

Algoritmos genéticos en la construcción de funciones de pertenencia borrosas

Pedro Y. Piñero, Leticia Arco, María M. García, Liesner Acevedo

Universidad Central “Marta Abreu” de Las Villas
Carretera a Camajuaní Km 7 1/2,
Villa Clara, CUBA, 54830
{ppp,frodo}@uci.cu , leticiaa@cei.uclv.edu.cu

Resumen

La teoría de los conjuntos borrosos ha adquirido singular importancia en los últimos tiempos dada su aplicación en numerosas ramas de las Ciencias de la Vida. En la teoría de los conjuntos borrosos y su aplicación en diferentes áreas, juega un papel fundamental la determinación de las funciones de pertenencia adecuadas. En esta investigación se hace un análisis del estado del arte de la construcción de diferentes tipos de funciones de pertenencia. Se expone el uso de los algoritmos genéticos en la construcción de las funciones de pertenencia de conjuntos borrosos. Como caso particular se analiza el uso de los Algoritmos Genéticos en la obtención de funciones de pertenencia campana Beta y Triángulo aunque se muestra en general las potencialidades de los AG en la obtención de funciones de pertenencia tanto en datos numéricos como simbólicos. Se discuten las potencialidades y desventajas de diversos métodos para la construcción de funciones de pertenencia, reportados en la bibliografía. Se comparan los algoritmos genéticos con el método propuesto por Hong para la obtención de funciones de pertenencia, por medio de la comparación de los resultados de aplicar ambos métodos en la clasificación de bases de casos internacionales.

Palabras clave: Algoritmos Genéticos, Funciones de pertenencia, Lógica borrosa, Conjuntos Borrosos.

1. Introducción

Los Algoritmos Genéticos (AG) surgen como herramientas para la solución de problemas complejos de búsqueda y optimización, producto del análisis de los sistemas adaptativos en la naturaleza, y como resultado de abstraer la esencia de su funcionamiento. El término Algoritmo Genético se usa por el hecho de que estos simulan los procesos de la evolución darwiniana a través del uso de operadores genéticos que operan sobre una población, de individuos o cromosomas, que “evoluciona” de una generación a otra [Holland75].

Los AG han ganado popularidad en los últimos años debido a la posibilidad de aplicarlos en una gran

gama de campos y a las pocas exigencias que imponen al problema a resolver. Han mostrado además bondades en la resolución de problemas multiobjetivos siendo los modelos de pareto los más conocidos.

Por otra parte numerosos problemas a los que nos enfrentamos están descritos por variables borrosas, variables que describen situaciones ambiguas. Esta clase de problemas deben ser resueltos utilizando la teoría de la lógica borrosa.

En este trabajo se propone el uso de los AG para la construcción de conjuntos borrosos y sus respectivos términos lingüísticos elementos básicos de la lógica borrosa. Se presenta el uso de los AG en

la determinación de los parámetros de las funciones de pertenencia, como un módulo necesario en un sistema de aprendizaje automatizado, que genera reglas de inferencia borrosas a partir de un conjunto de entrenamiento.

En la sección 2, se discuten brevemente las principales estrategias reportadas en la bibliografía para la construcción de funciones de pertenencia y los modelos matemáticos más usados en la representación de dichas funciones. En la sección 3 se presentan los elementos fundamentales que componen un algoritmo genético. En la sección 4 se discute el diseño de un algoritmo genético para la construcción automatizada de funciones de pertenencia. En la sección 5, se muestran los resultados experimentales de aplicar el algoritmo genético diseñado en la sección anterior en el aprendizaje automatizado. Se utiliza la base de casos de tiroides suministrada por el Garvan Institute of Medical Research, Sydney y la base de casos del corazón suministrada por European Statlog project, Dept. Statistics and Modelling Science, Strathclyde University in 1993. Ambas bases de datos aparecen publicadas en UCI Repository of Machine Learning databases, University of California Irvine [Blake98]. Finalmente en la sección 6 se discuten las conclusiones del trabajo.

