}$ nota la tasa producida, por unidad de superficie, del cultivo p en la zona i, usando el metodo k, entonces

$$\Sigma_k R_{ipk} X_{ipk} - \Sigma_j T_{pij} = 0$$

Naturalmente, se supone conocida la demanda del cultivo p en cada mercado j, que notamos $d_{pj}$, y que verificara

$$\Sigma_j T_{pij} \geq d_{pj}$$

Como resulta obvio, por las mismas razones que concurren en los costos y beneficios considerados en este modelo, tanto esta como la restriccion sobre la superficie total de terrenos que se pueden cultivar, son restricciones de evidente naturaleza fuzzy, cuya introduccion y tratamiento como tal no aportaria mayor dificultad al problema que tratamos.

Sin embargo, por razones metodologicas que se explican facilmente por el hecho de no haber contemplado en este articulo el caso de restricciones fuzzy, aqui haremos la hipotesis restrictiva, pero debil, de que esta restriccion es de naturaleza clasica, dejando para un posterior trabajo su consideracion como fuzzy. No obstante es importante incidir en que su tratamiento como restricciones fuzzy no aportaria una mayor complejidad a la resolucion del problema, [5].

Por otro lado, si notamos, $a_{psk}$ a la tasa de riesgo que se asocia al producir el cultivo p en un terreno de tipo s, para cada metodo de produccion, y $\alpha_{is}$ es una variable booleana que toma el valor 1 si el agricultor i tiene terrenos de tipo s (y 0 en caso contrario), tendremos que cada agricultor podra fijar su riesgo como

$$D_i \leq \sum_{p,s,k} a_{psk}\alpha_{is} X_{ipk}$$

Finalmente, notando como $b_{phk}$ la cantidad de producto quimico h que emplea, o produce, el metodo k para producir p (con el acuerdo de que $b_{phk}$ tendra valor negativo si se usa, como es el caso de los pesticidas, y positivo si se obtiene en la produccion, como es el caso del nitrogeno que aporta el estiercol), entonces una restriccion de calidad que puede establecerse para cada agricultor, asociada a cada producto quimico es la siguiente

$$C_{ih} \geq \sum_{p.k} b_{phk} X_{ipk}$$

Notamos que, estos dos ultimos grupos de restricciones plantean las mismas posibilidades que las de demanda anteriores pero que, por las mismas razones de antes, aqui no consideramos su version fuzzy.

Consecuentemente, el modelo que planteamos se formula como sigue,

$$\text{Min } \{\sum_{i.p.k} (cx^f)_{ipk} X_{ipk}, - \sum_{p,i,j} (ct^f)_{pij} T_{pij}\}$$
s.a:
$$L_i = \sum_{p.k} X_{ipk}$$
$$\sum_k R_{ipk} X_{ipk} - \sum_j T_{pij} = 0$$
$$\sum_j T_{pij} \geq d_{pj}$$
$$D_i \leq \sum_{p,s,k} a_{psk}\alpha lt_{is} X_{ipk}$$
$$C_{ih} \geq \sum_{p.k} b_{phk} X_{ipk}$$
$$X_{ipk}, T_{pij} \geq 0, \alpha_{is} = 0,1$$

en el que los coeficientes que intervienen en los objetivos los consideramos definidos por numeros fuzzy.

De acuerdo, con lo expuesto anteriormente, si cada agricultor quisiera emplear un indice de comparacion $I^i$ y cada transportista uno $I^j$, el modelo que de acuerdo con (9) se plantearia para resolver seria,

$$\text{Min : } \sum_{i.p.k} I^i(cx^f)_{pk} X_{ipk}$$
s.a:
$$\sum_{p,i,j} I^j(ct^f)_{pi} T_{pij} \geq \sum_{p,i,j} I^j(\lambda_{pi}{}^f),$$
$$L_i = \sum_{p.k} X_{ipk}$$
$$\sum_k R_{ipk} X_{ipk} - \sum_j T_{pij} = 0$$
$$\sum_j T_{pij} \geq d_{pj}$$
$$D_i \leq \sum_{p,s,k} a_{psk}\alpha_{is} X_{ipk}$$
$$C_{ih} \geq \sum_{p.k} b_{phk} X_{ipk}$$
$$X_{ipk}, T_{pij} \geq 0, \alpha_{is} = 0,1$$

Este modelo, aunque suficientemente general, aun puede extenderse mas, independientemente de la consideracion de restricciones fuzzy ya comentada, ya que cualquiera de las variables podria considerarse acotada, al igual que la superficie de terreno disponible podria expresarse con otro tipo de ecuaciones.

Asi mismo, en lo que concierne a las restricciones de riesgo, tambien podrian imponerse condiciones referentes a los niveles de contaminacion que pudieran producir los diferentes agentes quimicos, y las funciones objetivo, y por tanto los decisores implicados, podrian ser mas numerosos. En cualquier caso, estas adiciones no supondrian mayor complejidad al modelo, sino mas bien una mayor capacidad de informacion. Basicamente esto se debe al hecho de que empleando como metodo de solucion de este tipo de problemas, el conocido por Enfoque Parametrico, el modelo auxiliar que se utiliza para obtener una solucion fuzzy en los valores de las variables, no aumenta la dimension del problema original, si bien hay que resolver un problema parametrico. Dada la naturaleza lineal de los modelos aqui considerados, la resolucion de un problema parametrico, incluso aunque las funciones de pertenencia que se considerarn no fueran lineales, no implica complejidad.

## CONCLUSION

A partir de un modelo multiobjetivo, en el que los decisores participantes pueden expresar sus diferentes formas de entender las ordenaciones, de modos distintos, y suponiendo que los coeficientes que definen las respectivas funciones objetivo estan dados por numeros fuzzy, se ha presentado un modelo para el control de la calidad medioambiental de los terrenos agricolas que, ademas, incorpora la posibilidad de otpimizar otros

criterios, como es el que el transporte de las mercancias sea tan beneficioso para los transportistas como sea posible.

El modelo propuesto no es mas que una primera aproximacion a este tipo de problemas, que puede generalizarse de forma sencilla en una gran gama de sentidos, pero que en cualqiera de ellos se adecuaria inmediatamente al marco general aqui expuesto. El estudio de dichas generalizaciones, asi como sus respectivas particularizaciones a casos practicos concretos, se deja para posteriores trabajos.

**Referencias**


[1] R.E. Bellman y L.A. Zadeh (1970). Decision Making in a Fuzzy Environment. Man. Sci. 17 (B), 4, 141-164.

[2] J.M. Cadenas y J.L. Verdegay (1997). Using Ranking Functions in Multiobjective Fuzzy Linear Programming. Proc. of the Seventh IFSA World Congress, Praga.

[3] Ch. Carlsson y R. Fuller (1996). Fuzzy Multiple Criteria Decision Making. Recent Developments. Fuzzy Sets and Systems 78, 139-153.

[4] V. Chankong y Y.Y. Haimes (1983). Multiobjective Decision Making. Theory and Methodology. North-Holland.

[5] M. Delgado, J. Kacprzyk, J. L. Verdegay y M.A. Vila, Eds., (1994): Fuzzy Optimization: Recent Advances, Physica-Verlag.

[6] H.J. Greenberg (1995). Mathematical Programming Models for Environmental Quality Control. Operations Research 43, 4, 578-623.

[7] E.O. Heady y G.F. Vocke, Eds., (1992). Economic Models of Agricultural Land Conservation and Environmental Improvement. Iowa State Univ. Press, Ames, Iowa .

[8] H.W. Kuhn y A.W. Tucker (1951). Nonlinear Programming. En Proc. of the 2nd Berkeley Symposium on Mathematics, Statistics and Probability. University of California Press, Berkeley, 481-492.

[9] V. Pareto (1896). Cours d'Economie Politique. Rouge, Laussanne, Switzerland.

[10] M. Sakawa (1993). Fuzzy Sets and Interactive Multiobjective Optimization. Applied Information Technologies Series. Plenum Press.

[11] R. Slowinski y J. Teghem, Eds., (1990). Stochastic versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty. D. Reidel Publ. Co.

[12] X. Wang y E. Kerre (1996). On the classification and the Dependencies of the Ordering Methods. En D. Ruan (Ed.): Fuzzy Logic Foundations and Industrial Applications. International Series in Intelligent Technologies. Kluwer, 73-90.

[13] L.A. Zadeh (1963). Optimality and Nonscalar-Valued Performance Criteria. IEEE Trans. on Automatic Control AC-8, 59-60.

