73

UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLÁN

CONTROLADORES BORROSOS

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO MECÁNICO ELECTRICISTA

P R E S E N T A:

RICARDO PADILLA TORRES

ASESOR: DR. ZBIGNIEW OZIEWICZ KWASS

CUAUTITLÁN IZCALLI, EDO. DE MEX. 2001

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN
UNIDAD DE LA ADMINISTRACION ESCOLAR
DEPARTAMENTO DE EXAMENES PROFESIONALES

UNIVERSIDAD NACIONAL AVENMA DE MEXICO

ASUNTO: VOTOS APROBATORIOS

U. N. A. M.
FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN

DR. JUAN ANTONIO MONTARAZ CRESPO
DIRECTOR DE LA FES CUAUTITLAN
P R E S E N T E

DEPARTAMENTO DE EXAMENES PROFESIONALES

ATN: Q. Ma. del Carmen García Mijares
Jefe del Departamento de Exámenes
Profesionales de la FES Cuautitlán

Con base en el art. 28 del Reglamento General de Exámenes, nos permitimos comunicar a usted que revisamos la TESIS:

Controladores Borrosos

que presenta el pasante: Ricardo Padilla Torres
con número de cuenta. 8915948-6 para obtener el título de :
Ingeniero Mecánico Electricista

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VOTO APROBATORIO.

A T E N T A M E N T E
"POR MI RAZA HABLARA EL ESPIRITU"
Cuautitlán Izcalli, Méx. a 11 de Junio de 2001.

PRESIDENTE	Fís. José de Jesús Cruz Guzmán	
VOCAL	Ing. Nicolás Calva Tapia	
SECRETARIO	Dr. Zbigniew Oziewicz Kwesa	Zbigniew Oziewicz
PRIMER SUPLENTE	Ing. Joel Sánchez Pérez	
SEGUNDO SUPLENTE	Ing. Pedro Rendón Torres	

AGRADECIMIENTOS

La realización de este trabajo fue posible gracias a la guía, paciencia y consejo de el Dr. ZBIGNIEW OZIEWICZ KWASS.

A mis padres Ma. de Lourdes Torres Martínez y Roberto Padilla Vera, quienes me han apoyado durante mi vida.

Principalmente al Autor de la vida, Alfa y Omega, Principio y Fin, por siempre Honor, Gloria y Poder. YESHUA HA MASHIA

Esta tesis es parte de un temario más amplio de investigaciones que comprende los siguientes temas:

- En Cátedra: Álgebra y Lógica en la ciencia computacional en la Facultad de Estudios Superiores Cuautitlán para Maestría clave 2.04 nivel 3, impartido por el Dr. ZBIGNIEW OZIEWICZ KWASS.
- Proyecto CONACYT 27670E (1999-2000), Métodos gráficos en modelos algebraicos.
- Proyecto IN-109599(1999-2002), Lógicas no-clásicas y aplicaciones en ciencias de la computación, apoyado por DGAPA.

LOGROS

- Como tema de tesis se propone la elaboración de ejemplos conocidos y nuevos de sistemas borrosos.

- La recopilación de la información de la tesis fue muy amplia a pesar de el poco acceso a la información en español. La bibliografía en la tesis es muy reciente.

- Es un orgullo que el tema de la tesis sea contemplado como una materia en el curso de posgrado en mi facultad.

- Se pusieron en práctica mis conocimientos de el idioma inglés para traducir toda la bibliografía mencionada.

- El aprendizaje de los fundamentos teóricos de la lógica borrosa fue un gran reto.

- Esta tesis fue escrita en el software TEX y LATEX versión LaTex2e. El aprender y manejar adecuadamente este paquete fue un gran logro que requirió de paciencia y constancia.

- El tema de la tesis es de vanguardia mundial. La teoría de control borroso es muy joven y tiene mucho campo para ser explotada.

- Muchas de las gráficas presentadas en la tesis fueron elaboradas en MATLAB versión 5. El manejo de este software fue un reto adicional para mejorar la calidad y presentación de la tesis.

PERSONAL TEX INC

Contenido

Capítulo 1

INTRODUCCIÓN

Los problemas de control en los procesos industriales son dominados por comportamientos no-lineales y variantes en el tiempo, muchos ciclos internos y mucha interacción en los ciclos de control [1, 2]. Los controladores borrosos tienen en algunos casos la capacidad de simular las acciones de un operador humano. Para los niveles de control alto y supervisado se puede con algunos simples controladores combinados en prioridad jerárquica para controlar un sistema. Por proceso de control se entenderá como la automatización a larga escala de una planta industrial, tan compleja que sería imposible alcanzar una completa y satisfactoria particularidad de control especificada. Ejemplos típicos de control pueden ser el proceso de producción del vidrio, cemento, plástico y generación de energía eléctrica. Los principales objetivos de un controlador supervisado son: operación segura, altos niveles de producción y una operación poco costosa. Estos tres objetivos son usualmente imposibles de alcanzar simultáneamente, por lo que se tienen que dar en orden de prioridad.

Lógica borrosa es un concepto concebido por Lotfi Zadeh [3, 4, 5] empleando como una herramienta para resolver problemas de control, utilizando una metodología que por sí misma es simple, pequeña que puede ser aplicada desde los más sencillos dispositivos de control hasta las grandes estaciones de bases de datos para procesar información de manera más eficiente. Puede ser implementada en hardware, software y una combinación de ambos, además de muchos campos de la ingeniería. La lógica borrosa provee de un método sencillo para aterrizar en conclusiones definidas basadas en información ambigua, difusa, imprecisa, ruidosa o hasta incompleta.

La lógica borrosa incorpora una simple base de reglas fundamentadas en sentencias. Si X y Y entonces Z para resolver problemas, en vez de basarse únicamente en modelos matemáticos de sistemas. El modelo de lógica borrosa se basa en información empírica proveniente de la experiencia de un operador más que en el entendimiento técnico de un sistema.

La lógica borrosa fue concebida como un mejor método para obtener información, pero ha probado ser una excelente opción para muchas aplicaciones de control. Los controladores borrosos están conformados principalmente por:

- una base de reglas (si....entonces), que contienen una lógica borrosa cuantificable de los conocimientos de el experto en términos de variables lingüísticas que describen el comportamiento de el sistema.

- Un mecanismo de inferencia (módulo de inferencia borroso), que emula las decisiones de los expertos en la toma de decisiones, interpretando y aplicando algún conocimiento para controlar de la mejor manera posible el sistema.

- Un mecanismo borrosificador, el cual convierte las entradas de control en información que el mecanismo de inferencia puede utilizar para activar y aplicar reglas.

- El mecanismo desborrosificador, que convierte las conclusiones de el mecanismo de inferencia en salidas para el control de el sistema.

En la fig. 1 se presenta la arquitectura básica de un controlador borroso; y módulos que lo constituyen dentro del lazo de control.

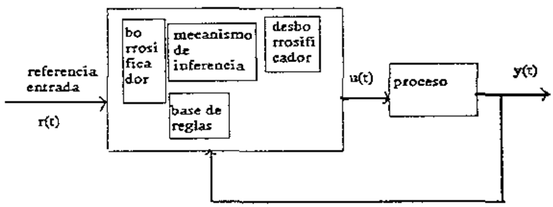


fig. (1) Arquitectura básica de un controlador borroso.

La presente tesis tratará de presentar la teoría de control borroso como una buena alternativa de control para sistemas lineales y no lineales; la cual difiere de los métodos convencionales de control, que se basan principalmente en el modelaje matemático de los sistemas físicos como herramienta para el diseño de modelos de control.

La teoría de control borroso es nueva, pues fue primeramente propuesta por Lotfi A. Zadeh en la Universidad de California en 1965 con su teoría "lógica borrosa", la cual desarrolló posteriormente en 1973 introduciendo el concepto de variables lingüísticas. La teoría de Lotfi A. Zadeh ha sido utilizada para implementar sistemas de control borroso que sean más flexibles al admitir variables lingüísticas (bajo, medio, alto) como variables de entrada a un sistema, a diferencia de valores exactos que son requeridos como variables de entrada en los sistemas convencionales de control.

Debido a la facilidad de la lógica borrosa para admitir variables lingüísticas como entrada a el sistema, es posible guardar y procesar información que sea ambigua e imprecisa [7].

En ésta tesis se presentarán los fundamentos de la teoría borrosa y la teoría de control borrosa como herramientas para implementar sistemas de control.

Bibliografía

[1] Charles E. Rohrs, James L. Melsa, Donald G. Schultz (1994), *Sistemas de Control Lineal*, McGraw Hill, ISBN 970-10-0411-6, páginas 13, 14, 77.

[2] J. J. D'Azzo, C. H. Houpis (1966), *Feedback Control System, Analysis and Synthesis*, McGraw Hill Kagakusha, *LCCCN 65-17391*.

[3] Zadeh Lotfi A. (1965), Fuzzy sets, *Inf. and control* 8, páginas 338-353.

[4] Zadeh Lotfi A. (1973), Outline of a new approach to the analysis of complex systems and decision proceses, *IEEE Trans. Systems, Man & Cybernetics* **1**, páginas 28-44.

[5] Zadeh Lotfi A. (1975), The concept of a lingüístic variable and its application to aproximate reasoning, *Information sciences* 8, páginas 43-80.

[6] Peirce. C. S. (1931), *Collected papers of Charles Sanders Peirce*, C. Hartshorne and P. Weiss, editors, Harvard University Press, páginas 9-11.

[7] Bezdek J. (1993), *Fuzzy Models - What Are They and Why?*, IEEE Transaction on Fuzzy systems, vol. 1, páginas 3-7.

Capítulo 2

ANTECEDENTES HISTÓRICOS

El concepto de Lógica Borrosa fue concebido por Lotfi A. Zadeh, un profesor en la Universidad de California en Berkley, presentando no como una metodología de control sino como una forma para procesar información por medio de la implementación de relaciones de pertenencia entre conjuntos que no sólo se limitarán a los antiguos conceptos de pertenencia o no pertenencia de elementos en determinados conjuntos. Ahora se les puede relacionar con otro tipo de función de pertenencia en la que se lograra tener distintos grados de pertenencia entre ellos.

La lógica clásica es bivaluada y se inventó por Boul en 1854, la lógica trivaluada por el polaco Łukasiewicz en 1918 y la lógica borrosa es ∞-valuada, [0,1]-valuada, valuada en el intervalo $[0,1] \in R$.

Antes de trabajar en la teoría borrosa, Zadeh fue muy bien conocido por su teoría en control [4]. En los 60's, él pensó que la teoría de control moderna había puesto mucho énfasis en la precisión y por lo tanto no podía de manera adecuada tomar sistemas complejos. En 1962, él escribió que para comprender sistemas biológicos, se necesitaba un diferente y radical concepto de matemáticas, las matemáticas de cantidades borrosas que no son describibles en términos de distribución probabilística. Posteriormente, él formalizó la idea en su artículo llamado "conjuntos borrosos".

Los alcances de su teoría de conjuntos, no fue aplicada a sistemas de control, sino hasta los años 70's, debido a la insuficiente capacidad de las computadoras de aquellos días.

Los ingenieros japoneses, con su sensibilidad a nuevas tecnologías, rápidamente encontraron que los controladores borrosos eran muy fácil de diseñar, y que funcionaban muy bien para resolver muchos problemas [7].

El Profesor Kaoru Hirota (Professor, Interdisciplinary Graduate School of

Science and Engineering, Tokyo Institute of Technology) ha trabajado desarrollando aplicaciones industriales en la tecnología borrosa y ha publicado muchos de sus trabajos .[8, 9, 10]

Debido a que el control borroso no requiere de modelos matemáticos de los procesos, éste puede ser aplicado a muchos sistemas donde la teoría de control convencional no podía ser usada, debido a la falta de modelos matemáticos precisos.

El Profesor Zadeh reconoció que la gente no requiere de información precisa y numérica, y aún así son capaces de realizar tareas de control adaptativo alto y formular modelos decisorios precisos [2]. Si los controladores de lazo cerrado pudieran ser programados para aceptar información imprecisa y con ruido, entonces serían mucho más efectivos, y tal vez más fácil de implementar [1, 11].

A pesar de que la manufactura estadounidense no ha aceptado esta tecnología, los europeos y japoneses sí la han aprovechado, creando productos que se utilizan en la vida diaria, como es el caso de los sistemas de transporte en trenes (highways) [4, 5, 6].

Bibliografía

[1] Zadeh Lotfi A. (1984), Making computers think like people, *IEEE Spectrum*, páginas 26-32.

[2] Bellman R. E & Zadeh Lotfi A. (1970), Decision Making in a Fuzzy Enviroment, *Management Science*, **7**, páginas 141-164.

[3] Zadeh Lotfi A. (1965), Fuzzy sets, *Inf. and control* 8, páginas 338-353.

[4] Bosacchi B. and Masaki Ichiro (1993), *Fuzzy Logic Technology and the Inteligent Highway system*, IEEE International Conference on Fuzzy Systems, vol. **1**, páginas 65-70.