2. Construcción de funciones de pertenencia de conjuntos borrosos.

Un conjunto borroso está determinado por funciones de pertenencia asociadas a él [Cox98]. Si X es una colección de objetos denotados genéricamente por x , entonces un conjunto borroso A en X se define como un conjunto de pares ordenados:

$$A = \{(x, \varphi_A(x)) | x \in X\}$$

Donde $\varphi_A(x)$ es llamada la función de pertenencia para el conjunto A . La función de pertenencia asigna a cada elemento de X un grado de pertenencia en el intervalo $[0,1]$. A X se llama universo de discurso y puede ser un espacio discreto o continuo.

2.1 Estrategias para la construcción de funciones de pertenencia.

Diversas estrategias han sido aplicadas para construir funciones de pertenencia de conjuntos borrosos. No obstante estas pueden ser agrupadas en los siguientes grupos [Baldwin98]:

- Evaluación subjetiva y construcción a partir de expertos: en esta estrategia los

conjuntos difusos pueden determinarse a partir del conocimiento de expertos en el dominio de aplicación. Estos simplemente dibujan o especifican diferentes curvas de pertenencia de las cuales eligen una. En algunos casos, la elección puede estar determinada mediante métodos que tienen su base en la Psicología, Inteligencia Artificial Distribuida y Sistemas Multiagente.

- *Frecuencias convertidas o probabilidades:* algunas veces, la información tomada a partir de histogramas de frecuencia u otras curvas de probabilidad se emplea como base para construir la función de pertenencia. Existe una gran variedad de métodos de conversión posibles y cada uno posee sus propias fortalezas y debilidades, tanto matemáticas como metodológicas. Sin embargo, es necesario recordar que las funciones de pertenencia no son funciones de densidad probabilística.
- *Medición física:* muchas aplicaciones de lógica borrosa usan la medición física, pero casi ninguna mide el grado de pertenencia directamente. En su lugar, la función de pertenencia se obtiene mediante otro método y los grados de pertenencia individuales se calculan a partir de ella.
- *Aprendizaje y adaptación:* la aplicación de las técnicas de aprendizaje automatizado posibilitan construir de forma automática a partir de datos numéricos y simbólicos las funciones de pertenencia, generalmente las funciones de pertenencia construida por esta vía son representadas como modelos matemáticos lineales.

Ejemplos que muestran algunas de las estrategias anteriores se citan a continuación:

Método de interpolación por el polinomio de Bernstein [Chen95] donde a partir de pares (valor, pertenencia) suministrados por expertos se obtiene la función de pertenencia asociada.

Mediante la determinación de los centros de gravedad de las variables lingüísticas, punto en que las funciones de pertenencia asociadas a las variables lingüísticas alcanzan su máximo valor [Narazaki94].

A partir de casos de entrenamiento, suministrados por expertos, utilizando algoritmos de aprendizaje automatizado como el descrito en [Hong98].

2.2 Diferentes representaciones de funciones de pertenencia

Existen numerosas formas de representar las funciones de pertenencia; existiendo una estrecha relación entre la forma de representación, los distintos enfoques que se siguen en la construcción de las funciones de pertenencia y la naturaleza del problema que ellas representan. La función de pertenencia más simple es la construida por un experto en forma de pares como se muestra:

(*Elemento*, *Valor de pertenencia*), donde *Elemento* es un posible valor del dominio y el *Valor de pertenencia*, indica el grado de pertenencia del elemento a determinado conjunto.

Entre las representaciones de funciones de pertenencia basadas en modelos matemáticos lineales se encuentran las curvas S, las campanas, los triángulos y los trapecios entre otras. En ocasiones, la aplicación de los modelos matemáticos lineales no es recomendada en problemas en los que intervengan casos que sean descritos por atributos simbólicos, por ejemplo: el color. Por tanto se sugiere el uso de modelos basados en la teoría de la medida.

2.2.1 Modelo lineal campana Beta.

En general, existen tres clases importantes de curvas “campana” – las campanas PI, las campanas Beta y las campanas gaussianas. La diferencia entre los tres tipos de curvas está dada por la pendiente de la curva así como por los valores de los puntos finales de la curva [Nauck00].