[14] H.J. Zimmermann (1978). Fuzzy Programming and Linear Programming with Several Objective Functions. Fuzzy Sets and Systems 1,1, 45-55.

# EL $\lambda$-CALCULO-B

**Daniel Sánchez Alvarez**
Departamento de Informática y Sistemas
Universidad de Murcia
E-mail: daniel@dif.um.es

## Resumen

En el desarrollo de cualquier lenguaje de programación la parte más dificultosa es el establecimiento de la semántica. Dificultad que se incrementa si el lenguaje a desarrollar debe manejar datos borrosos. Por ello el objetivo que vamos a abordar en este trabajo es modificar el $\lambda$-cálculo de Church para dar cuenta de la teoría de Zadeh. Una vez conseguido este objetivo podremos establecer la semántica denotacional de un lenguaje de programación borroso.

## 1 Introducción

El objetivo que vamos a abordar en este trabajo es la de añadir al $\lambda$-cálculo de Church, los elementos mínimos necesarios para dar cuenta de la teoría de Zadeh. Para ello conservaremos del $\lambda$-cálculo su notación sintáctica así como su definición de variable libre y ligada, etc..., la modificación que realizaremos será la de exijir que todo $\lambda$-término debe llevar anejo un elemento de un cierto conjunto $D$. Este conjunto será la materia de nuestro estudio. De partida solo postulamos que $D$ no es vacío, y tanto al definir conceptos tales como sustitución, así como al establecer los esquemas de axiomas y reglas, que nos permiten establecer la corrección de una fórmula, deberemos exigirle que esté dotado de ciertas operaciones y dichas operaciones reunan cierta propiedades.

El cálculo resultante será una variante del lambda cálculo etiquetado de Hyland y Wadsworth [1], en él cada término se etiqueta con un número natural y se utiliza como una herramienta par examinar los $\lambda$-modelos de $D_\infty$ y $P_\omega$. Posteriormente Levy [8] generaliza el lambda cálculo etiquetado al considerar la etiqueta de cada término como una lista y se utiliza para estudiar la optimización de las reducciones. Trabajos similares a este, pero utilizando otros paradigmas de computabilidad son [2, 9].

Como en todo sistema formal [3] los términos son representables por medio de árboles, una forma sencilla de visualizar nuestro cálculo, al que en adelante llamaremos lambda cálculo-b, será etiquetar las ramas con los elementos de $D$, de manera que los $\lambda$-b-términos tendrán dos partes: a) la que llamaremos parte-$\lambda$ perteneciente al $\lambda$-cálculo clásico y b) la que llamaremos parte-b que será una construccion realizada sobre el conjunto $D$. Por tanto al representar un término genérico $\mathbf{T}$, utilizaremos:

- El elemento del $\lambda$-cálculo clásico representado por $t$
- La lista sobre $D$ resultante de recorrer en profundidad el árbol, la representaremos por $\ell(\mathbf{T})$.

Además, cuando necesitemos manipular o poner de manifiesto el primer elemento de $\ell(\mathbf{T})$, escribiremos $\ell(\mathbf{T})$ como $c(\mathbf{T}) \cdot r(\mathbf{T})$, donde $c(\mathbf{T})$ es el primer elemento de $\ell(\mathbf{T})$ y $r(\mathbf{T})$ es el resto. De esta forma $\mathbf{T} = [t, c(\mathbf{T}) \cdot r(\mathbf{T})]$. Pasamos pues a definir formalmente nuestro cálculo.

## 2 Variables y Términos

Las variables serán elementos de un conjunto numerable, las representaremos por $x_1, x_2, \ldots$ o por $x, y, z, \ldots$

Las reglas de construcción de términos serán

a) Para cada $\gamma \in D$ existe una regla que representaremos por $I_\gamma$ tal que si $x$ es una variable entonces $I_\gamma(x) = [x, \gamma]$ es un término

b) Para cada $\gamma \in D$ existe una regla que representaremos por $F_\gamma$ tal que si $x$ es una variable y $T = [t, \ell(T)]$ es un término, entonces $F_\gamma(x, T) = [\lambda x.t, \gamma \cdot \ell(T)]$ es un término.

c) Para cada $\gamma \in D$ existe una regla que representaremos por $G_\gamma$ tal que si $T = [t, \ell(T)]$ y $U = [u, \ell(U)]$, son términos, entonces $G_\gamma(T, U) = [(t\ u), \gamma \cdot \ell(T) \cdot \ell(U)]$ es un término.

## 3 La sustitución y $\alpha$−equivalencia

Una de las diferencias de nuestro cálculo respecto del lambda cálculo clásico es que las variables no son términos. Las variables aquí solo se utilizan para señalar ligaduras en las abstracciones. Por ello hay que revisar el concepto y definición de sustitución.

Entenderemos la sustitución como la sustitución de un término cuya parte-$\lambda$ sea una variable por otro término. Debemos dejar constancia de la parte-b del término sustituido. Para ello proponemos que en $D$ exista una función $s$ de $D \times D$ en $D$. De esta manera la definición de sustitución sería:

**Definición**.- Sean $U$ y $T$ términos y $x$ una variable, se define $U\langle T/x\rangle$ como el resultado de sustituir $T$ en las ocurrencias libres de $x$ en $U$

1. Si la parte-$\lambda$ de $U$, $u$, es una variable
   (a) Si $u$ es igual a $x$
   $$[x, \alpha]\langle [t, c(T) \cdot r(T)]/x\rangle = [t, s(\alpha, c(T)) \cdot r(T)]$$
   (b) Si $u$ es distinto de $x$
   $$U\langle T/x\rangle = U$$
2. Si $U$ es un término de la forma
   $$[(w\ v), \gamma \cdot \ell(W) \cdot \ell(V)]$$
   es decir la última operación de construcción de $U$ fué una operación $G_\gamma$, entonces
   $$U\langle T/x\rangle = [(w\langle t/x\rangle\ v\langle t/x\rangle), \gamma \cdot \alpha \cdot \beta]$$
   donde $\alpha = \ell(W\langle T/x\rangle)$ y $\beta = \ell(V\langle T/x\rangle)$
3. Si $U$ es un término de la forma
   $$[\lambda y.w, \gamma \cdot \ell(W)]$$
   es decir la última operación de construcción de $U$ fué una operación $F_\gamma$, entonces
   $$U\langle T/x\rangle = F_\gamma(y, W\langle T/x\rangle) = [\lambda y.w\langle t/x\rangle, \gamma \cdot \ell(W\langle T/x\rangle)]$$
   No es necesario distingir casos suponiendo que $x \neq y$ y que $y$ no aparece libre en $T$ pues utilizamos el convenio de Barendregt sobre variables, ver [1].

**Definición**.- Dados dos elementos de $L$ decimos que $U \equiv_\alpha U'$ si sus partes-$\lambda$ son $\alpha$-equivalentes en el sentido clásico y sus partes-b coinciden.

**Teorema**.- La relación $\equiv_\alpha$ es una relación de equivalencia compatible en $L$. Llamaremos $\Lambda$ a $L/\equiv_\alpha$

### 3.1 Propiedades de la sustitución

De todas la propiedades de la sustitución la única interesante para nuestro estudio sobre $D$ es la siguiente

> Si $M, N$ y $P$ son términos, $x \neq y$ y $x$ no es libre en $P$, entonces
> $$M\langle N/x\rangle\langle P/y\rangle = M\langle P/y\rangle\langle N\langle P/y\rangle/x\rangle$$

Pues para que se verifique hemos de admitir que la función $s$ ha de ser tal que para cualesquiera valores $a, b, c \in D$

$$s(a, s(b, c)) = s(s(a, b), c) \qquad (1)$$

## 4 La $\beta$-reducción

Siguiendo a [1] partiremos de una relación, a la que llamaremos $\rho$, esta tendrá la propiedad del diamante y la $\beta$-reducción, definida como el cierre transitivo de $\rho$, también tendrá dicha propiedad.

Nuestra preocupación se dirigirá a la parte-$b$, pues deseamos que ella también tenga la propiedad del diamante. Para ello vamos a introducir unas funciones, la $f$ y la $g$, que aparecerán cada vez que hagamos una $\beta$-reducción.