[5] Favilla J., Machion A. and Gomide F. (1993), *Fuzzy Traffic Control: Adaptive Strategues*, IEEE International Conference on Fuzzy Systems, vol. **1**, páginas 506-511

[6] Iokibe T., Mochizuki N. and Kimura T. (1993), *Traffic Prediction Method by Fuzzy Logic, IEEE International Conference on Fuzzy Systems*, vol. **2**, páginas 673-678.

[7] King P. J. and Mamdani E. H. (1975), *The Application of Fuzzy Control Systems to Industrial Processes*, Automatica, vol. **3**, páginas 235-242.

[8] Hirota Kaoru (1996), *Journal of Chinese Fuzzy Systems Association*, vol. **2**, páginas 1-13.

[9] Hirota Kaoru y Pedrycz Witold (1994), *Fuzzy Sets and Systems*, Elsevier, vol. **68**, páginas 157-170.

[10] Hirota K., Tsurumaru T., Motegi A., Yubazaki N., Ohtani M., Mayajima T. (1995), *Neurocomputing*, Elsevier, vol. **9**, páginas 27-38.

[11] Jamshidi M., Titli A., Zadeh L. y Boverie S. (1997), *Applications of Fuzzy Logic*, Prentice Hall, ISBN 0-13-362831-0.

Capítulo 3

CONJUNTOS BORROSOS

Los conjuntos borrosos son un desarrollo más amplio del concepto matemático de conjuntos [1]. Los conjuntos fueron estudiados formalmente por el matemático alemán George Cantor (1845-1918). Su teoría de conjuntos tuvo mucha resistencia durante su vida, sin embargo, hoy en día bastantes matemáticos creen que es posible expresar casi, si no es que todos, los conceptos matemáticos, en el lenguaje de teoría de conjuntos. Muchas investigaciones se han realizado con el fin de buscar las consecuencias de la teoría de conjuntos borrosos. Para los ingenieros en control, lógica borrosa y relaciones borrosas son de mucha importancia, ya que es donde se tiene mayor aceptación y mayor número de aplicaciones en operación.

3.1 Conjunto convencional

Un conjunto puede ser descrito como una colección de objetos, los cuales pueden ser tratados como un todo [2]. Cantor describió a los conjuntos por sus miembros, tal que un elemento de dicho universo sea o no, un miembro de este universo. Los términos conjunto, colección y clase, son sinónimos, al igual que los términos pieza, elemento y miembro. Casi cualquier cosa llamada conjunto en conversación ordinaria, puede ser aceptable como un conjunto matemático. La siguiente lista es una bien definida colección de objetos, y por tanto se le puede dar el nombre de conjuntos:

- El conjunto de los enteros no negativos menores de 4. Este es un conjunto finito con cuatro miembros: 0, 1, 2 y 3.

- El conjunto de dinosaurios vivos en algún museo. Este conjunto no tiene miembros y se le llama un conjunto vacío.

- El conjunto de medidas mayores que 10 volts. A pesar de que éste conjunto es infinito, es posible determinar si una medida dada es o no un miembro del conjunto.

Un conjunto puede ser especificado por medio de sus miembros, los que caracterizan a un conjunto completamente. La lista de miembros A = 0, 1, 2, 3 especifican un conjunto finito. Nadie puede enumerar todos los elementos de un conjunto infinito, por lo que se tienen que establecer ciertas propiedades que caracterizarán a dichos elementos en el conjunto.

3.1.1 Ejemplo

Sea el predicado $x < 10$.

Éste conjunto es definido por los elementos de el universo en cuestión, el cual hace al predicado verdadero. Por lo tanto hay dos maneras de describir a un conjunto: explícitamente en una lista o implícitamente con un predicado.

3.2 Función Característica

Según Zadeh, muchos conjuntos tienen criterios que van más allá del simple " o éste o éste otro", en lo que se refiere a su relación de pertenencia [6].

El propósito de un conjunto convencional en matemáticas, es caracterizar formalmente algún concepto. Por ejemplo, el concepto de "números enteros que son mayores o iguales a 3 y menores o iguales a 10" pueden ser representados por el siguiente conjunto: $\{ x \in I : 3 \leq x \leq 10 \} = \{3,4,5,6,7,8,9,10\}$

donde I es el conjunto de enteros, nuestro universo o referencia que contiene a todos los elementos relevantes para dicho concepto en particular.

Un conjunto convencional, sea A, debe ser igualado con su función característica [1]

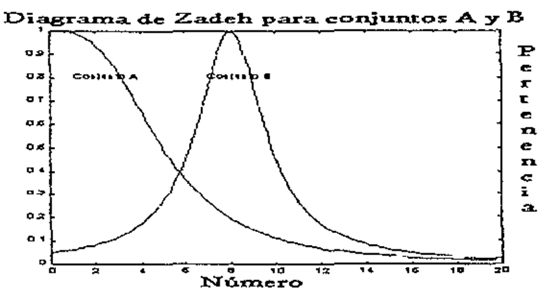
$$\mu_A(x) : X \to \{0, 1\}$$

la cual asocia a cada elemento x de el universo X, un número μ (x)$\in$\{0,1\} tal que $\mu_A(x) = 0$ significa que tal x $\in$ X no pertenece a el conjunto A, y $\mu_A(x)$=1 significa que tal x pertenece a el conjunto A. Así, el conjunto A=\{3,4,5,6,7,8,9,10\} puede ser representado por su función característica

$$\mu_A(x) = \begin{cases} 1, & \text{si } x \in A; \\ 0, & \text{para } x \notin A \end{cases}$$

Así, en un conjunto convencional hay un claro corte en la diferenciación entre elementos pertenecientes al conjunto o no. La transición de pertenencia a no pertenencia es abrupta. Sin embargo, cuando tratamos de formalizar por medio de un conjunto vago conceptos como "números enteros que son más o menos igual a 6", notamos que una abrupta y clara diferenciación entre los elementos pertenecientes y no pertenecientes es artificial.

Esto guió a Zadeh (1965), a la idea de conjunto borroso, la cual es una clase de objetos con límites no definidos claramente, en los cuales la transición de la pertenencia a la no pertenencia no es abrupta; así, elementos de un conjunto borroso pueden pertenecer a éste en grados parciales, de la pertenencia absoluta a la no pertenencia absoluta, a través de todos los valores inmediatos [4]. El siguiente diagrama muestra dos funciones de petenencia A (números pequeños) y B (cercanos a 8) [2].



La función característica es reemplazada por una función de pertenencia definida como

$\mu_A(x)$:U $\rightarrow$[0,1]. A continuación se presentan los distintos casos de valuación lógica:

- μ : U $\rightarrow$ $\{0, 1\}$ Lógica bi-valuada.

- μ : U $\rightarrow$ $\{0, \frac{1}{2}, 1\}$ Lógica tri-valuada.

- μ : U $\rightarrow$ $\{0, \frac{1}{3}, \frac{2}{3}, 1\}$ Lógica cuatri-valuada.

- μ : U $\rightarrow$ $[0, 1]$ Lógica ∞-valuada, no práctica.

Un conjunto borroso en un universo dado U es caracterizado por medio de una función de pertenencia $\mu_A(x)$ que toma valores en el intervalo [0, 1] [3, 7].

Por lo tanto, un conjunto borroso es una generalización de el conjunto convencional ya que permite que la función de pertenencia tome valores en el intervalo [0,1]. En otras palabras, la función de pertenencia de un conjunto convencional o clásico, sólo puede tomar dos valores, a decir cero y uno, mientras que la función de pertenencia de un conjunto borroso nos permite tomar valores comprendidos en el rango de cero a uno, en los que cada valor podrá obtener distintos niveles de proximidad a un valor bajo, medio o alto [5]. Estos valores pueden a su vez subdividirse en medio, bajo, muy bajo y así progresivamente, dependiendo de la precisión que se desee. Un conjunto borroso A en U, puede ser representado como un conjunto de pares ordenados de un elemento genérico x, y su valor de pertenencia, esto es:

$$A = \{(x, \mu_A(x)) \mid x \in U\}.$$

Donde U es continuo (por ejemplo U $=\mathbb{R}$), A es comúnmente descrito como

$$A = \int_U \mu_A(x)/x.$$

Donde la integral no denota integración, ésta denota la colección de todos los puntos $x \in U$ con la función de pertenencia μ_A(x). Cuando U es discreta, A es comúnmente descrito, como:

$$A = \Sigma_U \mu_A(x)/x.$$

Donde el símbolo de sumatoria no representa adición aritmética; éste denota la colección de todos los puntos $x \in U$ con su función de pertenencia μ_A(x). Para ejemplificar, consideremos el conjunto "gente joven". Un niño de un año de edad puede ser considerado como un miembro de el conjunto, y una persona de 100 años de edad no será considerada miembro de éste conjunto, pero qué podremos decir acerca de las personas de 20 o 40 años de edad?. Otro ejemplo es el reporte climatológico, acerca de temperaturas elevadas, vientos fuertes, o un día bonito.

En otros casos el criterio parece ser no-borroso, pero es percibido como borroso: un límite de velocidad de 60 kilómetros por hora, un hombre de 50 años de edad.

Zadeh propuso "un grado de pertenencia", tal que la transición de pertenencia a no pertenencia es gradual en vez de abrupta.

Un grado de pertenencia de un elemento es normalmente un número entre 0 y 1, algunas veces denotado por la letra griega μ . A mayor el número, mayor el grado de pertenencia. Zadeh reconoce los conjuntos de Cantor como un caso especial donde los elementos tienen pertenencia completa μ=1. Zadeh, sin embargo, no llama a los conjuntos de Cantor como conjuntos no-borrosos; hoy en día el término conjunto crisp es usado para evitar dilemas.

Nótese que Zadeh no dá una base formal para determinar el grado de pertenencia. La pertenencia para una persona de 50 años de edad en un conjunto de jóvenes, depende de el punto de vista de cada individuo. El grado de pertenencia es una precisa, pero subjetiva medida que depende de el contexto. Una función de pertenencia borrosa es diferente de una distribución probabilística.

3.3 Función de pertenencia

Cada elemento en el universo en cuestión, es un miembro de el conjunto borroso en algún grado, aún cero. El grado de pertenencia para todos sus miembros describe a un conjunto borroso, tal como negación. A los elementos del conjunto borroso se les asigna un *grado de pertenencia*, tal que la transición de pertenencia a no pertenencia es gradual y no abrupta. La función que relaciona a un número con cada elemento x de el universo es llamada función de pertenencia o membresía μ_A(x).

El conjunto de elementos que tienen una función de pertenencia no-cero se le llama soporte de el conjunto borroso.

3.4 Representación discreta y continua

Existen dos formas alternativas para representar a una función de pertenencia en una computadora: *forma discreta y forma continua*. En la forma continua, la función de pertenencia es una función matemática o posiblemente un programa. Una función de relación tiene como ejemplo la forma de campana, (también llamada *curva* π), forma s (*curva-s*), inversa de s (*curva-z*), triangular o trapezoidal.

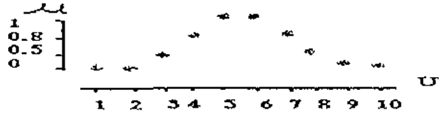
En la forma discreta, la función de pertenencia y el universo son ambos puntos discretos en una lista (*vector*). Algunas veces puede ser más conveniente utilizar una representación discreta.

Consideraremos a continuación dos ejemplos de representación continua y discreta que son un clásico en los artículos publicados por Zadeh [1965].

Sea U los números enteros de 1 a 10, esto es, U = $\{1, 2, ..., 10\}$. Entonces el conjunto borroso varios puede ser definido como

$$varios = 0.5/3 + 0.8/4 + 1/5 + 1/6 + 0.8/7 + 0.5/8$$

Ésto es, 5 y 6 pertenecen al conjunto borroso varios con grado 1, 4 y 7 con grado 0.8; 3 y 8 con grado 0.5, y 1, 2, 9 y 10 con grado 0.



fig(3.4) Representación de un conjunto borroso discreto

Sea U un intervalo [0,100] representando la edad de humanos ordinarios. Entonces podemos definir con conjuntos borrosos "joven" y "viejo" usando notación integral.

$$joven = \int_{o}^{25} 1/x + \int_{25}^{100} (1 + (\frac{x-25}{5})^2)^{-1}/x$$
$$viejo = \int_{50}^{100} (1 + (\frac{x-50}{5})^2)^{-1}/x$$

Los controladores borrosos usan una variedad de funciones de pertenencia. Un ejemplo común de una función que produce una curva en forma de campana, está basada en la función exponencial,

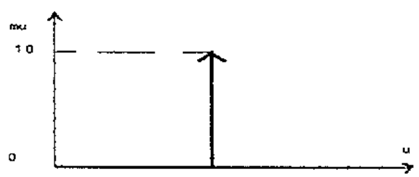
$$\mu(x) = exp[\frac{-(x-xo)^2}{2\sigma^2}].$$

Ésta es una curva Gausiana estándar con un valor máximo de 1, x es una variable independiente en el universo, xo es la posición de el pico relativo a el universo, y σ es la desviación estándar. Otra definición que no usa la exponencial es la correspondiente :

$$\mu(x) = [1 + (\frac{x-xo}{\sigma})^2]^{-1}$$

Los tipos de funciones más comunes son trapezoidal, singleton, triangular, tipo S, exponencial y tipo π, que a continuacin se describen.