Explicamos a continuación la curva campana Beta por ser utilizada en la sección 6 en un ejemplo resuelto. La curva Beta [Cox98] es una curva definida por dos parámetros, el punto central del dominio (γ) y un valor que indica la mitad del ancho de la curva en el punto de inflexión (β). Véase a continuación la ecuación y la gráfica:

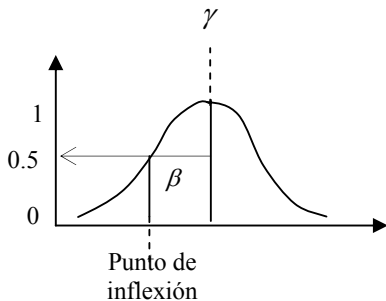


Figura 1 Esquema de la curva beta

$$B(x; \gamma, \beta) = \frac{1}{1 + \left(\frac{x - \gamma}{\beta}\right)^2}$$

Esta curva es asintótica respecto al eje de las abscisas.

2.2.2 Modelos lineales basados en triángulos y trapecios.

Las funciones de pertenencia triangulares pueden o no ser simétricas y están determinadas por tres parámetros como se muestra a continuación:

$$\text{Triángulo}(x, a, b, c) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & c \leq x \end{cases}$$

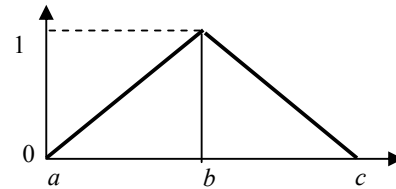


Figura 2 Esquema de función triángulo

Las aristas del triángulo indican un descenso lineal del grado de pertenencia a partir del punto b y hasta que el grado de pertenencia llega a ser cero.

Las funciones de pertenencia trapezoidales, por su parte pueden verse como triángulos truncados donde se forma una meseta. Los elementos de la meseta se corresponden con los elementos del universo donde se alcanza el máximo grado de pertenencia.

3. Componentes principales de un algoritmo genético.

Para dar solución a un problema utilizando AG se deben seguir los siguientes pasos:

- Definir la estructura de los cromosomas o individuos y la representación de los mismos (cada cromosoma es una posible solución potencial para nuestro problema). La estructura del cromosoma depende también del tipo de representación binaria o real de los genes.
- Determinar el criterio de evaluación o adaptación de los individuos. La función de evaluación juega un papel importante en la clasificación potencial de las soluciones en

términos de sus características; es el criterio de optimización y evaluación de la calidad de los individuos.

- Determinar los operadores genéticos a utilizar y su forma de aplicación.
- Definir el criterio de parada del algoritmo.
- Definir un criterio de reemplazo entre generaciones consecutivas y el tamaño promedio de las poblaciones.

3.1 Definición del cromosoma y representación

Se debe utilizar la estructura de datos más apropiada para el problema en cuestión. Si usted está optimizando un problema en que las variables sean reales entonces el cromosoma debe estar formado por componentes reales. Si una solución a su problema puede representarse con algunos números imaginarios y algún entero entonces es necesario definir un cromosoma con estas características.

Definir una representación apropiada es parte del arte de usar algoritmos genéticos. Se debe usar una representación mínima que sea completamente expresiva de forma tal que sea capaz de representar cualquier solución al problema, pero además que no admita soluciones no factibles. En caso de que el diseño de los cromosomas permita representar soluciones no factibles entonces corresponde diseñar la función objetivo de forma tal que esta solo de crédito parcial a las mismas.

La representación no debe contener información innecesaria para dar la solución al problema. Aunque puede haber mérito usando una representación que contiene material genético extra, esto tiende a aumentar el tamaño del espacio de la búsqueda y disminuir así la eficiencia del Algoritmo Genético durante la búsqueda.

El número de posibles representaciones es interminable. Usted puede escoger una representación completamente numérica como una serie de números reales que podrían mostrarse como números reales, o al estilo de [Holland75] y [Goldberg89], (como una cadena de bits que mapean a números reales).