### 4.1 La relación $\rho$

Dados dos elementos $M$ y $N$ de $\Lambda$ decimos que $M$ y $N$ están relacionados, y lo representaremos por $M\ \rho\ N$, si $N$ se obtiene aplicando un número finito de veces las siguientes reglas:

**1)** Para todo $T \in \Lambda$ $\quad T\ \rho\ T$

**2)** Si $T\ \rho\ T'$ entonces
$$[\lambda x.t, \alpha \cdot \ell(T)]\ \rho\ [\lambda x.t', \alpha \cdot \ell(T')]$$

**3)** Si $T\ \rho\ T'$ y $U\ \rho\ U'$ entonces
$$[(u\ t), \beta \cdot \ell(U) \cdot \ell(T)]\ \rho\ [(u'\ t'), \beta \cdot \ell(U') \cdot \ell(T')]$$

**4)** Si $T\ \rho\ T'$ y $U\ \rho\ U'$ entonces
$$[(\lambda x.u\ t), \alpha \cdot \beta \cdot c(U) \cdot r(U) \cdot c(T) \cdot r(T)]\ \rho\ [u'\langle t'/x\rangle, f(\alpha, \beta, c(U'\langle Z/x\rangle)) \cdot r(U'\langle Z/x\rangle)]$$

donde $Z = [t', g(\beta, c(T')) \cdot r(T')]$ y siendo $f$ una función de $D \times D \times D$ en $D$ y $g$ una función de $D \times D$ en $D$. Es decir $Z$ y $T'$ coinciden en su parte-$\lambda$, la única diferencia que existe entre ambas es que $c(Z) = g(\beta, c(T'))$, siendo $r(Z) = r(T')$.

El motivo que nos lleva a formular esta última regla es el siguiente: cuando tenemos un término de la forma $[((\lambda x.u)\ t), \alpha \cdot \beta \cdot \ell(U) \cdot \ell(T)]$ , donde $\alpha$ corresponde a la aplicación y $\beta$ corresponde a la abstracción, al llevar a cabo la $\rho$-relación, desde el punto de vista del $\lambda$-cálculo clásico, ambas desaparecen, nos quedamos sólo con el término $u\langle t/x\rangle$. En nuestro cálculo deseamos que en la parte-b quede constancia de haber realizado dicha $\rho$-relación, en la que han intervenido $\alpha$ y $\beta$. Por ello, proponemos, a fin de mantener la información, que la $\rho$-relación se realice de manera que afecte:

A todas la sustituciones (puede que no haya ninguna sustitución o puede que haya múltiples) que hagamos de $x$ por $T$ en $U$, para ello, si $\phi$ es el primer elemento-b de $T$ lo sustituimos por el resultado de aplicar una cierta función $g$ a $\beta$ y $\phi$.

Globalmente a $U$ y que sea independiente de que $x$ aparezca en $U$ o no, para ello, tomamos el primer elemento de la lista de $U$ después de haber efectuado las sustituciones del punto anterior y lo modificamos, por medio de una cierta función $f$ que dependerá de $\alpha, \beta$ y de dicho primer elemento.

Es decir realizaremos dos transformaciones:

1. Transformar el término por el cual vamos a sustituir la variable ligada

2. Transformar el término resultante de realizar la sustitución anterior.

Estas transformaciones las realizamos por medio de las funciones $f$ y $g$.

### 4.2 El $\lambda$-cálculo-$b$ y la propiedad del diamante

Deseamos que el cálculo$-b$ verifique el teorema de Church-Rosser. Siguiendo, paso a paso, la demostración de que la relación $\rho$ verifica la propiedad del diamante, cada etapa nos suministrará condiciones sobre $f, g$ y $s$ para que la parte-$b$ de nuestro cálculo cumpla también la propiedad deseada. El resultado será que para cualesquiera los valores de $a$, $b$, $c$, $d$ y $e \in D$, $s$, $f$ y $g$ deben ser tales que se verifique:

$$s(f(a,b,c),d) = f(a,b,s(c,d)) \quad (2)$$
$$s(a,f(b,c,d)) = f(s(a,b),c,d) \quad (3)$$
$$s(g(a,b),c) = g(a,s(b,c)) \quad (4)$$
$$f(g(a,b),c,d) = g(a,f(b,c,d)) \quad (5)$$
$$f(a,b,f(c,d,e)) = f(f(a,b,c),d,e) \quad (6)$$

## 5 El $\lambda\delta$-cálculo

Sea $X \subseteq \Lambda$ un conjunto de formas normales cerradas. Generalmente se toma $X \subseteq C$, y sea $\mathfrak{f} : \mathfrak{X} \longrightarrow \Lambda$ una función "definida externamente". En el lambda cálculo clásico, a fin de representar $\mathfrak{f}$, se añade una regla llamada $\delta$-regla definida de la siguiente forma: 1) Seleccionamos una constante de $C$ y se le dá una nombre, por ejemplo $\delta$ y 2) Las reglas de contracción $\delta M_1 \ldots M_k \longrightarrow \mathfrak{f}(\mathfrak{M}_1, \ldots, \mathfrak{M}_t)$, donde $M_1, \ldots, M_k \in X$ se añaden a las reglas del lambda cálculo.

Para una $\mathfrak{f}$ dada la fórmula anterior no es una regla de contracción es un esquema de reglas. La $\delta$-reducción no es una noción absoluta pues depende de la elección de $\mathfrak{f}$. La nueva sintáxis se introduce por medio del reemplazamiento. Su uso permite alcanzar una $\lambda$-expresión pura, es decir una $\lambda$-expresión donde no aparecen nombres añadidos, tras un número finito de pasos que envuelven simples reemplazamientos. De esta forma no es necesario modificar el $\lambda$-cálculo.

Necesitamos introducir este proceso en nuestro cálculo, para ello definimos una función $\mathcal{V}$ sobre los términos normales cerrados $\mathbf{T}$ tal que $\mathcal{V} : \ell(\mathbf{T}) \to D$. A partir de ella estableceremos una relación de equivalencia entre los términos que nos permitirá llevar a cabo las $\delta$-reducciones.

Aún cuando hasta este momento la parte-b la hemos representado por una lista, ha sido por comodidad y porque su estructura era deducible a partir de la parte-$\lambda$. Pero ahora, cuando vamos a estudiar $\mathcal{V}$, hemos de recuperar dicha estructura. La parte-b de un $\lambda$-término obedece a la siguiente gramática

$$B \to \lambda LB \mid @LBB \mid L$$

$$L \to D \mid s(D,L) \mid f(L,L,L) \mid g(L,L)$$

Es decir aparecen: los elementos de $D$ que han intervenido en su construcción, los símbolos $\lambda$ y @ que le dotan de una estructura y las operaciones $s, f$ y $g$ que guardan la historia de las tranformaciones.

Si aceptamos la existencia en $D$ de un elemento neutro para la operación $s$, al que llamaremos $n$, y haciendo uso de los resultados obtenidos anteriormente, introducimos la siguiente notación:

- Usaremos $+$ para representar de forma infija la operación $s$.
- Teniendo en cuenta 1, 2 y 3, podemos poner $f(a,b,c) = a + f(n,b,n) + c$, por lo que introduciremos el símbolo $\mathfrak{f}$ definido por $\mathfrak{f}(a) = f(n,a,n)\ \forall a \in D$
- Teniendo en cuenta 1 y 4 podemos poner $g(a,b) = g(a,n) + b$, por lo introduciremos el símbolo $\mathfrak{g}$ definido por $\mathfrak{g}(a) = g(a,n)\ \forall a \in D$
- $\lambda$ es una operación binaria prefija, utilizaremos $\odot$ para su representación infija
- @ es una operación ternaria prefija, utilizaremos $\oplus$ para representar la operación infija entre el primero y el resultado de operar el segundo y el tercero, operación que en forma infija representaremos por $\otimes$.

En el $\lambda$-cálculo clásico, ver [4], se parte de que los naturales son objetos de una clase inductiva generada a partir de un elemento inicial, el 0, por medio de una función sucesor, indicada por $'$, de tal forma que los números forman una secuencia de elementos distintos $0, 0', 0'', \ldots$. Cualquier secuencia generada sistemáticamente serviría para este propósito; ya que es necesario que exista un elemento inicial, el $d_0$, se tomaría como el cero, y el iterador $S_d$ como la función sucesor. Lo único que debemos comprobar es que, en cada aplicación del iterador, el elemento resultante es distinto de los generados anteriormente. En estas condiciones cualquier sistema numérico $\mathbf{d}$ está determinado por $d_0$ y $S_d$, por lo que se escribe $\mathbf{d} = (d_0, S_d)$.