La función de tipo singleton tiene valor 1, sólo para un punto a, y cero para el resto. Se utiliza habitualmente en sistemas borrosos simples para definir los conjuntos borrosos de las particiones de las variables de salida, pues permite simplificar los cálculos y requiere menos memoria para almacenar la base de reglas.

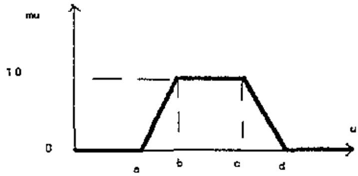


fig(3.1) función tipo sigleton

Se define como;

$$Sg(u;a) = \begin{cases} 1, & \text{si } u = a; \\ 0, & \text{si } u \neq a \end{cases} \tag{3.1}$$

La función tipo trapezoidal puede representarse como:



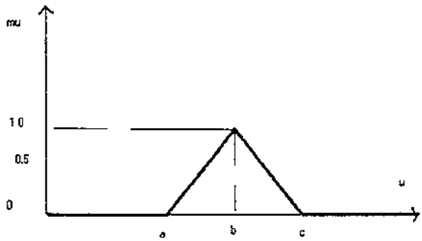
fig(3.2) función trapezoidal

La función de tipo trapezoidal se define por cuatro puntos a, b, c, d. Esta función es cero para valores menores de a y mayores de d, vale uno entre b y c, y toma valores en [1,0], entre a y b, y entre c y d. Se utiliza habitualmente en sistemas borrosos sencillos, pues permite definir un conjunto borroso con pocos datos y calcular su valor de pertenencia con pocos cálculos. Se emplean especialmente en sistemas basados en microprocesadores, pues con similar formato pueden codificarse también funciones de tipo S, función de tipo π, triangular y singleton, según se distribuyan los puntos a, b, c y d en la figura (3.2). Se define como;

$$TP(u;a,b,c,d) = \begin{cases} 0, & \text{si } u < a; \\ \left(\frac{u-a}{b-a}\right), & \text{si } a \leq u \leq b; \\ 1, & \text{si } b \leq u \leq c; \\ \left(\frac{d-u}{d-c}\right), & \text{si } c \leq u \leq d; \\ 0, & \text{si } u > d \end{cases} \tag{3.2}$$

Esta función resulta adecuada para modelar propiedades que comprenden un rango de valores (adulto, normal, adecuada, etc.). Para modelar una función triangular se hace b = c, para una función de tipo S (pero no suave) se hace c = d = max (U), y para una función de tipo singleton a = b = c = d. Una función coseno puede ser usada para generar una variedad de funciones de relación.

La función tipo T (triangular) puede definirse como:

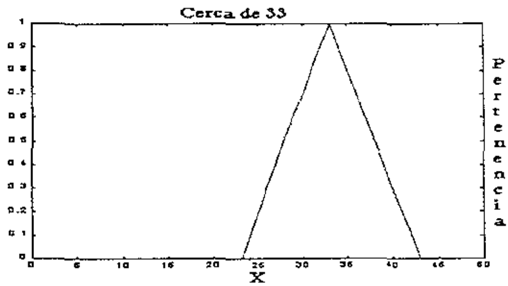


fig(3.3) función triangular

Esta función es adecuada para modelar propiedades con un valor de inclusión distinto de cero para un rango de valores estrecho en torno a un punto b.

$$T(u;a,b,c) = \begin{cases} 0, & \text{si } u < a; \\ (\frac{u-a}{b-a}), & \text{si } a \leq u \leq b; \\ (\frac{c-a}{c-b}), & \text{si } b \leq u \leq c; \\ 0, & \text{si } u > c \end{cases} \tag{3.3}$$

Por ejemplo una función de pertenencia para (x está cerca de 33) definida sobre x = [0:1:50] donde[a b c] = [23 33 43] se representa graficamente como:



La función de tipo S puede definirse como:

$$S(x:a,b)=\begin{cases}0, & \text{si } x<a;\\ 2(\frac{x-a}{b-a})^2, & \text{si } a\leq x\leq \frac{a+b}{2};\\ 1-2(\frac{x-b}{b-a})^2, & \text{si } \frac{a+b}{2}\leq x<b;\\ 1, & \text{si } x>b\end{cases} \qquad (3.4)$$

Esta función resulta adecuada para modelar propiedades como grande, mucho, positivo, etc. Se caracteriza por tener un valor de inclusión distinto de cero para un rango de valores por encima de cierto punto a, siendo 0 por debajo de a, y 1 para valores mayores de b. Su punto de cruce (valor 0.5) es (a+b)/2; y entre los puntos a y b es de tipo cuadrático (suave).

También se han utilizado funciones exponenciales para definir funciones de tipo S, como,

$$S(u;k,b) = \frac{1}{1+exp(-k(u-b))} \tag{3.5}$$

$$os(x_l, x_r, x) = \begin{cases} 0, & \text{para } x < x_l; \\ 1/2 + 1/2cos(\frac{x-x_r}{x_r-x_l}), & \text{para } x_l \leq x \leq x_r; \\ 1, & \text{para } x > x_r \end{cases} \tag{3.6}$$

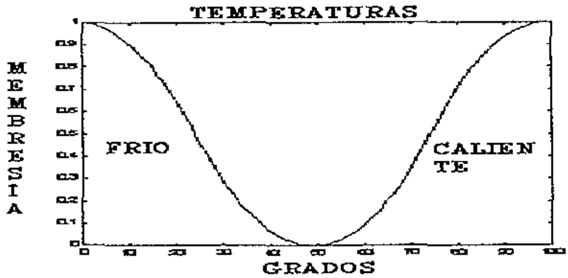
donde x_l es el punto de rompimiento a la izquierda, y x_r es el punto de rompimiento a la derecha. A continuación se presenta un ejemplo elaborado el el programa Matlab que nos representa la funsión de pertenencia tipo S.

```
EDU x=[0:100];
EDU frio=[50 25 0];
EDU caliente=[50 75 100];
EDU mf_frio=s_shape(x,frio);
EDU mf_caliente=s_shape(x,caliente);
EDU plot(x,[mf_frio;mf_caliente]);
EDU title('TEMPERATURA: FRIO Y CALIENTE');
EDU ylabel('MEMBRESA');
```

EDU xlabel('GRADOS');

EDU text(5,.45,'FRIO');

EDU text(77,.45,'CALIENTE');

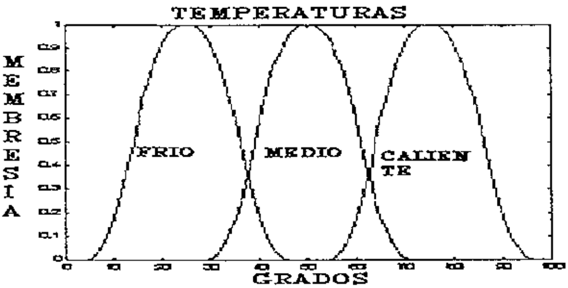


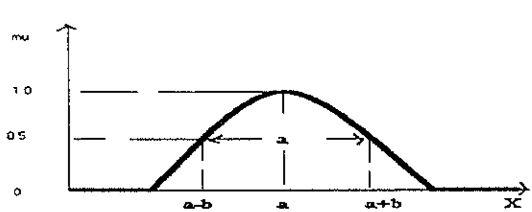
Por último, la función de tipo π puede definirse como sigue;

$$\Pi_1(x : a, b) = \frac{1}{1 + (\frac{x-a}{b})^2} \tag{3.7}$$

Esta función tiene forma de campana, y resulta adecuada para los conjuntos definidos en torno a un valor a, como medio, normal, cero, etc. Pueden definirse también utilizando expresiones analíticas exponenciales o cuadráticas, como la bien conocida campana de Gauss.

La siguiente figura nos presenta un ejemplo de la función de pertenencia tipo π





fig(3.5)función tipo Π

Bibliografía

[1] Witold Pedrycz (1998), *Computational Intelligence (An introduction)*, CRC Press, ISBN 0-8493-2643-5.

[2] Whitehead A. N. and Russell B. (1910), *Principia Mathematica*, vol. I, University of Cambridge Press.

[3] Arnold Kaufmann and Mandan M. Gupta (1988), *Fuzzy Mathematical Models in Engineering and Management science*, North-Holland, ISBN 04444705015, páginas 9-17.

[4] Zadeh Lotfi A. (1965), Fuzzy sets, *Inf. and control* vol. 8, páginas 338-353.

[5] Bart Kosko (1997), *Fuzzy engineering*, Prentice Hall, ISBN 0-13-1249916, páginas 8-12.

[6] Negoita C. V. and D. A. Ralescu (1975), *Applications of Fuzzy Sets to Systems Analysis*, Birkhauser, Besel and Stuttgart, and Halsted Press, New York.

[7] N. Rescher (1969), *Many-Valued Logic*, McGraw Hill, New York.

[8] J. Wesley Hines (1997), *Matlab Supplement to Fuzzy and Neural Approaches in Engineering*, A Wiley-Interscience Publication, New York 10158-0012, ISBN 0-471-19247-3, páginas 1-8.

Capítulo 4

VARIABLES LINGÜÍSTICAS.

Para especificar la base de reglas, el experto usará una "descripción lingüística" [5]; donde, expresiones lingüísticas son necesarias para especificar las entradas y salidas del sistema. Se usarán variables lingüísticas para describir sistemas borrosos.

Básicamente el razonamiento de Zadeh; en lo que se refiere a los alcances lingüísticos relacionados con complejos sistemas de control y procesos de control; es que los elementos básicos son variables lingüísticas ejemplificadas por "temperatura" por mencionar un ejemplo, la cual, toma valores lingüísticos como "alto", "bajo", "medio", etc., los cuales son igualados semánticamente con conjuntos borrosos.

Nótese que tales valores lingüísticos son comunes en la conversación cotidiana de el ser humano [1, 2, 3].

Al igual que una variable algebraica toma números como valores, una variable lingüística toma palabras o sentencias como valores (Zadeh en Zimmermann, 1991). El conjunto de valores que éste puede tomar es llamado su *conjunto término*. Cada valor en el conjunto término es una variable borrosa definida sobre una *variable base*. La variable base define al universo para todas las variables borrosas en el conjunto término. En resumen. La jerarquía es:

variable lingüística→variable borrosa→ variable base.

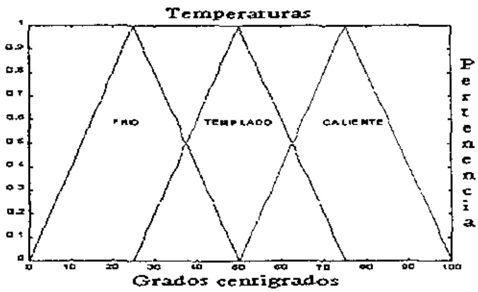
Sea x una variable lingüística con la etiqueta "edad". Los términos de ésta variable lingüística, los cuales son conjuntos borrosos, pueden ser "viejo", "joven", "muy joven", "muy viejo" de el término conjunto.

T=(*Viejo, Muy viejo, No muy viejo, más o menos viejo, algo joven, muy joven*)

Cada término es una variable borrosa definida en la variable base, la cual debe ser escalada de 0 a 100 años.

4.0.1 Ejemplo

Una variable borrosa llamada temperatura puede tener tres valores borrosos: frío, templado, caliente. Las funciones de pertenencia pueden ser construidas para definir a estos valores dentro del universo de discurso que son los grados centigrados.



Para representar la relación entre variables lingüísticas, se emplean sentencias condicionales borrosas. Por ejemplo, si tenemos dos variables lingüísticas, una primaria L y otra secundaria K, tal que el valor de L es un conjunto borroso A en X, y el valor de K es un conjunto borroso B en Y, entonces una relación entre L y K, en términos de sus valores A y B, respectivamente.

Pueden ser escritos como:

SI L = A ENTONCES K = B o dicho de otra manera SI A ENTONCES B.

4.1 Reglas Si.... entonces...

Las reglas borrosas combinan uno o más conjuntos borrosos de entrada, llamados antecedentes o premisas, y les asocian un conjunto borroso de salida llamado consecuente o consecuencia. Los conjuntos borrosos de la premisa se asocian mediante conjuntivas lógicas como y, o, etc. Una regla típica, de tipo Si... entonces, para un sistema de control sería "Si error es positivo-pequeño y derivada-de-error es negativo-pequeño, entonces acción es positiva-pequeña", que se suele expresar abreviadamente mediante expresiones de tipo:

Si E es PP y dE es NP entonces U es PP.

Las reglas borrosas permiten expresar el conocimiento de que se dispone acerca de la relación entre antecedentes y consecuentes. Para expresar este conocimiento de forma completa, normalmente se precisan varias reglas, que se agrupan para formar lo que se conoce como **base de reglas**, o conjunto de reglas que expresan las relaciones conocidas entre antecedente y consecuente.

La base de reglas se puede representar como una tabla de las reglas que la forman, o como una **memoria asociativa borrosa o FAM**(Fuzzy Associative Memory). Las FAM son matrices que representan la consecuencia de cada regla definida para cada combinación de dos entradas.