El problema puede depender de una secuencia de sucesos, en este caso una representación basada en el orden (lista o serie) podría ser apropiada. En muchos de estos casos, se deben escoger a operadores que mantengan la integridad de la secuencia; los cruzamientos podrían generar reordenamientos de la secuencia teniendo en cuenta que no haya elementos repetidos. Otros problemas

se prestan para ser representados según la estructura de un árbol, se podría desear aquí representar las soluciones explícitamente como árboles y aplicar directamente sobre estos los operadores.

Alternativamente, muchas personas organizan árboles sobre arreglos o cadenas parseables y aplican los operadores a estas cadenas en lugar de a los árboles.

Algunos problemas incluyen una mezcla de elementos continuos y discretos, caso en que se puede necesitar crear una nueva estructura para representar la mezcla de información. En estos casos se deben definir operadores genéticos que respeten la estructura de la solución, por ejemplo, en la búsqueda de una solución en la que intervengan reales y enteros podrían usarse operadores de cruzamiento que crucen partes enteras con partes enteras y partes flotantes con partes flotantes, pero que nunca mezclen partes enteras con flotantes. Para cualquier representación que se tome, se debe estar seguro de escoger operadores apropiados para ella.

Como último aspecto de esta subsección definiremos población como una colección de cromosomas o individuos.

3.2 Operadores genéticos

Entre los operadores genéticos más utilizados en la resolución de problemas utilizando AG se señalan: [Golberg89]

La *selección* que aporta mayor potencia y robustez a la técnica de búsqueda, este operador cumple la función de hacer una selección de los mejores individuos de manera que estos sean considerados en el proceso de generación de la nueva población. Existen varios mecanismos para su ejecución como por ejemplo el método de la ruleta, piscina de selección, etc.

El *operador de cruce* que permite el cambio de código genético entre los individuos, introduciendo así, instancias de nuevas combinaciones de genes. Este facilita al AG una búsqueda con eficiencia en espacios de varias dimensiones y su probabilidad de ocurrencia debe ser alta para lograr una elevada introducción de nuevas soluciones del problema en cuestión.

El *operador de mutación* que provoca la posible aparición de nuevas características en el individuo al mutar uno de sus genes, posibilitando así, la búsqueda en nuevas

áreas del espacio solución del problema. La probabilidad de que este proceso ocurra debe ser muy baja asegurando de esta forma que no se convierta en sobreactivo destruyendo las ventajas del cruzamiento.

3.3 Función objetivo o evaluación

La función evaluación juega un papel importante en la clasificación potencial de las soluciones en términos de sus características; es el criterio de optimización y evaluación de la calidad de los individuos. Se desarrolla a partir de los valores del fenotipo.

Por ejemplo supongamos que tenemos los cromosomas v_1 y v_2 :

$v_1(1000101110110101000111)$,

$v_2(000000111000000010000)$

y que estos representan a los valores $x_1 = 0.637179$ y $x_2 = -0.958973$ respectivamente. Si evaluamos a los cromosomas según determinada función $f(v)$ y obtenemos como resultado:

$f(v_1) = 1.586345$ y $f(v_2) = 0.078878$ entonces, si estamos maximizando, podemos concluir que el cromosoma v_1 es el mejor de los dos.

Es importante también notar la distinción entre las fitness y el valor de la función objetivo. El valor del fitness es el valor que utiliza el algoritmo genético para clasificar la aptitud de los individuos ante el cruzamiento. El AG usa generalmente el valor del fitness no el de la función de evaluación durante el proceso de selección.

3.3.1 Evaluación multiobjetivo

En los problemas prácticos, a menudo queremos optimizar más de un objetivo. Estos pueden contradecirse entre sí y puede ser poco satisfactorio combinarlos en un solo objetivo o reducirlos de alguna manera para optimizar solo un objetivo cada vez.

Los métodos clásicos asignan pesos a cada uno de los objetivos según la importancia de lo mismos, pero este método es muy subjetivo. Este método puede simplificar el comportamiento de los objetivos, y es a menudo difícil encontrar pesos que reflejen con precisión la vida real.