En nuestro caso un sistema numérico estará formado por dos familias:

- La familia $\mathfrak{d}_o$, de términos del lambda cálculo-b, tal que su parte $\lambda$ coincide con $d_0$ y su parte-b es cualquier lista adecuada a la estructura de $d_0$
- La familia $\mathfrak{S}_\mathfrak{d}$, de términos del lambda cálculo-b, tal que su su parte $\lambda$ coincide con $S_d$ y su parte $b$ es cualquiera adecuada a la estructura de $S_d$

Dado un sistema numérico $(\mathfrak{d}_o, \mathfrak{S}_\mathfrak{d})$ y la función $\mathcal{V}$ existirá una interrelación entre ellos si queremos que nuestro cálculo tenga unas determinadas propiedades. Por ejemplo sería deseable que :

A) Si $d_0$ es cualquier elemento de la familia $\mathfrak{d}_o$ y $S_d$ es el elemento de la familia $\mathfrak{S}_\mathfrak{d}$ tal que todos los elementos que aparecen en su parte-$b$ son iguales a $n$, el neutro de $s$, entonces

$$\mathcal{V}(d_0) = \mathcal{V}([\overbrace{(S_d(S_d(\ldots(S_d}^{m}\ d_0))\ldots))), \overbrace{n \cdot n \ldots \cdot n}^{m} \cdot \ell(d_0)]$$

B) Si $\mathbf{I}$ es la Identidad $\mathbf{I} = [\lambda x.x, n \cdot n]$

$$\mathcal{V}((\overbrace{(S_d\ (S_d\ (\ldots\ (S_d}^{m}\ (\mathbf{I}\ d_0))\ldots)), \alpha \cdot \overbrace{n \cdot n \ldots \cdot n}^{m} \cdot \ell(d_0) =$$

$$\mathcal{V}((\mathbf{I}\ \overbrace{S_d\ (S_d\ (\ldots\ (S_d}^{m}\ d_0), \overbrace{n \cdot n \ldots \cdot n}^{m} \cdot \alpha \cdot \ell(d_0)$$

$\forall m \in D$

Lo cual nos permitirá descartar funciones $\mathcal{V}$ arbitrarias así como determinar equivalencias en los términos cuyas partes-$\lambda$ coincidan.

Hemos supuesto que la función $\mathcal{V}$ depende de $\lambda, @, s, \mathfrak{f}$ y $\mathfrak{g}$. El hacer que se verifiquen los puntos A) y B) nos dará un camino para poder simplificar esta dependencia. En particular:

- Para que se cumpla A) $\mathcal{V}(\mathbf{0}) = \mathcal{V}(\mathbf{1})$ siendo:
  - $\mathcal{V}(\ell(\mathbf{0})) = a \odot (b \odot c)$
  - $\mathcal{V}(\ell(\mathbf{1})) = A \odot (n \odot (n \oplus (n \otimes (W + n)))$. Donde
    - $A = e + \mathfrak{f}(n) + n$
    - $W = n + \mathfrak{f}(Z) + c + \mathfrak{g}(Z)$
    - $Z = n + \mathfrak{f}(n + \mathfrak{g}(n) + a) + b$
- Para que se cumpla B) se debe cumplir que el término $\mathbf{T} \equiv [(\mathbf{I}\ \mathbf{I}), n] = [\lambda x.x, n + \mathfrak{f}(n) + \mathfrak{g}(n) + n \cdot n]$ verifique que $\mathcal{V}(\ell(\mathbf{T})) = \mathcal{V}(\ell(\mathbf{I}))$ para lo cual es necesario que $\mathfrak{f}(n) + \mathfrak{g}(n) = n$.

Esto nos lleva a imponer las siguientes condiciones:

- la función $\lambda$ coincida con $s$
- $\mathfrak{f}(b) = b\ \forall b \in D$, es decir $f(n,b,n)$ es la función identidad sobre el segundo parámetro
- $\mathfrak{g}(b) = n\ \forall b \in D$, es decir $g(b,n)$ es la función constante al neutro de $s$
- $\oplus$ y $\otimes$ deben ser funciones cuyo neutro coincida con el neutro de $+$ (ejemplo la función min y producto o la función máx y suma reunen estas condiciones)

## 6 La relación $\mathcal{R}$

Una vez fijadas las funciones $+, \oplus$ y $\otimes$ queda fijada la función de valuación $\mathcal{V}$, la representaremos por $\mathcal{V}_{(+,\oplus,\otimes)}$, o por $\mathcal{V}$ cuando no haya lugar a confusión, la cual a su vez establece una relación en el conjunto de los términos cerrados en forma normal que definimos a continuación.

**Definición.**- Decimos que los términos $A = [a, \ell(A)]$ y $B = [b, \ell(B)]$ están $_{(+,\oplus,\otimes)}$-relacionados y lo representaremos por

$$A \, \mathcal{R}_{(+,\oplus,\otimes)} \, B$$

si y solo si $a \equiv b$ y $\mathcal{V}_{(+,\oplus,\otimes)}(\ell(A)) = \mathcal{V}_{(+,\oplus,\otimes)}(\ell(B))$

Para que $\mathcal{R}_{(+,\oplus,\otimes)}$ sea una relación compatible es suficiente admitir que $\oplus$ y $\otimes$ coinciden con $s$ y que esta es conmutativa. En tal caso podemos definir el representante canónico de un término cerrado de la siguiente forma

**Definición.**- Dado un término $A = [a, \ell(A)]$ tal que $\mathcal{V}(\ell(A)) = \phi$, un representate de la clase de equivalencia será $[a, \{\phi\}]$ o simplemente $[a, \phi]$ donde $\{\phi\}$, es una lista no vacia cuya primer elemento es $\phi$ y los restantes son iguales a $n$, el elemento neutro de $+$.

## 7 Conclusiones

Con el objetivo de introducir los conjuntos borrosos en el lambda cálculo hemos modificado el concepto de $\lambda$-término, añadiendole una etiqueta en la que se refleje su grado de pertenencia a un determinado conjunto. Ello nos ha llevado a reformular la sustitución y la $\beta$-reducción, a fin de dar cuenta de como se manipulan dichas etiquetas. Esta ampliación contiene al $\lambda$-cálculo clásico como un caso particular. Las etiquetas deben pertenecer a un conjunto $D$ en el cual debemos tener definidas tres operaciones: dos de ellas $\mathfrak{f}$ y $\mathfrak{g}$ triviales y una operación binaria $s$ con las siguientes propiedades:

- Debe tener un neutro
- Debe ser asociativa
- Debe ser conmutativa

en particular si $D = [0, 1]$ $s$ puede ser cualquier t-norma o t-conorma.

A partir de estos elementos podemos diseñar un lenguaje de programación con el que dar cuenta del paradigma borroso. En [9] esto se lleva a cabo usando las gramáticas con atributos para describir la semántica. Nosotros utilizaremos el método denotacional [6], por lo que nos queda por realizar, entre otras, las siguientes tareas:

a) Representación y operaciones básicas de los números borrosos . Para ello utilizaremos junto con [10, 7, 5] los sistemas númericos $(\mathfrak{d}_o, \mathfrak{S}_\mathfrak{d})$ y la relación $\mathcal{R}$

b) Enriquecimiento del $\lambda$-cálculo-b, para facilitar la tarea de programación, con estructuras tales comos listas y comandos usuales en los lenguajes de programación.

c) Como consecuencia de los dos puntos anteriores replantearnos el significado de algunas construcciones clásicas de los lenguajes tales como el condicional.

## Referencias


[1] H.P. Barendregt. *The Lambda calculus its syntax and semantics.* North-Holland, Amsterdam, 1985.

[2] Buenaventura Clares. Una introducción a la W-calculabilidad. Tesis Doctoral, Universidad de Granada, 1983.

[3] Haskell B. Curry and Robert Feys. *Lógica combinatoria*, volume 1. Editorial Tecnos, S.A., Madrid, 1967.

[4] Haskell B. Curry, J. Roger Hindley, and Jonathan P. Seldin. *Combinatory Logic*, volume 2. North-Holland, Amsterdam, 1972.

[5] D. Dubois and H. Prade. *Fuzzy Set and System: Theory and applications.* Academic Press, New York, 1980.

[6] J.E.Stoy. *Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory.* MIT Press, Cambridge.Mass., 1977.

[7] Arnold Kaufmann and Madan M. Gupta. *Introduction to fuzzy arithmetica. Theory and Applications.* Van Nostrand Reinhold Company, New York, 1985.

[8] Jean-Jacques Lévy. Réductions correctes et optimales dans le lambda-calculo. Tesis Doctoral, Universite Paris VII, 1978.