Las FAM permiten realizar una representación gráfica clara de las relaciones entre dos variables lingüísticas de entrada y la variable lingüística de salida, pero requiere que se indiquen explícitamente todas las reglas que se pueden formar con estas dos variables de entrada. Cuando el número de conjuntos de cada una de las particiones de entrada crece, las FAM se hacen difícilmente manejables. Es posible también definir FAM de más de dos dimensiones, pero su tamaño se hace rápidamente excesivo y son más difíciles aún de manejar. En su lugar, se suele trabajar con varias FAM de dimensión dos, para así definir subconjuntos de reglas que asocien las entradas de dos en dos en la base de reglas general.

Formalmente hablando, una **base de reglas borrosas** es una colección de reglas R^l con el formato:

$$R^l : \text{Si} \quad x_1 \quad es \quad F_1^l \quad \text{y.......} y\, x_n \quad es \quad F_n^l \quad \text{entonces} \quad y \quad es \quad G^l \tag{4.1}$$

Donde F_i^l y G^l son conjuntos borrosos en $U_i \subset \Re$ y V $\subset \Re$, respectivamente, y **x**=$(x_l, ...x_n)^T \in U_l x....x U_n$ e $y \in V$ son variables lingüísticas. Este formato de reglas se conoce como **borroso puro o de tipo Mamdani**, por ser quien primero las propuso en 1974 para realizar un controlador borroso que estabiliza un sistema, en torno a su punto de trabajo. Otro formato frecuente para las reglas es el llamado de tipo Sugeno. En este caso, la función de salida es una combinación lineal de las variables de entrada, o en un caso más general, una función genérica de las variables de entrada [4].

$$R^l : \text{Si} \quad x_1 \quad \text{es} \quad F_1^l \quad \text{y}.......y\, x_n \quad \text{es} \quad F_n^l \quad \text{entonces} \quad y_l = f_l(x) \qquad (4.2)$$

Si llamamos M al número de reglas Si...entonces de la base de reglas, entonces l = 1, 2,...,M en las ecuaciones anteriores. El vector **x** representa el conjunto de las entradas, mientras que "y" es la salida del sistema borroso. Los sistemas borrosos descritos con n entradas x_i y una sola salida y, se conocen como MISO (Multiple Input Single Output), mientras que los que tienen varias salidas (de 1 hasta k) se conocen como MIMO (Multiple Input Multiple Output). Para estos últimos sistemas, se puede generalizar el formato anterior de las reglas, o bien descomponiéndolo en k sistemas de tipo MISO.

Las reglas de tipo Mamdani permiten expresar el conocimiento previo disponible sobre el sistema, expresando así, el adquirido durante el proceso de optimización de reglas. Por su parte, las reglas de tipo Sugeno simplifican los cálculos de la salida, pero en general no resultan tan adecuadas para expresar el conocimiento de los expertos.

4.2 Término primario

Un término primario es un término o un conjunto que debe ser definido a priori, siendo éste, Joven o Viejo, donde los conjuntos "Muy Joven y No Joven" son conjuntos modificados.

Bibliografía

[1] Kosko Bart and Isaka S. (July 1993) *"Fuzzy Logic", Scientific American*, vol. **269**, No. 1, páginas 76 - 81.

[2] Kosko Bart (1991) *Neural Networks and Fuzzy Systems*, Prentice Hall.

[3] Resher N. (1969) *Many - Valued Logic*, McGraw Hill.

[4] Mamdani E. H. (December 1977), *"Application of Fuzzy Logic to Approximate* Reasoning Using lingüístic Synthesis." IEEE Transactions on Computers, vol. C - 26, No. 12, páginas 1182 - 1191.

[5] Zadeh Lotfi A. (1975) *The concept of a lingüístic variable and its application* to approximate reasoning **I, II, III**, Inf. Sci., vol. **8**, páginas 199-257, 301-357; vol. **9**, páginas 43-80.

Capítulo 5

CONCEPTOS BÁSICOS

5.1 Elementos básicos

5.1.1 Universo

Los elementos de un conjunto borroso son tomados de un universo cualquiera. El universo contiene todos los elementos que pueden ser considerados. Aún el universo depende de otro universo mayor [2].

- El conjunto de gente joven puede contener a todos los seres humanos como su universo.

Alternativamente éste podría ser el número entre 0 y 100; los cuales pueden representar la edad de las personas en dicho universo.

- El conjunto x$\gg$10 (mucho mayor que 10) podría tener como universo a todos los números positivos.

Una aplicación de el universo es el suprimir a todos los datos no necesarios, por ejemplo a todos los números negativos en el ejemplo anterior.

En caso que estemos tratando con una cantidad no numérica, por ejemplo saborear, la cual no puede ser medida en una escala numérica, no podremos usar un universo numérico [4]. Los elementos son entonces tomados de *continuidad lógica*; como un ejemplo de dicho universo se tiene:

$$(agrio,\ dulce,\ amargo,\ salado,\ caliente, \ldots.).$$

5.1.2 Soporte

El soporte de un conjunto borroso A en el universo U es un conjunto que contiene todos los elementos de U que no tiene valores de pertenencia cero en A, ésto es

$$supp(A) = \{x \in U \mid \mu_A(x) > 0\}$$

donde supp(A) denota el soporte de un conjunto borroso A. Por ejemplo el soporte de el conjunto borroso "varios" en el conjunto de enteros $\{3, 4, 5, 6, 7, 8\}$. Si el soporte de un conjunto borroso es vacío, se le llama conjunto borroso vacío. Un singleton borroso es un conjunto borroso cuyo soporte es un solo punto en U.

5.1.3 Centro

El centro de un conjunto borroso es definido como sigue: si el valor medio de todos los puntos a los que la función de pertenencia de un conjunto borroso alcanza su máximo valor [4], es finito, entonces se define este valor medio, como el centro de un conjunto borroso; si el valor medio es infinitamente positivo o negativo, entonces el centro es definido como muy pequeño o muy grande entre todos los valores de puntos que alcanzan el valor máximo de pertenencia.

5.1.4 Punto intermedio

El punto intermedio de un conjunto borroso, es aquel en el cual el valor de A es igual a $\frac{1}{2}$.

5.1.5 Punto alto

El punto alto de un conjunto borroso es el valor más grande alcanzado por cualquier punto. El valor más alto que un elemento puede alcanzar es 1.

5.1.6 Corte-α

El $corte - \alpha$ de un conjunto borroso A es un conjunto A_α que contiene a todos los elementos en U que tienen valores de pertenencia en A, mayores o iguales a α, esto es

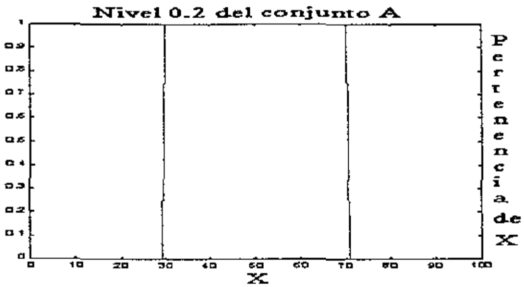
$$A_\alpha = \{x \in U \mid \mu_A(x) \geq \alpha\}$$

Por ejemplo, para α=0.3, the $corte - \alpha$ de el conjunto borroso [-1,1] es el conjunto [-0.7,0.7] y para $\alpha = 0.9$, esto es [-0.1,0.1.] Para comprender más el corte alfa, se puede considerar como el complemento de un determinado conjunto. Considerando el conjunto borroso cuya función de pertenencia es:

$$\mu_A(x) = \frac{1}{1 + 0.01 * (x - 50)^2}$$

Suponiendo que estamos interesados en la porción de la función de pertenencia donde el soporte es mayor que 0.2.

El corte alfa = 0.2 se grafica.



Donde el intervalo de 30 a 70 es el corte-α. La siguiente matriz elaborada en el programa de Matlab nos despliega la tabla de valores en forma de columnas, donde se nos muestra todos los grados de pertenencia o membresía que son mayores o iguales a α=0.2.

$a = [1/(1+0.01*(10-50)^2)\ 1/(1+0.01*(20-50)^2), ..., \ 1/(1+0.01*(100-50)^2)]$

a = Columnas 1 hasta 7

0.0588 0.1000 **0.2000 0.5000 1.0000 0.5000 0.2000**

Columnas 8 hasta 10

0.1000 0.0588 0.0385

5.2 Normalización

Un conjunto borroso es normalizado si el mayor valor de pertenencia es igual a 1. Se puede normalizar un conjunto dividiendo a todos sus elementos entre el mayor en el conjunto a/max(a).

5.3 Singletons

Estrictamente hablando, un conjunto borroso A es una colección de pares ordenados [5].

$$A = \{(x, \mu(x))\}.$$

El elemento x pertenece a el universo y $\mu(x)$ es su grado de pertenencia en A. Un solo par (x,$\mu(x)$) es llamado borroso *singleton*; así, el conjunto completo puede ser visto como la unión de sus constituyentes singletons. Algunas veces es conveniente pensar que un conjunto es tan sólo un vector.

$$\mathbf{a} = (\mu(x1), \mu(x2), \dots, \mu(xn))$$

Se entiende que cada punto i (1,2,...,n) corresponde a un punto en el universo de n puntos.

Bibliografía

[1] Abraham Kandel (1986), *Fuzzy Mathematical Techniques with Applications*, Addison-Wesley Publishing Company, ISBN 0-201-11752-5, páginas 28-44, 72-97.

[2] R. Giles (1976), *Lukasiewicz Logic and fuzzy set theory*, International Journal Man-Machine Studies, vol. 8, páginas 313-327.

[3] Siegfried Weber (1983), *Fuzzy Sets and Systems II*, A General Concept of Fuzzy Connectives, Negations and Implications Based On + - Norms and + - Conorms, North Holland, páginas 115 - 134.

[4] Wyllis Bandler and Ladislav Kohout (1980), *Fuzzy Sets and Systems 4*, Fuzzy Power Sets and Fuzzy Implication Operators, North-Holland Publishing, páginas 13-30.

Capítulo 6

OPERACIONES CON CONJUNTOS BORROSOS

6.1 Álgebra de conjuntos

La operación de conjuntos borrosos crea un nuevo conjunto a partir de uno o varios conjuntos dados [1]; donde si tenemos los conjuntos A y B, la intersección de ambos conjuntos será un conjunto borroso nuevo con su propia función de pertenencia. Los operadores básicos para el complemento, unión e intersección de conjuntos borrosos son los siguientes:

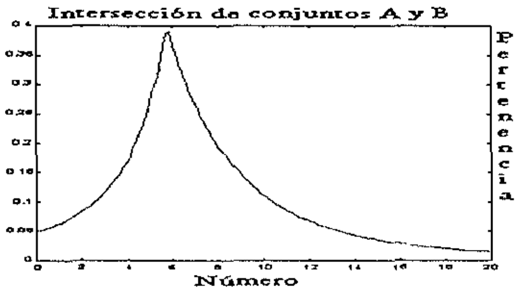
$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \tag{6.1}$$

$$\mu_{A\cup B}(x) = max[\mu_A(x), \mu_B(x)] \tag{6.2}$$

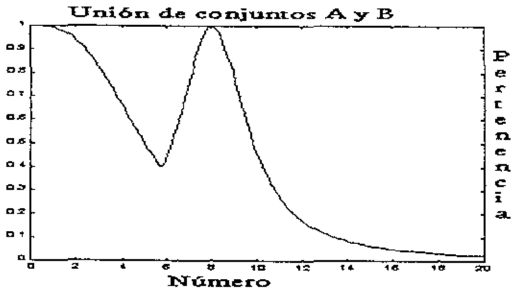
$$\mu_{A\cap B}(x) = min[\mu_A(x), \mu_B(x)] \tag{6.3}$$

Sea A y B conjuntos borrosos en un universo. El diagrama de Zadeh [2] es una representación gráfica que muestra el grado de pertenencia de los valores de entrada a conjuntos borrosos. El diagrama de Zadeh para dos funciones de pertenencia A (pequeños números) y B (alrededor de 8) se muestra a continuación.

La intersección de A y B es A ∩B ≡ **a min b**. El operador min es un comparador mínimo de elemento a elemento entre dos correspondientes elementos [2] **a** y **b**.



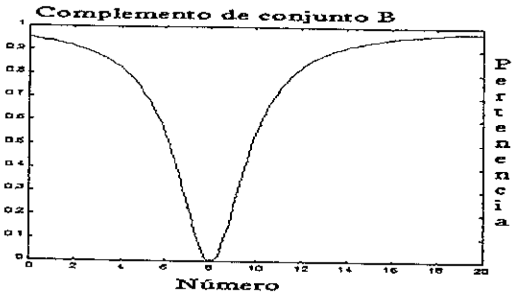
La unión de A y B es A $\cup$B $\equiv$ **a max b** donde **max** es un comparador máximo de elemento a elemento.



Un conjunto borroso X es un *subconjunto borroso* de el conjunto Y, escrito $X \subseteq Y$, si su función de relación es menor o igual a la función de relación de Y.

El complemento de A es $\bar{A}$ $\equiv$1-**a** donde cada valor de pertenencia en **a** es sustraido a 1.