Usando los Algoritmos Genéticos, podemos encontrar un conjunto de soluciones tal que ninguna solución sea peor que otra. Un experto en el problema entonces seleccionará la solución que más se adapte a sus necesidades.

Considérese un problema de optimización multiobjetivo con k objetivos $f_i : i = 1, 2, \dots, k$ y n variables decisión $x_i : i = 1, 2, \dots, k$:

$$f(x) = (f_1(x); \dots; f_k(x))$$

El objetivo de la optimización es minimizar $f_i : i = 1, 2, \dots, k$ sujeto a las restricciones:

$$g_i(x) \leq 0; i = 1, \dots, m$$

Debido a que los k objetivos pueden estar en contradicción unos con otros, es normalmente difícil de obtener el mínimo global para cada objetivo de forma simultánea. El objetivo de la Optimización Multiobjetivo (MOO) es lograr un conjunto de soluciones que sean Pareto óptimas [Jaochu01]. Los conceptos relacionados con este tema se definen como sigue:

La dominación de Pareto: Un vector $u = (u_1, u_2, \dots, u_k)$ se dice que domina¹ a otro vector $v = (v_1, v_2, \dots, v_k)$ si y sólo si $u_i \leq v_i \quad \forall i \in [1..k]$ y existe al menos un i tal que $u_i < v_i$.

Pareto óptimo: Una solución x que se dice que es Pareto óptima si y sólo si no existe otra solución x' tal que $f(x)$ es dominada por $f(x')$. El conjunto que agrupa todas las soluciones que son Pareto óptimas para un problema de optimización multiobjetivo dado, se conoce como conjunto Pareto óptimo y se denota (p^*).

Frente Pareto: Dado un problema de optimización multiobjetivo y su respectivo conjunto Pareto óptimo (p^*), se denomina frente Pareto (pf^*) al conjunto que se define como:

$$pf^* = \{f(x) = (f_1(x), f_2(x), \dots, f_k(x))\}$$

Existen numerosas estrategias de optimización multiobjetivo usando AG, las basadas en Pareto y las no basadas en Pareto. De forma general podemos citar:

- Optimización basada en la evaluación de vectores: donde en cada proceso de selección los individuos son separados en subpoblaciones y desde cada subpoblación se seleccionan individuos por un criterio de optimización diferente.
- Agregación Pesada convencional: es un

¹ Dominación se refiere a un proceso de minimización.

acercamiento simple a la optimización multiobjetivo. En este método, los objetivos diferentes se resumen a un solo escalar con un peso prescrito.

- La Agregación con Pesos Dinámicos [Jaochu01]: en ella los pesos se cambian gradualmente. Este último modelo es que utilizamos en nuestro diseño.

3.4 Criterios de parada, mecanismos de reemplazo y población

Los criterios de parada del algoritmo genético están muy relacionados con la naturaleza del problema en cuestión. No obstante podemos citar algunos de los criterios de parada más generales:

- Hasta número de generaciones determinado.
- Hasta que el mejor individuo de una población, alcance un valor predeterminado de la función de evaluación.
- Hasta que determinada cantidad de individuos de una población sobrepase un valor de umbral predeterminado.
- Hasta que la mayoría de los individuos de una población se encuentren en la vecindad de un punto. O sea que la mayoría de los individuos haya convergido a un punto.

Los mecanismos de reemplazo por su parte están muy relacionados con el tipo de algoritmo genético que se implemente siendo los más populares:

- Simple, descrito en [Goldberg89], este algoritmo no realiza reemplazo y tiene elitismo opcional. En cada generación el algoritmo crea una nueva población de individuos a partir de la cual se repite el proceso. La vieja población es desechada en cada generación.
- Con poblaciones de tamaño fijo, la nueva población se forma a partir de la población vieja y sus descendientes. Para conformar la nueva población el usuario debe especificar el porcentaje de individuos viejos a reemplazar en cada generación por los nuevos individuos.
- Incremental en el que en cada generación solo uno o dos descendientes sobreviven.
- Algoritmo genético con migraciones: este algoritmo permite la evolución de varias poblaciones de forma paralela, donde cada una de ellas evoluciona siguiendo un modelo de algoritmo genético con tamaño fijo. En cada generación el algoritmo establece una política de emigración entre las diferentes poblaciones.