[9] Rafael Morales. Calculabilidad difusa mediante gramáticas. Diseño de un lenguaje imperativo de programación difusa. Tesis Doctoral, Universidad de Málaga, 1991.

[10] L.A. Zadeh. Fuzzy sets. *Inform. Control*, 8:338–353, 1965.

# Satisfiability Testing in Finitely-Valued Logics*

**Felip Manyà**
Departament d'Informàtica
Universitat de Lleida (UdL)
Plaça Víctor Siurana, 1
25003 Lleida (Spain)
e-mail:felip@eup.udl.es

**Gonzalo Escalada-Imaz**
Artificial Intelligence Research Institute
IIIA-CSIC
Campus UAB
08193 Bellaterra (Spain)
e-mail:gonzalo@iiia.csic.es

## Abstract

We present a proof procedure for solving the satisfiability problem in any finitely-valued propositional logic. On the one hand, our proof procedure takes as input signed cnf-formulas since the satisfiability problem in any finitely-valued logic can be translated into a satisfiability problem over signed cnf-formulas in polynomial time. On the other hand, it can be understood as an extended Davis-Putnam procedure for dealing with signed cnf-formulas and equipped with an optimized multiple-valued branching rule based on the concept of maximal set of truth values, which we introduce in this paper.

**Keywords:** Multiple-valued logic, satisfiability problem, signed cnf-formula.

## 1 Introduction

The satisfiability problem in classical propositional logic (i.e. to determine whether a two-valued formula is satisfiable) is known for being the original NP-complete problem [3]. In the last years, it has attracted the interest of several research communities (Operations Research, Complexity Theory, Computational Logic, ...), which have published a wealth of papers reporting algorithms for solving this problem and, because of the state of maturity reached, even competitions for testing them in a wide variety of problems have been organized [2, 11]. Among the most competitive algorithms developed so far are those based on the so-called Davis-Putnam procedure [5, 4, 12].

In this paper we investigate the satisfiability problem in finitely-valued propositional logics taking the logic of signed cnf-formulas (cf. Section 2) and the Davis-Putnam procedure as a starting point.

Signed cnf-formulas are important since an instance of the satisfiability problem in any finitely-valued logic can be translated into a satisfiability problem over signed cnf-formulas in polynomial time [8]. Thus, satisfiability checking procedures for signed cnf-formulas as the one described in this paper suffice for determining whether a finitely-valued formula is satisfiable, provided that a satisfiability equivalent signed cnf-formula has been derived using, for example, the methods described in [1, 8].

Our proof procedure can be understood as a generalization of the classical Davis-Putnam procedure. We think that this is a promising approach since the semantics of signed cnf-formulas is classical above the literal level. Therefore, for extending two-valued proof procedures to signed cnf-formulas with success we should concentrate on the decisions made on the literal level, since it is there where the differences arise. This fact leads us to define an optimized multiple-valued branching rule based on the concept of maximal set of truth values, which we introduce in this paper. This concept is aimed at formalizing the minimum number of assignments of truth values to a given propositional atom that a satisfiability checking procedure over signed cnf-formulas has to test.

See [10] for an overview of proof methods for testing the satisfiablility of multiple-valued formulas. Satisfiabibility checking procedures for subclasses of signed cnf-formulas have been introduced in [6, 9, 13]

This paper is organized as follows: In Section 2 we define the logic of signed cnf-formulas; in Section 3 we present the Davis-Putnam procedure; in Section 4 we define the concept of maximal set of truth values and an optimized signed branching rule; in Section 5 we offer a satisfiability checking procedure for signed cnf-formulas and prove its correctness. We finish the paper with some concluding remarks.

Research partially supported by the projects TIC96-1038-C04-03 and TIC96-0721-C02-02 funded by the CICYT.

## 2 Signed cnf-formulas

Signed formulas have been widely used in the subfield of automated theorem proving in multiple-valued logics (cf. [7]). In particular, several complete resolution-based calculi have been developed for the subclass of signed cnf-formulas we introduce in this section [1, 9, 14]. Next, we define their basic concepts of syntax and semantics.

**Definition 1 signed cnf-formulas** *Let $N$ be a finite set of truth values, $S$ a subset of $N$ ($S \subseteq N$) and $p$ a propositional atom. An expression of the form $S{:}p$ is a signed literal and $S$ is its sign. A signed clause is a finite set of signed literals. A signed cnf-formula is a finite set of signed clauses.*

**Definition 2 satisfiability** *An interpretation $\mathbf{I}$ is a mapping that assigns to every propositional atom a value of the set of truth values $N$. An interpretation $\mathbf{I}$ satisfies a signed literal $S : p$ iff $\mathbf{I}(p) \in S$. An interpretation $\mathbf{I}$ satisfies a signed clause iff $\mathbf{I}$ satisfies at least one of its signed literals. A signed cnf-formula $\Gamma$ is satisfiable iff there exists at least one interpretation that satisfies all the signed clauses in $\Gamma$. A signed cnf-formula that is not satisfiable is unsatisfiable. The empty signed clause is always unsatisfiable and the empty signed cnf-formula is always satisfiable.*

It is worth noticing that the main semantic difference between classical and signed cnf-formulas arise in the definition of interpretation, where the number of truth values is greater than two. Then, the concept of satisfiability of signed clauses and cnf-formulas is the same as the classical one, but with respect to multiple-valued interpretations. It is in this sense that we say that the semantics of signed formulas is classical above the literal level.

Observe that signed cnf-formulas are clause forms which are independent of the multiple-valued logic we are dealing with, except for the number of truth values. Such greater generality is important since, in many multiple-valued logics, it is difficult to find a clause form in the object language of the logic.

## 3 The Davis-Putnam procedure

As our intention is to extend the Davis-Putnam procedure to the context of signed cnf-formulas, we shall first analyze carefully the details of the two-valued version.

The Davis-Putnam procedure for propositional cnf-formulas is based on two inference rules known as unit resolution and branching rule:

**unit resolution**

$$\frac{L \qquad \overline{L} \vee C}{C}$$

**branching rule**

$$\frac{}{p \mid \neg p}$$

The algorithm automatizes the application of the previous rules as follows: First, it applies unit resolution until either the empty clause is deduced or a saturation state is reached. Actually, what it does for each unit clause $\{L\}$ of a given cnf-formula $\Gamma$ is to remove all clauses containing the literal $L$ and, from the remaining clauses, delete all occurrences of the literal $\overline{L}$ (this process is called *unit resolution*). This way, after several applications of unit resolution, the algorithm produces a simpler cnf-formula $\Gamma'$, without unit clauses, from the cnf-formula $\Gamma$. Second, it takes a literal $L'$ from $\Gamma'$ and reduces the problem of testing the satisfiability of $\Gamma'$ to two new subproblems: to know whether $\Gamma' \cup \{L'\}$ is satisfiable or $\Gamma' \cup \{\overline{L'}\}$ is satisfiable (this process is called *branching*). Observe that the new cnf-formulas contain a unit clause and unit resolution can be applied to them. These steps are repeated and the algorithm stops when some subproblem contains the empty formula (then, the input cnf-formula is satisfiable) or all the subproblems contain the empty clause (then, the input cnf-formula is unsatisfiable). This algorithm is complete for the satisfiability problem for classical cnf-formulas. A pseudo-code version of the Davis-Putnam procedure is displayed in Figure 1.

This procedure can be also understood as the construction of a proof tree: Nodes are labelled by cnf-formulas and the root node contains the input cnf-formula $\Gamma$. Before expanding a node, function unit-resolve applies repeatedly unit resolution to the cnf-formula of the node. Then, a new cnf-formula $\Gamma'$ is obtained and pick-literal selects a literal $L$ of $\Gamma'$. Two branches, labelled $\Gamma' \cup \{L\}$ and $\Gamma' \cup \{\overline{L}\}$, are created and their leaf nodes contain the cnf-formulas obtained after applying function unit-resolve to $\Gamma' \cup \{L\}$ and $\Gamma' \cup \{\overline{L}\}$, respectively. The proof tree is created in a depth-first manner and the procedure backtracks in those nodes that contain the empty clause. The search terminates when the empty formula is deduced (in this case, the input cnf-formula $\Gamma$ is satisfiable) or until the procedure backtracks to the root node (in this case, all the leaf nodes contain the empty clause and the input cnf-formula $\Gamma$ is unsatisfiable).