La gráfica siguiente muestra el grado de pertenencia del complemento del conjunto B previamente mencionado.



6.1.1 Ejemplo

Comprar un carro. Una familia de 4 integrantes quiere comprar un carro. Un indicador de cuán confortable quieren ellos que éste sea, es el número de asientos en el mismo. Pero ellos también quieren que sea amplio. Sea **u** = (1,2,3,4,5,6,7,8,9,10) el conjunto de carros disponibles descritos por el número de asientos. Entonces el conjunto borroso **c** (para confortable) puede ser descrito como:

c=[0.2 0.5 0.8 1 0.7 0.3 0 0 0 0]

y sea I el conjunto borroso amplio definido: **I**=[0 0 0.2 0.4 0.6 0.8 1 1 1 1]

La intersección de confortable y amplio es entonces:

c ∩ I=[0 0 0.2 0.4 0.6 0.3 0 0 0 0]

La interpretación de la respuesta es que cinco asientos es óptimo, pero sólo satisface en un grado de 0.6. La segunda solución son 4 asientos.

La unión de confortable y amplio es entonces:

c∪**I**=[0.2 0.5 0.8 1 0.7 0.8 1 1 1 1]

Aquí, cuatro asientos es satisfactorio porque éste es confortable, y de 7-10 asientos también, porque esto significaría que es un auto grande o amplio.

El complemento de amplio es:

$$\bar{I} = [1\ 1\ 0.8\ 0.6\ 0.4\ 0.2\ 0\ 0\ 0\ 0].$$

Los operadores ∪ y ∩ asocian, conmutan y más. Éstas propiedades son importantes, porque ellas ayudan a predecir la solución de sentencias largas [4].

6.2 Norma-S y Norma-t como operadores

Sea S una función que asocia a A y B en la unión de ambos, entonces se le llamará una *NORMA-S* que requiere de cuatro axiomas para que se cumpla [2]. Cualquier función que cumple con dichos axiomas se le llama norma-s y algunas de tales funciones son las siguientes:

- Clase Dombi (Dombi[1982]):

$$s_\lambda(a,b) = \frac{1}{1 + [(\frac{1}{a}-1)^{-\lambda} + (\frac{1}{b}-1)^{-\lambda}]^{\frac{-1}{\lambda}}} \tag{6.4}$$

  $\lambda \in (0,\infty)$

- Clase Dubois-Prade (Dubois and Prade [1980]:)

$$s_\alpha(a,b) = \frac{a+b-ab-min(a,b,1-\alpha)}{max(1-a,1-b,\alpha)} \tag{6.5}$$

  $\alpha \in [0,1]$.

- Clase Yager (Yager [1980]):

$$s_\omega(a,b) = min[1,(a^w + b^w)^{\frac{1}{w}}] \tag{6.6}$$

  $\omega \in (0,\infty)$.

Otras definiciones de tales operadores son conocidas, sin embargo, las designaciones **max** y **min** son más comunes.

- Suma drástica:

$$S_{ds}(a,b) = \begin{cases} a, & \text{si b} = 0; \\ b, & \text{si a} = 0\ ; \\ 1, & \text{en cualquier otro caso} \end{cases} \tag{6.7}$$

- Suma Einstein:

$$S_{es}(a,b) = \frac{a+b}{1+ab} \tag{6.8}$$

- Suma Algebraica:

$$S_{as}(a,b) = a + b - ab \tag{6.9}$$

- Máximo: (6.2)

Sea t una función que transforma las funciones de pertenencia de dos conjuntos borrosos A y B en la función de pertenencia de la intersección de

A y B. A tal función se le llama *NORMA-T* que contiene cuatro axiomas para que dicha norma se cumpla [2]. La función borrosa básica **min** es una t - norma [5] .

A continuación otras normas-t:

- Clase Dombi (Dombi[1982]):

$$t_\lambda(a,b) = \frac{1}{1 + [(\frac{1}{a} - 1)^\lambda + (\frac{1}{b} - 1)^\lambda]^{\frac{1}{\lambda}}} \tag{6.10}$$

$\lambda \in (0,\infty)$

- Clase Dubois-Prade (Dubois and Prade [1980]:)

$$t_\alpha(a,b) = \frac{ab}{max(a,b,\alpha)} \tag{6.11}$$

$\alpha \in [0,1]$.

- Clase Yager (Yager [1980]):

$$t_\omega(a,b) = 1 - min[1, ((1-a)^w + (1-b)^w)^{\frac{1}{w}}] \tag{6.12}$$

$\omega \in (0,\infty)$.

- Producto Drástico :

$$t_{dp}(a,b) = \begin{cases} a, & \text{si b = 1;} \\ b, & \text{si a = 1;} \\ 0, & \text{en cualquier otro caso} \end{cases} \tag{6.13}$$

- Producto Einstein :

$$t_{ep}(a,b) = \frac{ab}{2-(a+b-ab)} \tag{6.14}$$

- Producto Algebráico :

$$t_{ap}(a,b) = ab \tag{6.15}$$

- Minimo: (6.3)

Y para finalizar todo el espectro de posibilidades tenemos los operadores intermedios:

- Max-min promedios:

$$v_\lambda(a,b) = \lambda\, max(a,b) + (1-\lambda)\, min(a,b) \tag{6.16}$$

$\lambda \in$ [0,1]

- Media generalizada:

$$v_\alpha(a,b) = (\frac{a^\alpha + b^\alpha}{2})^{\frac{1}{\alpha}} \tag{6.17}$$

$\alpha \in R(\alpha \neq 0)$

- Y borrosa:

$$v_p(a,b) = p\, min(a,b) + \frac{(1-p)(a+b)}{2} \tag{6.18}$$

$p \in$ [0,1]

- O borroso:

$$v_\gamma(a,b) = \gamma\, max(a,b) + \frac{(1-\gamma)(a+b)}{2} \tag{6.19}$$

$\gamma \in$ [0,1]

Bibliografía

[1] W. Bandler and L. Kohout (1980), *Fuzzy power sets and fuzzy implication operators*, Fuzzy Sets and Systems, vol. 4, páginas 13-30.

[2] Li-xin Wang (1997), *A Course in Fuzzy Systems and Control*, Prentice Hall PTR, Upper Saddle River, NJ 07458, ISBN 013540882-2, páginas 20-46.

[3] J. Wesley Hines (1997), *Matlab Supplement to Fuzzy and Neural Approaches in Engineering*, A Wiley-Interscience Publication, New York 10158-0012, ISBN 0-471-19247-3, páginas 1-8.

[4] J. Dombi (1982), *A general class of fuzzy operators, De Morgan class of fuzzy operators and fuzziness measures induced by fuzzy operators*, Fuzzy Sets and Systems, vol. 8, páginas 149-163.

Capítulo 7

LÓGICA BORROSA

La lógica comenzó como el estudio de el lenguaje en argumentos y persuaciones, pudiéndose usar ésta para juzgar la veracidad de una cadena de razonamientos de una manera matemática comprobable. En la lógica bivaluada una proposición puede ser verdadera o falsa, pero nunca ambas. La veracidad o falsedad que se le asigna a un enunciado, se le llama su valor de verdad. En la lógica borrosa [1, 2] una proposición puede ser verdadera o falsa, o aún tener un valor intermedio de verdad y falsedad, tal como podría ser *a lo mejor verdadero*. La sentencia *el nivel de agua en un tinaco es alto* es un ejemplo de dicho tipo de proposiciones en un controlador borroso. Puede ser conveniente el restringir a los posibles valores de verdad a un dominio discreto, como { 0, $\frac{1}{2}$, 1} para los valores de falso, a lo mejor verdadero y verdadero; en tal caso nos encontramos con lógica multivaluada.

Para obtener conclusiones a partir de las sentencias *Si.. entonces* existen reglas de inferencia lógica que son una derivación de las reglas de inferencia de la lógica clásica como son:

- Modus Ponens (p$\wedge$ (p $\rightarrow$ q))$\rightarrow$ q

$$\mu'_B(y) = \sup_{x \in U} t[\mu'_A(x), \mu_{A \rightarrow B}(x, y)]$$

  Premisa 1: x es A

  Premisa 2: si x es A entonces y es B

  Conclusión: y es B

- Modus Tollens ($\bar{p}\wedge$(p$\rightarrow$ q))$\rightarrow$ $\bar{p}$

$$\mu'_A(x) = \sup_{y \in V} t[\mu'_B(x), \mu_{A \rightarrow B}(x, y)]$$

  Premisa 1: y es no B

  Premisa 2: si x es A entonces y es B

  Conclusión: x es no A

- Silogismo Hipotético ((p→q)∧(p→r))→ (p→r)

$$\mu_{A'\to C'}(x,z)=\sup_{y\in V} t[\mu_{A'\to B'}(x,y),\mu_{B'\to C'}(y,z)]$$

Premisa 1: si x es A entonces y es B

Premisa 2: si y es B entonces z es C

Conclusión : si x es A entonces z es C.

Podemos usar diferentes normas-t con distintas reglas de implicación y obtener una diversidad de resultados.

Lógica borrosa es un sistema metodológico para resolver problemas de control, que nos lleva a la implementación desde pequeños sistemas de control, hasta micro controladores a gran escala, redes, bases de datos, etc. Ésta incorpora una simple base de reglas **Si x es y entonces z** para resolver el problema de control en vez de diseñar un modelo matemático de el sistema [3]. La lógica borrosa se basa en crear modelos empíricos de sentencias. La lógica borrosa requiere de algunos parámetros numéricos a fin de operar en términos de un error significante y un cambio en el error, pero valores exactos de estos números, no son usualmente críticos, a menos que así lo requiera el sistema y en tal caso también se podrá determinar empíricamente.

La lógica borrosa ofrece muchas características que la hacen particularmente buena opción para muchos problemas de control [7].

- Es inherentemente robusta debido a que no requiere entradas precisas y libres de ruido, además de que pueden ser programadas para que funcionen en caso de que un sensor en retroalimentación fallare. La salida de control es una función cómoda, a pesar de llegar a tener muchas entradas en el sistema.

- Ya que el controlador borroso utiliza reglas para gobernar la tarea de control, ésta puede ser modificada y debilitada fácilmente para mejorar o modificar completamente el funcionamiento de el sistema. Nuevos sensores pueden ser añadidos con tan sólo modificar adecuadamente el sistema de reglas.

- No se limita a una cantidad pequeña de entradas de retroalimentación y una o dos entradas de control, ni tampoco es necesario el medir o computar los cambios de rango de los parametros, a fin de ser implementados. Cualquier sensor puede proporcionar información para que funcione el sistema.

- Lógica borrosa puede controlar sistemas no-lineales que de otra manera sería difícil o imposible de modelar matemáticamente.

7.1 Conectores

En cualquier conversación diaria o en el lenguaje matemático, las sentencias son conectadas con palabras como *y, o, si entonces* (implicaciones) *sí* y *sólo si*; las cuales son llamadas conectores. Una sentencia que es modificada [4] por la palabra *no* se le llama negación de la sentencia original. La palabra *y* es usada para juntar a dos sentencias para formar la conjunción de dos sentencias. De igual manera se pueden unir a dos sentencias con la palabra *o* y formar así una disyunción de dos sentencias. A partir de dos sentencias se puede construir una sola sentencia con la forma **Si....entonces** que se llamará sentencia condicional. La sentencia que es seguida de la palabra **Si** se le llama *el antecedente* y la sentencia que sigue a la palabra **entonces** se le llama *el consecuente*. La palabra **Sí sólo si** es usada para obtener a partir de dos sentencias, una sola sentencia bicondicional.

A continuación se muestran algunos símbolos empleados en lógica para representar a los conectores [5, 6].

$\vee$ para o; $\wedge$ para y; $\Rightarrow$ para Si...entonces; $\Leftrightarrow$ para Sí sólo si;

$\neg$ para no

Los conectores tales como Y y O, y modificadores tales como NO, Muy, y Más o menos pueden ser usados para generar valores compuestos.

- O, corresponde a la composición max o unión
- Y, corresponde a composición min o intersección
- NO, corresponde a el complemento
- Muy y Más o menos, corresponden a varios grados de intensificación

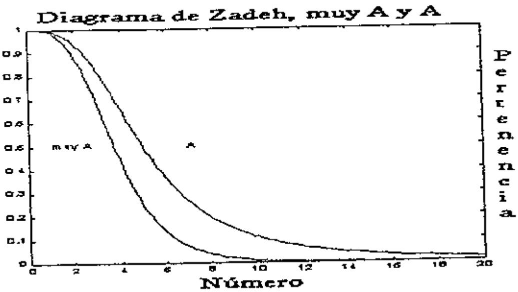
7.2 Modificadores

Un modificador lingüístico, es una operación que modifica el significado de un término. Por ejemplo en el enunciado "muy cercano a cero", la palabra *muy* modifica a *cercano a cero* el cual es un conjunto borroso. Un modificador es una operación con conjuntos borrosos. Otros modificadores también pueden ser (un poco, más o menos, posiblemente y definitivamente).

El modificador puede ser representado de manera matemática como sigue:

muy A $\equiv A^2$ también conocido como concentración del conjunto borroso A,

más o menos A$\equiv A^{0.5}$ conocido como dilatación del conjunto borroso A.