4. Diseño de un algoritmo genético para la construcción de funciones de pertenencia

En esta sección se describe el diseño de un algoritmo genético para la construcción de funciones de pertenencia de conjuntos borrosos.

Antes de definir las componentes del algoritmo genético asociado debemos señalar algunas propiedades que deben cumplir las funciones de pertenencia.

Cada variable del problema en cuestión es una variable lingüística formada por un conjunto de términos lingüísticos, donde a cada uno de estos está asociada una función de pertenencia y en general cumplen la siguiente propiedad:

Propiedad (1):

Las curvas adyacentes de una misma variable lingüística se solapan.

El grado de solapamiento entre curvas adyacentes es un número real que indica el porcentaje del área bajo la curva que es común a estas curvas.

El proceso de construcción de las funciones de pertenencia consta de dos partes: la construcción de las funciones de pertenencia iniciales y el ajuste de estas funciones.

Para determinar las funciones de pertenencia iniciales el conjunto de entrenamiento se puede someter a un proceso de discretización o de clusterización, en dependencia de la naturaleza de los datos y de la información brindada por los expertos. Una vez determinados los conjuntos numéricos correspondientes a cada término lingüístico entonces se construyen las funciones tomando como punto de mayor valor de la función el punto medio del intervalo correspondiente a cada término lingüístico.

Una vez determinadas las funciones de pertenencia iniciales y todos sus parámetros nuestro problema se reduce al reajuste de los parámetros de funciones adyacentes con el objetivo de que nuestras funciones cumplan las restricciones de solapamiento planteadas anteriormente. En el proceso de ajuste el punto de mayor valor de pertenencia de cada función inicial no es modificado. Todas las modificaciones establecidas en este punto se rigen por las siguientes propiedades:

En el caso de las funciones campana Beta:

Propiedad (2a):

$$\frac{\beta_{f_0}}{\beta_{g_0}} = \frac{\beta_f}{\beta_g} \text{ donde } \beta_{f_0} \text{ y } \beta_{g_0} \text{ indican los valores}$$

Beta de las funciones iniciales f_0 y g_0 respectivamente, mientras que β_f y β_g indican los valores Beta de las funciones reajustadas f y g respectivamente.

La figura 3 muestra la forma final de dos Campanas Beta adyacentes y pertenecientes a la misma variable lingüística luego del proceso de ajuste.

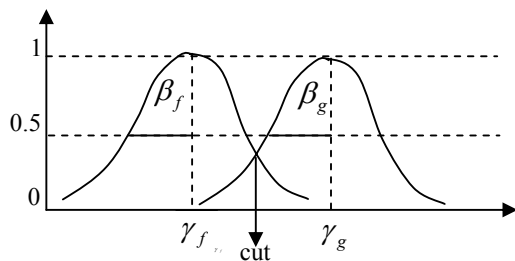


Figura 3 Solapamiento de las Campana Beta

En el caso de las funciones triangulares:
Propiedad (2b):

$$\frac{c_{f_0} - b_f}{c_{g_0} - b_g} = \frac{c_f - b_g}{c_g - b_g} \text{ donde } b_f \text{ y } b_g \text{ son los puntos}$$

donde las funciones f y g alcanzan el mayor grado de pertenencia respectivamente. Estos puntos son determinados durante la construcción de las funciones de pertenencia iniciales y no se modifican en el proceso de reajuste. Los parámetros c_f y c_g representan los puntos de f y g respectivamente, donde el grado de pertenencia de estas funciones alcanza el valor de cero; c_{f_0} y c_{g_0} son los puntos extremos de f_0 y g_0 .

La figura 4 muestra la forma final de dos funciones triángulo adyacentes y pertenecientes a la misma variable lingüística luego del proceso de ajuste.