## 4 Maximal set of truth values

Before extending the Davis-Putnam procedure to the setting of signed cnf-formulas, we shall first analyze

```
function dp-sat (Γ: set of clause) :boolean;
var L: literal;
begin
  Γ := unit-resolve(Γ);
  if Γ = ∅ then return(true);
  if □ ∈ Γ then return(false);
  L := pick-literal(Γ);
  if dp-sat (Γ ∪ {L}) then
    return(true)
  else
    return(dp-sat(Γ ∪ {L̄}))
end;

function unit-resolve (Γ: set of clause) : set of clause;
var L: literal;
var C: clause;
begin
  while ∃{L} ∈ Γ and □ ∉ Γ do
  begin
    Γ := {C | L ∉ C ∈ Γ};
    Γ := {C − {L̄} | C ∈ Γ}
  end
return(Γ)
end;
```

Figure 1: Davis-Putnam procedure for classical cnf-formulas

how to extend the branching rule in such a way that the number of branches for each node of the proof tree is as small as possible.

In the two-valued case, the branching rule states that a cnf-formula $\Gamma$ is satisfiable iff $\Gamma \cup \{p\}$ is satisfiable or $\Gamma \cup \{\neg p\}$ is satisfiable, where $p$ is a propositional atom occurring in $\Gamma$. In the multiple-valued case, a straightforward generalization of the two-valued branching rule would be as follows: A signed cnf-formula $\Gamma'$ is satisfiable iff $\Gamma' \cup \{\{0\}:p\}$ is satisfiable or $\Gamma' \cup \{\{\frac{1}{n-1}\}:p\}$ is satisfiable or ... or $\Gamma' \cup \{\{\frac{n-2}{n-1}\}:p\}$ is satisfiable or $\Gamma' \cup \{\{1\}:p\}$ is satisfiable, where $p$ is a propositional atom occurring in $\Gamma'$ and $N = \{0, \frac{1}{n-1}, \ldots, \frac{n-2}{n-1}, 1\}$ is the set of truth values.[1] Therefore, if we want to expand a node of a proof tree, we should develop $|N|$ branches, where $|N|$ denotes the cardinality of $N$. However we shall show that, in most cases, it is enough that the branching rule considers only a subset of $N$.

In the following we will assume, without loss of generality, that a signed cnf-formula has the property that a propositional atom occurs at most once in each clause. If this is not the case and we have a clause of the form $S_1 : p \vee \cdots \vee S_n : p \vee D$, then we replace it by the clause $(S_1 \cup \cdots \cup S_n) : p \vee D$. Observe that both clauses are logically equivalent.

From now on, we denote by $N_\Gamma^p$ the set formed by the truth values of $N$ that appear in a signed cnf-formula $\Gamma$ in literals of the form $S:p$.

Given a signed cnf-formula $\Gamma$ and a propositional atom $p$ occurring in $\Gamma$, it can happen that some truth values $y_1, \ldots, y_n$ appear exactly in the same signs of literals of the form $S : p$. In this case, it suffices that the branching rule considers only one of the truth values from $y_1, \ldots, y_n$, since if a model of $\Gamma$ assigns to $p$ one truth value from $y_1, \ldots, y_n$, the interpretation that assigns to $p$ any other truth value from $y_1, \ldots, y_n$ and is identical for the remaining propositional atoms also satisfies $\Gamma$. To identify the truth values that appear exactly in the same signs of literals of the form $S:p$ we define the following equivalence relation over $N_\Gamma^p$:

$$x \approx_p y \Longleftrightarrow \forall S{:}p \in C \in \Gamma \text{ it holds that } x \in S \text{ iff } y \in S$$

Next, we define the following partial order relation over the equivalence classes of $N_\Gamma^p / \approx_p$:

$$\overline{x} \preceq_p \overline{y} \Longleftrightarrow \forall S{:}p \in C \in \Gamma \text{ if } x \in S \text{ then } y \in S$$

Observe that if $\overline{x} \preceq_p \overline{y}$, the elements in class $\overline{y}$ appear, among other signs, in all the signs where appear the elements in class $\overline{x}$. Therefore, the branching rule can ignore the elements of class $\overline{x}$, since if a model of $\Gamma$ assigns to $p$ a truth value from class $\overline{x}$, the interpretation that assigns to $p$ any truth value from class $\overline{y}$ and is identical for the remaining propositional atoms also satisfies $\Gamma$. Then, we have that it suffices that the branching rule considers only the maximal elements of the relation $\preceq_p$. Even more, it is enough that it considers only one representative element of each one of the maximal elements of $\preceq_p$ since maximal elements are equivalence classes. This is the idea behind the concept of maximal set of truth values.

**Definition 3 maximal set of signs** *Let $\Gamma$ be a signed cnf-formula, $p$ a propositional atom occurring in $\Gamma$, $\approx_p$ the equivalence relation over $N_\Gamma^p$ defined above and $\preceq_p$ the partial order relation over $N_\Gamma^p / \approx_p$ defined above. The maximal elements of $\preceq_p$ are the maximal set of signs of $p$ wrt to $\Gamma$.*

**Definition 4 maximal set of truth values** *Let $\Gamma$ be a signed cnf-formula, $p$ a propositional atom occurring in $\Gamma$ and $\{\overline{x_1}, \ldots, \overline{x_n}\}$ the maximal set of signs of $p$ wrt to $\Gamma$. A set of the form $\{x_1, \ldots, x_n\}$, where $x_1 \in \overline{x_1}, \ldots, x_n \in \overline{x_n}$, is a maximal set of truth values of $p$ wrt $\Gamma$.*

[1] We take a finite set of equidistant rational numbers between 0 and 1 as set of truth values, but the results of this paper are also valid for any other finite set.

**Example 1** *Given a signed cnf-formula $\Gamma$, assume that the signed clauses that contain signed literals of the form $S:p$ are the following ones:*

$$\{\tfrac{1}{6},1\}:p \vee D_1$$
$$\{0,\tfrac{1}{3},\tfrac{5}{6}\}:p \vee D_2$$
$$\{0,\tfrac{1}{3},\tfrac{1}{2}\}:p \vee D_3$$
$$\{0,\tfrac{1}{6},\tfrac{1}{3},1\}:p \vee D_4$$
$$\{\tfrac{1}{6},\tfrac{2}{3},1\}:p \vee D_5$$

*The equivalence classes of $N_\Gamma^p/\approx_p$ are the following ones:*

$$\overline{\tfrac{1}{6}} = \{\tfrac{1}{6},1\}$$
$$\overline{\tfrac{1}{3}} = \{0,\tfrac{1}{3}\}$$
$$\overline{\tfrac{1}{2}} = \{\tfrac{1}{2}\}$$
$$\overline{\tfrac{2}{3}} = \{\tfrac{2}{3}\}$$
$$\overline{\tfrac{5}{6}} = \{\tfrac{5}{6}\}$$

*The maximal elements of the partial order relation $\preceq_p$ are $\overline{\frac{1}{3}}$ and $\overline{\frac{1}{6}}$. Therefore, $\{\{\frac{1}{6}\},\{\frac{1}{3}\}\}$ is a maximal set of truth values of $p$ wrt $\Gamma$.*

Once we have introduced the concept of maximal set of truth values, we shall prove that a branching rule considering a maximal set of truth values instead of the whole set of truth values is complete.

**Proposition 1 signed branching rule** *Let $\Gamma$ be a signed cnf-formula, $p$ an atom occurring in $\Gamma$ and $\{x_1,\ldots,x_n\}$ a maximal set of truth values of $p$ wrt $\Gamma$. Then, $\Gamma$ is satisfiable iff for some $i$, where $1 \leq i \leq n$, we have that $\Gamma \cup \{\{x_i\}:p\}$ is satisfiable.*

*Proof:* Suppose that $\Gamma$ is satisfiable. Let $\mathbf{I}$ be a model of $\Gamma$ and $\mathbf{I}(p) = \alpha$. We distinguish two cases:

1. There is no literal $S : p \in C \in \Gamma$ such that $\alpha \in S$. Then, the value that assigns $\mathbf{I}$ to $p$ has no effect on the satisfiability of $\Gamma$. We define an interpretation $\mathbf{I}'$ which assigns any truth value $x_i \in \{x_1,\ldots,x_n\}$ to $p$ and is identical for the remaining atoms. Then, we have that $\mathbf{I}'$ satisfies both $\Gamma$ and $\Gamma \cup \{\{x_i\}:p\}$.