En una representación discreta, sea el universo **U**=(0, 20, 40, 60, 80). Y dado un conjunto borroso:

joven = [1, 0.6, 0.1, 0, 0]

entonces podemos derivar la función de pertenencia discreta para el conjunto muy joven por medio de la exponenciación cuadrada de cada uno de sus elementos,

$joven^2 = [1, 0.36, 0.01, 0, 0]$

donde el conjunto muy, muy joven se induce de la siguiente manera:

$joven^4 = [1, 0.13, 0, 0, 0]$.

Los conjuntos derivados heredan el universo de el conjunto primario. Otros ejemplos de modificadores son:

extremadamente a = a^3

levemente a = $a^{1/3}$

Una familia completa de modificadores puede ser generada por a^p donde p es alguna potencia entre cero e infinito.

Bibliografía

[1] Brubaker D. I. (1991), *Introduction to Fuzzy Logic Systems*, The Huntington Group. Menlo Park, CA.

[2] Li-xin Wang (1997), *A Course in Fuzzy Systems and Control*, Prentice Hall PTR, Upper Saddle River, NJ 07458, ISBN 013540882-2, páginas 73-86.

[3] Klir George J. and Yuan Bo. (1995), *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Upper Saddle River, NJ., Prentice Hall.

[4] S. Weber (1983), *A general concept of fuzzy connectives, negations and implications based on t-norms and t-conorms*, Fuzzy Sets and Systems, vol. **11**, páginas 115-134.

[5]] W. Bandler and L. Kohout (1980), *Fuzzy power sets and fuzzy implication operators*, Fuzzy Sets and Systems, vol. 4, páginas 13-30.

[6] J. Dombi (1982), *A general class of fuzzy operators. De Morgan class of fuzzy operators and fuzziness measures induced by fuzzy operators*, Fuzzy Sets and Systems, vol. 8, páginas 149-163.

[7] Novák V. (1992), *Fuzzy Logic As a Basis of Approximate Reasoning*, In: Zadeh Lotfi A., Kacprzyk J, Fuzzy Logic for the Management of Uncertainty, J. Wiley & Sons, New York.

Capítulo 8

INFERENCIA BORROSA

A fin de obtener conclusiones provenientes de la base de reglas se necesita un mecanismo que produzca un resultado a partir de una colección de reglas *Si....entonces*. Esto es posible usando la regla de inferencia composicional (CROI) [1, 2, 3]. En este sentido la palabra inferencia significa concluir a partir de una evidencia dada, deducir o tener como una consecuencia lógica.

8.1 Relaciones entre conjuntos

. En cualquier controlador borroso, la relación entre objetos juega un rol fundamental. Algunas relaciones [1] afectan elementos dentro de el mismo universo: una medición es más grande que la otra, un evento ocurre primero que otro, etc. Otras relaciones afectan elementos de universos: la medición es mayor y su rango de cambio es positivo, el eje coordenado x es mayor mientras que el eje coordenado y es menor. Estos ejemplos son relaciones entre dos objetos, pero en la práctica podemos encontrarnos con relaciones que contienen mayor número de objetos.

Una relación binaria o simplemente una relación $\Re$ de un conjunto A a otro conjunto B, asigna a cada par ordenado (a,b) $\in A \times B$, exactamente uno de los siguientes enunciados es válido

(i) a está relacionado con b.

(ii) a no está relacionado con b.

El producto cartesiano $A \times B$ es el conjunto de todas las combinaciones posibles entre los elementos de ambos conjuntos. Una relación borrosa entre dos conjuntos A y B es un subconjunto de el producto cartesiano $U \times V$ entre sus respectivos universos U y V, o dicho de otra manera una relación borrosa es un conjunto borroso definido en el producto cartesiano de los conjuntos

$$U_1, U_2, ..., U_n$$

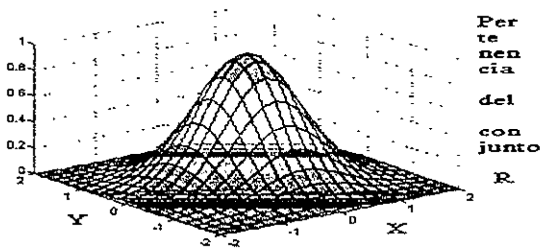
con el esquema representativo de un conjunto borroso:

$$A = \{(x, \mu_A(x)) | x \in U\} \tag{8.1}$$

Suponiendo que una relación R es definida como " x está cerca del origen AND cerca de y ". Esto puede ser expresado como:

$$\mu_A(x) = e^{-(x^2+y^2)}$$

El Universo de discurso se gráfica.



Asumiendo, por ejemplo que Javier, el sobrino de Papá se parece a Víctor en una escala de 0.8 grados, y que Javier se parece a Luis 0.9 grados.

Tenemos por lo tanto, una relación entre subconjuntos de sobrinos en la familia. Esto se puede representar en una matriz con un renglón.

R_1 =Javier (0.8, 0.9)

Víctor Luis

Una relación borrosa Q en

$$U_1 \times U_2 \times ... \times U_n$$

es definida como el conjunto borroso:

$$Q = \{((U_1, U_2, ..., U_n), \mu_Q(U_1, U_2, ..., U_n))|U_1, U_2, ..., U_n \in U_1 \times U_2 \times ... \times U_n\} \tag{8.2}$$

donde μ_Q:$U_1 \times U_2 \times ... \times U_n \rightarrow [0,1]$.

8.1.1 Composición

A fin de mostrar cómo dos relaciones pueden ser combinadas, se asumirá otra relación entre los mismos elementos.

$$R_2 = \begin{array}{lc} & \text{Papá} \\ \text{Víctor} & 0.5 \\ \text{Luis} & 0.6 \end{array}$$

Se puede intentar tratar de encontrar que tanto Javier se parece a Papá por medio de una combinación de ambas matrices:

1. Javier se parece 0.8 a Víctor, y Víctor se parece 0.5 a Papá.

2. Javier se parece 0.9 a Luis, y Luis se parece 0.6 a Papá.

El enunciado 1 contiene una cadena de relaciones, la cual se puede combinar con una operación de intersección. El resultado a dicha operación correspondería a escoger el valor de relación menor, de acuerdo con lo antes mencionado en operaciones con conjuntos borrosos.

Por lo tanto, el resultado a dicha combinación, considerando el valor más pequeño será:

3. Javier se parece 0.5 a Papá.

4. Javier se parece 0.6 a Papá. Tanto 3 como 4 parecen ser igualmente válidas, por lo que parece razonable el aplicar la operación **unión**. Esto correspondería a escoger la más fuerte relación de ambas y será así:

5. Javier se parece 0.6 a Papá.

La regla general cuando se está combinando o componiendo relaciones borrosas, es el tomar el valor más pequeño en conexiones tipo serie y un valor máximo en conexiones en paralelo, lo cual será conveniente hacer con un **producto interno**.

P∘Q es la composición de P(U,V) y Q(V,W) sí y sólo si:

$$\mu_{P\circ Q}(x,z) = \max_{y\in V} t[\mu_p(x,y), \mu_Q(y,z)]$$

para cada (x,y)$\in$ U$\times$ W, donde t es una norma-t.

La (CROI) regla composicional de inferencia mencionada anteriormente es de la siguiente forma:

$$\mu'_B(y) = \sup_{x \in U} t[\mu'_A(x), \mu_Q(x,y)]$$

El producto interno es similar a una matriz ordinaria (producto punto), excepto que la multiplicación es reemplazada por ($\cap$) y sumatoria por ($\cup$). Suponiendo que **R** es una matriz de $m \times p$ y **S** es otra matriz de p $\times$ n. Entonces el producto interno $\cup.\cap$ es una matriz $m \times n$ $\mathbf{T}$=(t_{ij}) donde ij son las entradas obtenidas por medio de combinar el renglón i-ésimo de **R** con la columna j-ésimo de **S**, tal que:

t_{ij}= $(r_{i1} \cap s_{1j})\cup$ $(r_{i2} \cap s_{2j})\cup ... \cup(r_{ip} \cap s_{pj}) = \bigcup_{k=1}^{p} r_{ik} \cap s_{kj}$. Como una notación para generalizar el producto interno, se usará $f \cdot g$, donde **f** y **g** son funciones cualquiera que toman dos argumentos, en este caso $\cup$ y $\cap$. Con las definiciones anteriores acerca de las operaciones de conjuntos, la composición se reduce a lo que se le llama *composición max-min* (Zadeh in Zimmermann, 1991). Si **R** es una relación de **a** a **b** y **S** es una relación de **b** a **c**, entonces la composición de **R** y **S** es una relación de **a** a **c** (ley transitiva).

8.1.2 Producto interno

Tomando los datos de los ejemplos anteriores R_1 y R_2 se tiene:

$R_1 \cup\!\cdot\!\cap R_2$= (0.8 0.9 $\cup\!\cdot\!\cap$ 0.5 0.6) = (0.8 $\cap$ 0.5)$\cup$(0.9 $\cap$ 0.6)

$$0.5 \cup 0.6 = \quad 0.6$$

Lo cual es igual al resultado anterior.

La composición **max-min** es distributiva con respecto a la **unión,**

$$(\mathbf{R}\cap\mathbf{T})\cup\!\cdot\!\cap\mathbf{S} = (\mathbf{R}\cup\!\cdot\!\cap\mathbf{S})\cup(\mathbf{T}\cup\!\cdot\!\cap\mathbf{S}),$$

pero no con respecto a la intersección. Algunas veces la operación **min** en la composición *max-min* es sustituida por de multiplicación; y por ello se le dá el nombre de composición max-star.

Las composiciones más utilizadas son:

- Composición max-min

$$\mu_{P \circ Q}(x,z) = \max\, min_{y \in V}[\mu_p(x,y), \mu_Q(y,z)]$$

.

donde (x,z) $\in$ U $\times$ W.

- Composición max-product

$$\mu_{P \odot Q}(x,z) = \max_{y \in V}[\mu_p(x,y) \cdot \mu_Q(y,z)]$$

donde (x,z) $\in$ U $\times$ W.

8.1.3 Ejemplo de composición max-min.

Sea X={x1,x2}, Y={y1,y2,y3} y Z={z1,z2}.

Dadas dos relaciones borrosas R1(X,Y) y R2(Y,Z).

$$\mathbf{R1} \circ \mathbf{R2} = \begin{pmatrix} 0.3 & 0 & 0.7 \\ 0.8 & 1 & 0.4 \end{pmatrix} \begin{pmatrix} 0.5 & 1 \\ 0 & 0.9 \\ 0.2 & 0.6 \end{pmatrix}$$

```
EDU R1=[0.3 0 0.7;0.8 1 0.4];

EDU R2=[0.5 1;0 0.9;0.2 0.6];

EDU [r1,c2]=size(R1); [r2,c2]=size(R2);

EDU R0=zeros(r1,c2);

EDU for i=1:r1;

for j=1:c2;

R0(i,j)=max(min(R1(i,:),R2(:,j)'));

end

end

EDU R0
```

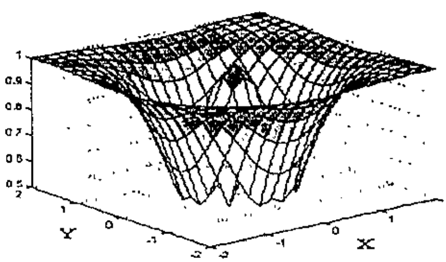
R0 =

0.3000 0.6000

0.5000 0.9000

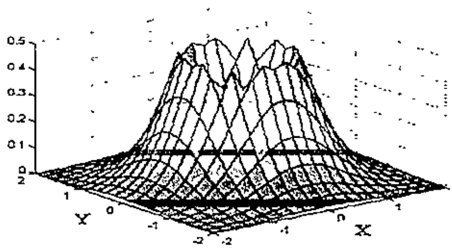
Suponiendo que tenemos una relación R1 que es definida como " x está cerca de y AND cerca del origen " , y una relación R2 es definida como " x No está cerca de el origen ". La unión R1 OR R2 se muestra en la gráfica.

$$\mu_{R1}(x) = e^{-(x^2+y^2)} \quad \mu_{R2}(x) = 1 - e^{-(x^2+y^2)}$$



La intersección de las relaciones borrosas R1 AND R2 se grafica a continuación.

$$\mu_{R1}(x) = e^{-(x^2+y^2)} \quad \mu_{R2}(x) = 1 - e^{-(x^2+y^2)}$$



8.2 Implicaciones como relaciones borrosas

Si se considera una sentencia que diga: **Si** el nivel de agua en un tanque es bajo, entonces si se abre la $llave_1$; entonces se tendrá una implicación, debido a que el valor de el nivel del agua implicará el valor de la $llave_1$ en un controlador dado.

Existen algunos tipos de implicaciones, pero a continuación se mencionará la **Implicación de Mamdani** (*Mamdani* 1977) [1].

$$\mathbf{a} \Rightarrow \mathbf{a} \equiv \mathbf{a}\ \mathbf{min}\ \mathbf{b}$$

donde **min** es el producto externo, aplicando **min** a cada elemento de el producto cartesiano de **a** y **b**. Donde **a** puede ser representado por un vector columna, y **b** por un vector renglón y así el producto externo podrá ser encontrado como en la aplicación de una multiplicación entre los miembros de dicha matriz. Otros tipos de implicaciones son:

- Implicación Dienes-Recher Si se tiene la implicación:

$$Si \quad fp_1 \quad entonces \quad fp_2$$

  ésta es interpretada como una relación borrosa Q_D en $U \times V$ con la función de pertenencia:

$$\mu_{QD}(x,y) = max[1 - \mu_{fp1}(x), \mu_{fp2}(y)]$$

  que es similar a escribir $\tilde{p} \vee q$ en lógica clásica

- Implicación Lukasiewicz usando s-norma, w=1,

$$\mu_{QL}(x,y) = min[1, 1 - \mu_{fp1}(x) + \mu_{fp2}(y)]$$

- Implicación de Zadeh

$$\mu_{QZ}(x,y) = max[min(\mu_{fp1}(x), \mu_{fp2}(y), 1 - \mu_{fp1}(x)]$$

- Implicación de Gödel

$$\mu_{QG}(x,y) = \begin{cases} 1, & \text{si } \mu_{fp1}(x) \leq \mu_{fp2}(y); \\ \mu_{fp2}(y), & \text{En cualquier otro caso} \end{cases}$$

- Implicaciones Mamdani

$$\mu_{QMM}(x,y) = min[\mu_{fp1}(x), \mu_{fp2}(y)]$$

$$\mu_{QMP}(x,y) = \mu_{fp1}(x)\, \mu_{fp2}(y)$$

8.3 Dispositivos de inferencia

Como ya se ha visto en los anteriores capítulos, dependiendo de el tipo de composición que se tenga entre dos conjuntos borrosos, la base de inferencia puede ser de muchos tipos:

- Mecanismo de inferencia de producto

  a) se aplica en base de reglas individuales con combinación unión

b) se utiliza implicación del producto tipo Mamdani

c) se utiliza producto algebraico para todas las normas-t y max para normas-s.

$$\mu_{B'}(y) = \max_{t=1}^{M}[sup(\mu_{A'}(x) \prod_{i=L}^{n} \mu^L_{A'i}(xi)\mu^L_{B'}(y))]$$

- Mecanismo de inferencia del Mínimo

a) se aplica en base de reglas individuales con combinación unión

b) se utiliza implicación del mínimo tipo Mamdani

c) se utiliza min para todas las t-normas y max para normas-s.

$$\mu_{B'}(y) = \max_{t=1}^{M}[sup\, min(\mu_{A'}(x), \mu^L_{A'1}(x1), ..., \mu^L_{A'n}(xn), \mu^L_{B'}(y))]$$

Si se utiliza un conjunto borroso A' que sea singleton, el mecanismo de inferencia se reduce a:

$$\mu_{B'}(y) = \max_{t=1}^{M}[min(\mu^L_{A'1}(x1^*), ..., \mu^L_{A'n}(xn^*), \mu^L_{B'}(y))]$$

donde x^* en algún punto en U.

- Mecanismo de inferencia de Lukasiewicz

a) se aplica en base de reglas individuales con intersección

b) se utiliza implicación de Lukasiewicz

c) se utiliza min para las normas-t

$$\mu_{B'}(y) = \min_{t=1}^{M}[sup\, min(\mu_{A'}(x), \mu_{Ru}^{L}(x,y))]$$

$$= \min_{L=1}^{M}\{sup\, min[\mu_{A'}(x), 1 - \min_{i=1}^{n}(\mu_{Ai}^{L}(xi)) + \mu_{B'}^{L}(y)]\}$$

- Mecanismo de inferencia de Zadeh

a) se aplica en base de reglas individuales con intersección

b) se utiliza implicación de Zadeh

c) se utiliza min para las normas-t

$$= \mu_{B'}(y) = \min_{i=1}^{M}\{sup\, min[\mu_{A'}(x), \tag{8.3}$$

$$max(min_{(}\mu_{Ai}^{L}(xi)), ..., \mu_{A'n}^{L}(xn), \mu_{B'}^{L}(y), 1 - \min_{i=1}^{n}(\mu_{Ai}^{L}(xi)))]\} \tag{8.4}$$

- Mecanismo de inferencia de Dienes-Rescher

a) como en Zadeh pero con implicación Dienes

$$\mu_{B'}(y) = \min_{i=1}^{M}[sup\, min[\mu_{A'}(x), max(1 - \min_{i=1}^{n}(\mu_{Ai}^{L}(xi)), \mu_{B'}^{L}(y))]$$

Si el conjunto A' es un conjunto borroso singleton se reducen los términos a: A'=1 y sup se elimina. Los anteriores mecanismos de inferencia fueron el resultado de la combinación de los incisos a, b, c; La combinación de todos los tipos de implicaciones, las distintas normas nos dá una combinatoria muy grande, pero sólo algunas de tales combinaciones son de real significado en la teoría de control.

8.4 Borrosificador

El borrosificador establece una relación entre puntos de entrada no borrosos al sistema $x = (x1, ...xn)^t$, y sus correspondientes conjuntos borrosos A en U (Las variables procedentes de el exterior serán, en general, valores no borrosos, y habrá que borrosificarlas previamente). Se pueden utilizar diversas estrategias de borrosificación:

- Borrosificador singleton. Es el método de borrosificación más utilizado, principalmente en sistemas de control, y consiste en considerar los propios valores discretos como conjuntos borrosos. Desde otra forma, para cada valor de entrada x se define un conjunto A' que lo soporta, con función de pertenencia $\mu_A(x)$, de modo que $\mu(x) = 1$,(x'=x), y $\mu_A(x') = 0$, para todos los otros x' $\in$ U en los que x'$\neq$x.

- En este método de borrosificación se utiliza una función exponencial del tipo siguiente,

$$\mu_A(x') = a.exp[-(\frac{x' - x}{\sigma})^2] \tag{8.5}$$

función con forma de campana, centrada en el valor x de entrada, de anchura σ y amplitud a.

8.5 Desborrosificador

El conjunto borroso resultante de la inferencia debe ser convertido a un número que debe ser enviado a el proceso como una señal de control. A ésta operación se le llama desborrosificación.

El conjunto borroso resultante es, pues, desborrosificado a una señal precisa de control.

8.5.1 Métodos de desborrosificación:

Centro de gravedad (COG)

El valor preciso de salida u es la abscisa bajo el centro de gravedad de el conjunto borroso.

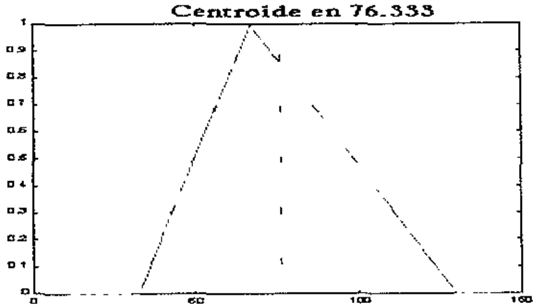
$$u = \frac{\Sigma_i \mu(x_i) x_i}{\Sigma_i \mu(x_i)} \tag{8.6}$$

Donde x_i es un punto cualquiera en un universo discreto, y $\mu(x_i)$ es su valor de pertenencia en la función de pertenencia. La expresión puede ser interpretada como el valor intermedio de los elementos que sustentan a dicho conjunto. Para el caso donde el universo sea continuo, se reemplaza la sumatoria por integrales. A éste método también se le llama *centroide de área*.

La siguiente gráfica se realiza en MATLAB ver. 5. [6]

```
x=[10:150];
EDU y=triangle(x,[32 67 130]);
EDU center=centroid(x,y);
EDU c_plot(x,y,center,'CENTROIDE);
```

EDU c_plot(x,y,center,'CENTROIDE');



Método de centro de gravedad para singletons (COGS)

Si la función de pertenencia de las conclusiones son singletons, el valor de salida será:

$$u = \frac{\Sigma_i \mu(s_i) s_i}{\Sigma_i \mu(s_i)} \tag{8.7}$$

Donde s_i es la posición de el singleton i en el universo, y $\mu(s_i)$ es igual a la fuerza de disparo α_i de la regla i. Este método tiene una complejidad computacional relativamente baja, y u es diferenciable con respecto a singletons s_i, el cual es muy usado en sistemas neuroborrosos.

Bisector de área (BOA)

Este método toma la abscisa de la linea vertical que divide a el área bajo la curva en dos curvas iguales. Esto en el caso continuo.

$$u = \{x \mid \int_{\mathrm{Min}}^{x} \mu(x)dx = \int_{x}^{\mathrm{Max}} \mu(x)dx\}$$

Aquí x es el conjunto móvil en el universo, $\mu(x)$ es la pertenencia, Min es el valor más lejano a la izquierda de el universo, y Max es el valor más lejano a la derecha. Su complejidad computacional es relativamente alta, y puede ser ambigua. Por ejemplo, si el conjunto borroso consiste de dos singletons en cualquier punto entre dos, se dividiría el área en 2 partes; consecuentemente, es más seguro decir que en el caso discreto, BOA no está definido.

Bibliografía

[1] Li-xin Wang (1997), *A Course in Fuzzy Systems and Control*, Prentice Hall PTR, Upper Saddle River, NJ 07458, ISBN 013540882-2, páginas 48-84.

[2] Kóczy L. T. (1995), *Algorithmic Aspects of Fuzzy Control*, International Journal of Approximate Reasoning, **12**, páginas 159-219.

[3] Dvorák A. (1996), *Properties of the Generalized Fuzzy logic Inference*, Proceeding of SIC'96, Budapest, páginas 75-80.

[4] Bart Kosko (1997), *Fuzzy Engineering*, Prentice Hall, ISBN 0-13-124991-6

[5] Witold Pedrycz (1993), *Fuzzy Control and Fuzzy Systems*, Research studies Press LTD, second extended edition, ISBN 0-471-93475-5.

[6] J. Wesley Hines (1997), *Matlab Supplement to Fuzzy and Neural Approaches in Engineerıng*, A Wiley-Interscience Publication, New York 10158-0012, ISBN 0-471-19247-3, página 30.

[7] Chin-Teng Lin and C.S. George Lee (1996), *Neural Fuzzy Systems, A Neuro-fuzzy synergism to Intelligent Systems*, Prentice Hall P T R, ISBN 0-13-235169-2, páginas 156-159.

Capítulo 9

DISEÑO DE SISTEMAS DE CONTROL BORROSO

La razón principal para diseñar sistemas de control borroso es, como ya se mencionó, la gran dificultad de el modelaje y simulación de sistemas complejos de control que se encuentran en el mundo real, especialmente cuando tareas de implementación son consideradas. Aún si se contara con un relevante modelo matemático de su sistema dinámico, sería muy laboriosa su resolución e implementación.

La metodología de control borroso nos ayuda a representar, manipular e implantar conocimiento heurístico humano para controlar un sistema [1].

Los controladores borrosos constan de cuatro principales componentes:

- La base de reglas que contiene el conocimiento, en la forma de conjuntos de reglas que nos ayudarán a controlar mejor el sistema.

- El mecanismo de inferencia, que evalua cuales reglas de control son relevantes en un momento dado y después decide cual entrada a la planta será la más adecuada.

- La interface de borrosificación, que simplemente modifica las entradas para que puedan ser interpretadas y comparadas a las reglas en la base de reglas.

- La interface de desborrosificación, que convierte las conclusiones alcanzadas por medio de el mecanismo de inferencia a las entradas a la planta.

Básicamente, se puede ver al controlador borroso como un tomador de decisiones artificial que opera en un sistema de lazo cerrado en tiempo real. Éste reune información de la salida y(t), compara ésta con la referencia de entrada r(t), y después decide cual deberá ser la entrada a la planta u(t), para asegurar que los objetivos del controlador se cumplan.

Para diseñar un controlador borroso, el ingeniero en control debe reunir información, que algunas veces esta información puede venir de un operador humano que realiza operaciones de control, mientras que en otras ocasiones el ingeniero de control puede recopilar información a partir de el entendimiento adecuado de la dinámica de la planta y transmitirlo así a una serie de reglas que muestren cómo controlar el sistema sin necesidad de ayuda exterior. Las reglas, básicamente dicen; Si la salida en la planta y las referencias de entrada se comportan de cierta manera, entonces la entrada a la planta deberá ser de cierto valor. Un conjunto completo de estas reglas es cargado a la base de reglas, y una estrategia de inferencia es escogida, entonces el sistema está listo para ser probado, para ver si las especificaciones de lazo cerrado son alcanzadas.

9.1 Cómo implementar la lógica borrosa

A continuación se enlista un procedimiento para implementar un sistema de. control borroso. Este método propuesto está basado en la recopilación de información obtenida.

1. Defina los objetivos de control y criterios: Qué estoy tratando de con-

trolar?, Qué tengo que hacer para controlar el sistema?, Qué clase de respuesta necesito?, Cuáles son las probables fallas del modelo?.

2. Determine las relaciones de entrada y salida, y escoja un número mínimo de variables de entrada al mecanismo de lógica borrosa (típicamente error y rango en el cambio de error).