2. $\Gamma$ contains literals $S_1:p,\ldots,S_m:p$ such that $\alpha \in S_1,\ldots,\alpha \in S_m$. Then, $\alpha \in S_1 \cap \cdots \cap S_m$, but there exists also a maximal truth value $x_i$ such that $x_i \in S_1 \cap \cdots \cap S_m$. The interpretation $\mathbf{I}'$ which assigns $x_i$ to $p$ and is identical for the remaining atoms satisfies both $\Gamma$ and $\Gamma \cup \{\{x_i\}:p\}$.

Suppose that there exists an $i$ $(1 \leq i \leq n)$ such that $\Gamma \cup \{\{x_i\}:p\}$ is satisfiable. Then, it is clear that $\Gamma$ is satisfiable. ■

Observe that a branching rule based on the concept of maximal set of truth values can reduce greatly the size of the proof tree. For instance, when using the signed cnf-formula of Example 1 we only should expand two branches (instead of seven) if we branch on the propositional atom $p$.

Given a signed cnf-formula $\Gamma$ and a propositional atom occurring in $\Gamma$, the maximal sets of truth values of $p$ wrt $\Gamma$ can be computed in polynomial time. The cost of calculating the equivalence classes of $N_\Gamma^p/\approx_p$ and calculating the maximal elements of $\preceq_p$ with a brute force search algorithm is in $\mathcal{O}(|N|^2 \times |\Gamma|)$. Nevertheless, we believe that this complexity can be improved using appropriate data structures.

We claim that the usefulness of the concept of maximal set of truth values is not confined to define optimized branching rules, it could be very beneficial in the multiple-valued setting when we work with enumerative proof procedures (i.e. proof procedures whose principle is to assign truth values to propositional atoms) since it allows to reduce the number of truth values that have to be considered for a given propositional atom.

## 5 A satisfiability checking procedure for signed cnf-formulas

The aim of this section is to design a satisfiability checking procedure for signed cnf-formulas taking the Davis-Putnam procedure as a starting point. As explained in Section 3, this procedure relies on branching and unit resolution. Since we have already defined an optimized signed branching rule based on the concept of maximal set of truth values, our next step is to define a signed version of unit resolution.

**Proposition 2 signed unit resolution** *Let $\Gamma$ be a signed cnf-formula that contains $n$ $(n > 0)$ signed unit clauses $\{S_1 : p\} \in \Gamma,\ldots,\{S_n : p\} \in \Gamma$ with the same atom but with different sign and $S_1 \cap \cdots \cap S_n \neq \emptyset$. Let $\Gamma'$ be like $\Gamma$ except that all clauses containing a literal $S:p$ such that $S_1 \cap \cdots \cap S_n \subseteq S$ have been removed and, from the remaining clauses, all occurrences of literals $S':p$ such that $S_1 \cap \cdots \cap S_n \cap S' = \emptyset$ have been deleted. Let $\Gamma''$ be like $\Gamma'$ except that each occurrence of a literal $S'':p$ has been replaced by $(S_1 \cap \cdots \cap S_n \cap S''):p$. Then, $\Gamma$ is satisfiable iff $\Gamma''$ is satisfiable.*

*Proof:* Suppose that $\Gamma$ is satisfiable. As $\{S_1 : p\} \in \Gamma$, $\ldots$, $\{S_n : p\} \in \Gamma$, there exists a model $\mathbf{I}$ of $\Gamma$ such that $\mathbf{I}(p) \in S_1 \cap \cdots \cap S_n$. On the one hand, $\mathbf{I}$ is a model of any subset of $\Gamma$. On the other hand, if we remove the literals $S':p$ such that $S_1 \cap \cdots \cap S_n \cap S' = \emptyset$, $\mathbf{I}$ is also a model of $\Gamma'$ since $\mathbf{I}(p) \notin S'$. Now, as $\mathbf{I}(p) \in S_1 \cap \cdots \cap S_n$, we can replace in $\Gamma'$ each occurrence of a literal $S'':p$

```
function signed-sat (Γ: set of clause) :boolean;
var p: atom;
var x: truth-value;
var set: set of truth-value;
begin
 Γ := signed-unit-resolve(Γ);
 if Γ = ∅ then return(true);
 if □ ∈ Γ then return(false);
 p := pick-atom(Γ);
 set := maximal-set-truth-values(Γ,p);
 if ∃x ∈ set such that signed-sat(Γ ∪ {{x}:p}) then
   return(true)
  else
   return(false)
end;
```

```
function signed-unit-resolve (Γ: set of clause) : set of clause;
var p: atom;
var C, D: clause;
var S, S_1, ..., S_n: set of truth-value;
begin
 while ∃{S_1:p} ∈ Γ, ..., {S_n:p} ∈ Γ and □ ∉ Γ do
 begin
  Γ := {C | ∄S:p ∈ C ∈ Γ such that S_1 ∩ ··· ∩ S_n ⊆ S};
  Γ := {C − {S:p | S:p ∈ C ∧ S_1 ∩ ··· ∩ S_n ∩ S = ∅} | C ∈ Γ};
  for each C = S:p ∨ D ∈ Γ do
   C := (S_1 ∩ ··· ∩ S_n ∩ S):p ∨ D
  endfor
 end
return(Γ)
end;
```

Figure 2: A Davis-Putnam based procedure for signed cnf-formulas

by $(S_1 \cap \cdots \cap S_n \cap S'') : p$ and we have that $\mathbf{I}$ satisfies $\Gamma''$.

Suppose that $\Gamma''$ is satisfiable. By construction of $\Gamma''$, we have that all its occurrences of literals of the kind $S_i : p$ in $\Gamma''$ verify that $S_i \subseteq S_1 \cap \cdots \cap S_n$. Thus, there exists a model $\mathbf{I}$ of $\Gamma''$ such that $\mathbf{I}(p) \in S_1 \cap \cdots \cap S_n$. To obtain $\Gamma'$ from $\Gamma''$ we add some truth values to the signs of the atom $p$, but anyway $\mathbf{I}$ continues being a model of $\Gamma'$. To obtain $\Gamma$ from $\Gamma'$ we add new clauses to $\Gamma'$ and literals to clauses of $\Gamma'$. The addition of literals has no effect on the satisfiability and the new clauses contain a literal $S : p$ such that $S_1 \cap \cdots \cap S_n \subseteq S$. Therefore, $\mathbf{I}(p) \in S_1 \cap \cdots \cap S_n \subseteq S$ and $\mathbf{I}$ satisfies $\Gamma$. ■

The pseudo-code of a satisfiability checking procedure based on the notions of signed branching (cf. Proposition 1) and signed unit resolution (cf. Proposition 2) is displayed in Figure 2. Function signed-sat takes as input a signed cnf-formula $\Gamma$ with the property that a propositional atom occurs at most once in each clause. It returns true if $\Gamma$ is satisfiable and false if $\Gamma$ is unsatisfiable. Function signed-unit-resolve applies repeatedly signed unit resolution, function pick-atom selects the propositional atom to which the signed branching rule is applied and function maximal-set-truth-values$(\Gamma, p)$ computes a maximal set of truth values of the propositional atom $p$ wrt the signed cnf-formula $\Gamma$. Pick-atom can select a propositional atom randomly or using some branching heuristic (this question is discussed in [13]).

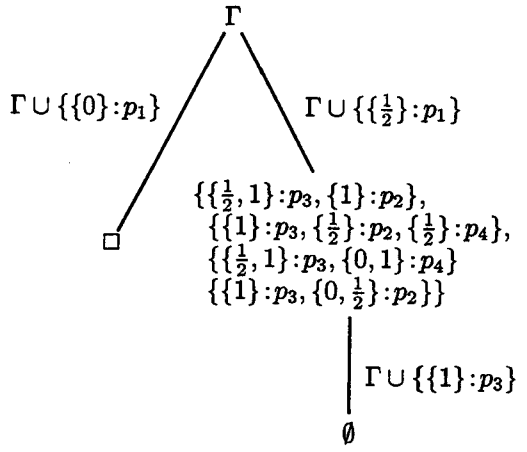



Figure 3: A signed-sat proof tree for the signed cnf-formula $\Gamma$ in Example 2

**Example 2** *Let* $\Gamma$ *be the following signed cnf-formula:*

$$\Gamma = \{\{\{\tfrac{1}{2},1\}:p_1,\{0,\tfrac{1}{2}\}:p_4\},\{\{0\}:p_1,\{\tfrac{1}{2},1\}:p_3,\{1\}:p_2\},$$
$$\{\{0\}:p_1,\{1\}:p_3,\{\tfrac{1}{2}\}:p_2,\{\tfrac{1}{2}\}:p_4\},\{\{\tfrac{1}{2},1\}:p_1,\{1\}:p_4\},$$
$$\{\{\tfrac{1}{2},1\}:p_3,\{0,1\}:p_4\},\{\{1\}:p_3,\{0,\tfrac{1}{2}\}:p_2\}\}$$

*Figure 3 shows a proof tree of the satisfiability of* $\Gamma$ *obtained using function signed-sat.*

Next, we shall prove that the proof method embodied in our algorithm is sound and complete, as well as that the procedure terminates.