3. Usando la estructura de base de reglas, desglose el problema de control en una serie de reglas. Si X y Y...entonces Z que defina la salida deseada del sistema para las condiciones de entrada dadas. El número y complejidad de las reglas depende en el número de parámetros de entrada que serán procesados y el número de variables borrosas asociadas con cada parámetro. Si es posible se recomienda usar al menos una variable y su derivada en el tiempo.

4. Cree las funciones de relación que definan a los valores de entrada/salida asociadas a las reglas.

5. Pruebe el sistema, evalue los resultados, ajuste las reglas y las funciones de pertenencia, y trate de nuevo hasta alcanzar los resultados esperados.

9.2 Ejemplos de controladores

Considere un sistema borroso de dos entradas y una salida cuya estructura en reglas es la siguiente:

Si XI es Al y X2 es A2, entonces y es Al

Si XI es A2 y X2 es Al, entonces y es A2

y función de pertenencia es:

$$\mu_{A1}(u) = \begin{cases} 1 - |u|, & \text{si } -1 \leq u \leq 1; \\ 0, & \text{en cualquier otro caso} \end{cases} \tag{9.1}$$

$$\mu_{A2}(u) = \begin{cases} 1 - |u-1|, & \text{si } 0 \leq u \leq 2; \\ 0, & \text{en cualquier otro caso} \end{cases} \tag{9.2}$$

a)sean las entradas (x1*,x2*)=(0.3,0.6)

b)borrosificador singleton

c)salida y^*= a incógnita

d)utilizando:

1) mecanismo de inferencia de producto y desborrosificador de centro medio

2) mecanismo de inferencia de producto y desborrosificador de centro de gravedad

3) mecanismo de inferencia de Lukasiewicz y desborrosificador de centro medio

1. $\mu_{B'}(y) = max\left[\mu_{A1}(0.3)\mu_{A2}(0.6)\mu_{A1}(y), \mu_{A2}(0.3)\mu_{A1}(0.6)\mu_{A2}(y)\right]$

   $= max\left[0.7 * 0.6 * \mu_{A1}(y), 0.3 * 0.4 * \mu_{A2}(y)\right]$

   $= max\left[0.42\mu_{A1}(y), 0.12\mu_{A2}(y)\right]$

   $y^* = \frac{0.12}{0.42+0.12} = 0.2222$

2. $\int_v (y)dy = 0.42 + 0.12 - \frac{1}{2}\frac{0.42*0.12}{0.12+0.42} = 0.4933$

   $\int_v (y)dy = -\frac{1}{6}0.42 + 0.12 +$

   $\frac{1}{6}\frac{(0.42)^3}{(0.42+0.12)} = 0.0923$

   $y^* = \frac{0.0923}{0.4933} = 0.1871$

3. $\mathrm{y}^* = \frac{0*1+1*1}{1+1} = 0.5$

9.2.1 Ejemplo

El siguiente ejemplo representa un algoritmo borroso que controla la velocidad de un ventilador. La entrada al sistema es el valor de la temperatura y la salida es un valor de la velocidad del ventilador. Suponiendo que el sistema se presenta como sigue:

si la temperatura es fría entonces velocidad de ventilador es baja

si la temperatura es moderada entonces velocidad de ventilador es media

si la temperatura es caliente entonces velocidad de ventilador es elevada.

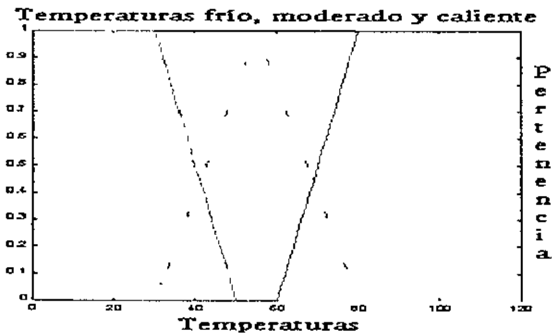
Este sistema tiene tres reglas borrosas donde los antecedentes (fría, moderada, caliente) y consecuentes (baja, media, elevada) se definen por los siguientes conjuntos borrosos. La siguiente codificación se realiza en el lenguaje de programación MATLAB ver. 5.

```
x=[0:1:120]; Temperatura
y=[0:1:10]; velocidad
cool_mf=trapzoid(x,[0 0 30 50]);
moderate_mf=triangle(x,[30 55 80]);
hot_mf=trapzoid(x,[60 80 120 120]);
antecedent_mf=[cool_mf;moderate_mf;hot_mf];
```

plot(x, antecedent_mf);

xlabel('Temperaturas');

ylabel('Pertenencia');



low_mf = trapzoid(y,[0 0 2 5]);

medium_mf = trapzoid(y,[2 4 6 8]);

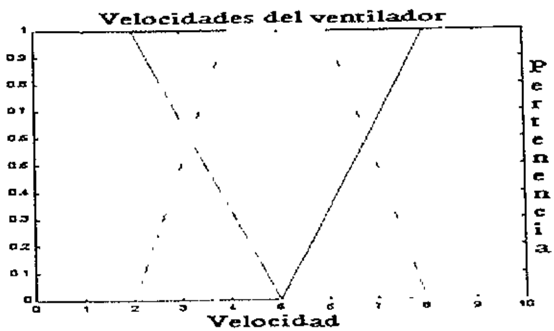
high_mf = trapzoid(y,[5 8 1 0 1 0]);

consequent_mf = [low_mf;medium_mf;high_mf];

plot (y, consequent_mf) title (' Baja, media y elevada velocidad');

xlabel('Velocidad del ventilador');

ylabel('Pertenencia');



Ahora que tenemos las funciones de pertenencia definidas, podemos seguir los siguientes pasos propuestos [2].

1. Borrosificar las entradas

2. Aplicar un operador borroso

3. Aplicar una implicación borrosa

4. Agregación de salidas

5. Desborrosificar

Primero borrosificamos la entrada. La salida de el primer paso es el grado de cumplimiento (fulfillment) de cada regla. En este paso no hay operadores.

```
temp=72;
dof1=cool_mf(find(x==temp));
x=[0:1:120];
dof1=cool_mf(find(x==temp));
cool_mf=trapzoid(x,[0 0 30 50]);
dof1=cool_mf(find(x==temp));
moderate_mf=triangle(x,[30 55 80]);
dof2=moderate_mf(find(x==temp));
hot_mf=trapzoid(x,[60 80 120 120]);
dof3=hot_mf(find(x==temp));
DOF=[dof1;dof2;dof3]
DOF =
0
```

0.3200

0.6000

Utilizando la implicación del producto de Larson. (μ_A(x) . μ_B(y))

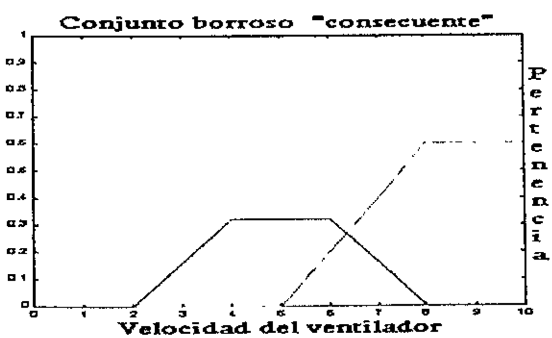
consequent1=product(low_mf,dof1);

consequent3=product(medium_mf,dof2);

consequent2=product(high_mf,dof3);

plot(y,[consequent1;consequent2;consequent3])

axis([0 10 0 1.0])

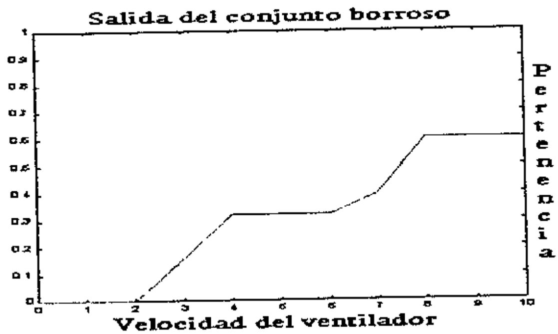


Ahora agregaremos los conjuntos borrosos consecuentes. Usaremos el operador max.

Output_mf=max([consequent1;consequent2;consequent3]);

plot(y, Output_mf)

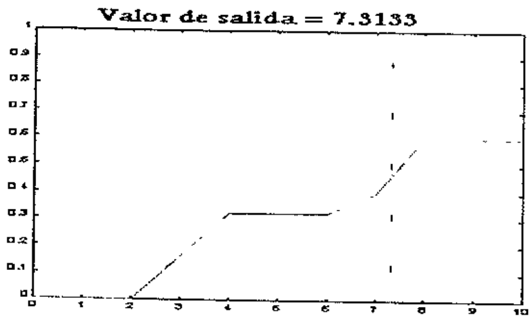
axis([0 10 0 1])



Para finalizar se presenta la gráfica de desborrosificación con el valor de salida.

output=centroid(y,Output_mf);

c_plot(y,Output_mf,output,'Crisp Output');



Bibliografía

[1] Chen Y., Lin K. and Hsu S.(1992), *A Self Learning FUZZY Controller*, IEEE International Conference on Fuzzy Systems, páginas 189-196.

[2] J. Wesley Hines (1997), *Matlab Supplement to Fuzzy and Neural Approaches in Engineering*, A Wiley-Interscience Publication, New York 10158-0012, ISBN 0-471-19247-3, páginas 37-43.

[3] Li-xin Wang (1997), *A Course in Fuzzy Systems and Control*, Prentice Hall PTR, Upper Saddle River, NJ 07458, ISBN 013540882-2.

[4] Michael Doob (1984), *A Gentle Introduction to TEX. A Manual for self-study*, Department for Mathematics, The University of Manitoba, Winnipeg, Manitoba, Canada R3T2N2.

[5] Grätzer George A. (1996), *Math into LATEX, An introduction to LATEX and AMS-LATEX*, BIRKHÄUSER, ISBN 0-81763805-9.

[6] Janette Cardoso, Heloisa Camargo (1999), *Fuzziness in Petri Nets*, Physic-Verlag, ISBN 3-7908-1158-0.

[7] Abraham Kandel (1986), *Fuzzy Mathematical Techniques with Applications*, Addison-Wesley, ISBN 0-201-11752-5.

[8] Arnold Kaufmann, Madan M. Gupta (1988), *Fuzzy Mathematical Models* in engineering and Management Science, North-Holland, ISBN 0 444 705015.

[9] Bart Kosko (1997), *Fuzzy engineering*, Prentice Hall, ISBN 0-13-124991-6.

[10] Witold Pedrycz (1998), *Computational Intelligence (An introduction)*, CRC Press, ISBN 0-8493-2643-5.

[11] Witold Pedrycz (1993), *Fuzzy Control and Fuzzy Systems*, Research studies Press LTD, second extended edition, ISBN 0-471-93475-5.

Capítulo 10

CONCLUSIONES

Los controladores borrosos ofrecen una metodología bien definida para los problemas de control lineales como no lineales. Con lo expuesto en la presente tesis no cubre por completo toda la teoría de control borroso, pero si se presenta un buen avance en las bases de ésta, tanto herramientas como procedimientos que puedan guiar a cualquiera que explore en esta área de control.

Los japoneses están desarrollando esta teoría y han logrado implementarla en mecanismos útiles para la vida diaria. El desarrollo de esta teoría no es, a pesar de ser un procedimiento metodológico, una receta de cocina que se pueda seguir al pie de la letra para su implantación y buen funcionamiento.

Debido a que el conocimiento que se transmite en estos sistemas es de tipo heurístico, también se tiene que usar la teoría de control borroso como un sistema de aproximación que trate de simular todo tipo de comportamiento de la mejor manera. Sí existe un modelo borroso exacto que pueda aproximar cualquier sistema, pero, para llegar a dicho modelo no será tarea fácil, pues se tiene que conocer en cierta medida parte de tal comportamiento (entradas y salidas del sistema) para que a partir de ellas se pueda empezar a desarrollar un sistema borroso que se aproxime más a la realidad.

Parecerá que caemos en el mismo problema de la teoría de control moderna, en donde se necesitan de modelos matemáticos "exactos" para diseñar modelos de control, pero no en caso de los controladores borrosos, pues estos no necesitan valores exactos como entradas y sus rangos de respuesta van más allá de la pertenencia o no pertenencia, facilitando mucho el modelaje del controlador.

Se necesitará mucha práctica y conocimiento documentado para reconocer cuando es mejor utilizar un tipo de borrosificador, un mecanismo de inferencia con sus múltiples posibilidades de composiciones, tipos de normas, etc, y mecanismos desborrosificadores.

Los controladores borrosos pueden aceptar particiones múltiples como variables de entrada y proporcionar salidas de control para cualquier sistema.

En este trabajo se concluyó, a partir de información documentada, que se cuenta con una nueva teoría de control que puede ser, y está siendo muy explotada a niveles internacionales. Por lo tanto es importante que se empiece a desarrollar esta teoría de control en las aulas universitarias.

Es necesario mencionar que la teoría de control borroso ya se está empezando a estudiar en nuestro país en los centros de investigación superior como la UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO a nivel maestría y en el CENTRO DE INVESTIGACIONES AVANZADAS DE EL INSTITUTO POLITÉCNICO NACIONAL.