**Theorem 1 soundness, completeness and termination** *Given a signed cnf-formula* $\Gamma$, *function signed-sat terminates and returns true or false. If it returns true, then* $\Gamma$ *is satisfiable. If it returns false, then* $\Gamma$ *is unsatisfiable.*

*Proof:* First, we show that if function signed-sat terminates either some branch of the proof tree created by signed-sat contains the empty formula (i.e. returns true) or every branch of the proof tree contains the empty clause (i.e. returns false). Suppose we have a

proof tree with no branch containing the empty formula and some leaf node of a branch with a signed cnf-formula $\Gamma'$ which does not contain the empty clause; we show we do not have termination. If $\Gamma'$ contains signed unit clauses, then we can apply signed-unit-resolve; otherwise, we select a porpositional atom $p$ occurring in $\Gamma'$ and we can do branching. Either way, we do not have termination.

Next we show that every proof attempt must terminate. Signed-unit-resolve decreases the number of occurrences of literals and after applying the signed branching rule we obtain new formulas with a signed unit clause $\{\{x\}:p\}$. We can apply signed-unit-resolve to these new formulas and at least all the occurrences of literals containing the atom $p$ are removed. Since we began with a finite number of occurrences of literals, these rules can only be applied a finite number of times.

As we have shown that signed unit resolution (see Proposition 2) and signed branching (see Proposition 1) preserve satisfiability, it is clear that when signed-sat returns false (i.e all the branches of the proof tree contain the empty clause), the input formula $\Gamma$ is unsatisfiable, and when returns true (i.e. there is a branch with the empty formula), $\Gamma$ is satisfiable. ■

## 6 Concluding remarks

We have presented a satisfiabibility checking procedure for signed cnf-formulas that can be understood as a generalization of the two-valued Davis-Putnam procedure. To this end, we have developed a multiple valued unit resolution and an optimized multiple valued branching rule based on the concept of maximal set of truth values. We have also proved the correctness of the procedure.

In order to obtain competitive satisfiability testing algorithms, we should consider to enhance the proof procedure designed with efficient data structures for representing signed cnf-formulas and clever heuristics for selecting the literal to which the branching rule is applied.

## References


[1] M. Baaz and C. G. Fermüller. Resolution-based theorem proving for many-valued logics. *Journal of Symbolic Computation*, 19:353–391, 1995.

[2] M. Buro and H. K. Büning. Report on a SAT competition. *EATCS Bulletin*, 1(49):143–151, 1993.

[3] S. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.

[4] M. Davis, G. Logemann, and D. Loveland. A machine program for theorem-proving. *Comm. of the ACM*, 5:394–397, 1962.

[5] M. Davis and H. Putnam. A computing procedure for quantification theory. *JACM*, 7(3):201–215, 1960.

[6] G. Escalada-Imaz and F. Manyà. The satisfiability problem for multiple-valued Horn formulæ. In *Proc. International Symposium on Multiple-Valued Logics, ISMVL'94, Boston/MA, USA*, pages 250–256. IEEE Press, Los Alamitos, 1994.

[7] R. Hähnle. *Automated Deduction in Multiple-Valued Logics*, volume 10 of *International Series of Monographs in Computer Science*. Oxford University Press, 1994.

[8] R. Hähnle. Short conjunctive normal forms in finitely-valued logics. *Journal of Logic and Computation*, 4(6):905–927, 1994.

[9] R. Hähnle. Exploiting data dependencies in many-valued logics. *Journal of Applied Non-Classical Logics*, 6:49–69, 1996.

[10] R. Hähnle. Tableaux for many-valued logics. In M. D'Agostino, D. Gabbay, R. Hähnle, and J. Possega, editors, *Handbook of Tableau Methods*. Kluwer, Dordrecht, 1996. To appear.

[11] F. Harche, J. N. Hooker, and G. Thompson. A computational study of satisfiability algorithms problems for propositional logic. *ORSA J. Computing*, 6:423–435, 1994.

[12] D. W. Loveland. *Automated Theorem Proving. A Logical Basis*, volume 6 of *Fundamental Studies in Computer Science*. North-Holland, 1978.

[13] F. Manyà. *Proof Procedures for Multiple-Valued Propositional Logics*. PhD thesis, Universitat Autònoma de Barcelona, 1996.

[14] N. V. Murray and E. Rosenthal. Signed formulas: A liftable meta logic for multiple-valued logics. In *Proceedings ISMIS'93, Trondheim, Norway*, pages 275–284. Springer LNCS 689, 1993.

# Índice de Autores


Alsinet, T. 159
Altarriba, N. 77
Aranda, M.C. 51
Aranguren, G. 77
Aymerich, F.X. 153

Balsa, M. 57
Barea, R. 195
Barriga, A. 257, 269, 335
Benítez, J.M. 183
Bernad, P.E. 263
Bertoluzza, C. 213
Blanco, A. 183
Bodini, A. 213
Boixader, D. 1
Bolaños, M.J. 221
Boquete, L. 195
Burillo, P. 275, 293

Cadenas, J.M. 165, 171, 305
Calvente, J. 245
Calvo, T. 33
Canet, P. 33
Cano, J.M. 189
Carbonell, M. 33
Carbonell, P. 233
Carneiro, V. 239
Castro, J.J. 127
Castro, J.L. 127
Colubi, A. 287
Cordón, O. 109
Couso, I. 121, 177
Cruz, J.M. de la 83
Cubillo, S. 27

Delgado, M. 15, 115, 183
Díez, M. 329
Dimitriadis, Y. 189
Domingo, J.S. 323
Dormido, S. 83

Echanove, J. 141
Elorza, J. 275
Escalada, G. 317
Escoda, I. 201
Espin, A. 127
Esteva, F. 21

Ferreiro, R. 239
Flores, A. 171
Fuentes, R. 293

Galindo, J. 51
García, J.L. 45
García, P. 21
García, R. 195
Garitagoitia, J.R. 141
Garmendia, L. 281
Garrido, M.C. 165
Gibert, K. 217
Gil, M.A. 287
Gil, P. 177, 299
Gili, J. 153
Giral, R. 245
Godo, L. 21, 227
Gomáriz, S. 251
Gómez, A.F. 115
Gómez, J. 115
González, L. 293
González de M., J.R. 141
Guerrero, J.A. 251
Guinjoan, F. 251

Herms, A. 201
Herrera, F. 109
Herrero, M.J. 7
Hinojosa, M.A. 269

Jacas, J. 1
Jesús, M.J. del 109
Jiménez, C.J. 257, 269
Jiménez, J. 329
Juan, N. 133

Lago, E. 269
Lasso, P. 89
Leyva, R. 245
Llamazares, B. 45
López, D. 257, 269, 335
López, H. 207
López, J. 189
López, M. 207
López de Arroyabe, J. 77
Lubiano, A. 287

Magdalena, L. 103
Manyà, F. 317
Marín, R. 57
Martín, F. 171
Martínez, H. 115
Martínez, L. 245
Mas, M. 39
Matellán, V. 147
Mayor, G. 39
Mazo, M. 195
Medina, J.M. 51
Molina, J.M. 147
Monasterio, F. 89, 103
Montes, S. 299, 329
Montseny, E. 153
Moreno, F.J. 257

Navarrete, I. 57
Nettleton, D. 217

Pardo, X.C. 239
Pradera, A. 27

Realp, E. 323
Recasens, J. 1
Reche, F. 221
Requena, I. 183
Rivero, C. 89, 103
Rodríguez, A. 7
Rodríguez, M. 77
Ruiz, A. 165

Sánchez, D. 15, 311
Sánchez, L. 121
Sánchez, S. 257, 269, 335
Santos, M. 83, 133
Sanz, A. 141, 201
Sobrevilla, P. 153
Sobrino, A. 71
Sommaruga, L. 147
Suñer, J. 39

Torra, V. 65, 227
Torrens, J. 39
Trillas, E. 7, 27

Uribe, J.P. 95
Urzelai, J. 95

Verdegay, J.L. 305
Vidal, E. 245
Vidal, J. 239
Vila, M.A. 15

Zurita, J.M. 127