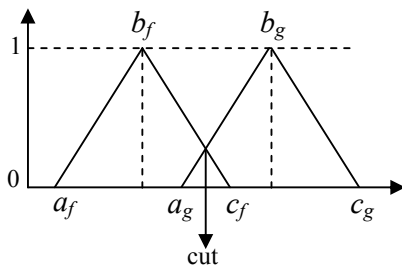


Figura 4 Solapamiento de las funciones Triángulo

La variable cut representa en las figuras 3 y 4 el punto de intersección entre las funciones

adyacentes. Los puntos medios de las curvas no se modifican durante el proceso de ajuste de los parámetros.

Al verificar que se cumplan estas relaciones se evita que en el proceso de reajuste las curvas se deformen desproporcionalmente y que se destruyan las potencialidades del proceso de discretización o clusterización.

La construcción de todas las funciones de pertenencia correspondientes a una misma variable lingüística depende de la determinación de los parámetros de las dos primeras funciones de pertenencia adyacentes. Las restantes funciones que parten de éstas se obtienen por medio de simples análisis matemáticos tomando como base el cumplimiento de las propiedades (1), (2a) y (2b).

4.1. Definición del cromosoma.

Los puntos medios de las funciones iniciales se mantienen constantes en las funciones reajustadas, por tanto intervienen solo como constantes en la resolución del problema.

En ambos casos los cromosomas estarán formados por parámetros que describen a las funciones. Como la naturaleza de estos parámetros es real, entonces la representación del cromosoma adoptada es real.

Otro aspecto tomado en consideración para decidir el tipo de representación es el hecho de que se plantea que en la práctica la representación real arroja mejores resultados que la binaria [Michalewics92].

En el caso del modelo lineal campana Beta, las soluciones están condicionadas sólo por los valores de los parámetros $Beta_f$ y $Beta_g$ correspondientes a las dos primeras funciones adyacentes de una misma variable lingüística, de ahí que nuestro cromosoma tenga la siguiente estructura:

$$\boxed{\beta_f \quad \beta_g}$$

En el caso del modelo lineal triángulo, las soluciones finales están condicionadas por los parámetros c_f , a_g y cut . Donde c_f es el máximo valor a partir del cual la función de pertenencia f devuelve el valor de cero, a_g es el mínimo valor a partir del cual la función g devuelve el valor de cero y cut es el punto donde se cortan las funciones f y g . Luego la estructura del cromosoma es:

a_g	c_f	cut
-------	-------	-------

4.2 Operadores genéticos utilizados.

Tanto en la obtención de las funciones de pertenencia triangulares como de las campanas beta aplicamos los operadores genéticos clásicos reportados en la bibliografía y comentados en la sección anterior.

Como operador de selección se utilizó la ruleta y se aplicó un escalado basado en la media y la varianza. La selección de los individuos se hace teniendo en cuenta el valor de sus *fitness*:

$rfitness = a * fitness + b$ siendo *fitness* el resultado de evaluar al individuo o cromosoma en la función de evaluación. Los parámetros *a* y *b* son obtenidos a partir de estadígrafos poblacionales como la media y la varianza. En [Golberg89] se muestran las potencialidades del mecanismo de selección aplicado.

Como operador de cruzamiento se adoptó el cruzamiento en un punto con probabilidad 0.9 y como operador de mutación aplicamos una mutación simple en un punto con probabilidad 0.03.

En la selección de los parámetros para los operadores genéticos se utilizó el sistema SPSS versión 8.0. Se realizaron 20 corridas del algoritmo con cada una de las diferentes combinaciones de parámetros para los operadores de cruzamiento y mutación; se realizaron pruebas de comparación de medias y finalmente se determinó que la combinación aplicada en nuestro algoritmo arrojaba resultados significativamente mejores que otras combinaciones donde se utilizaban operadores de cruzamiento y mutación en dos puntos [Piñero00].

4.3. Función de evaluación

La función de evaluación debe garantizar cumplir las propiedades (1) y (2a) para el caso de las funciones beta, mientras que para el caso de las funciones triangulares la función de evaluación debe cumplir las propiedades (1) y (2b). Como se muestra en cualquiera de los dos casos estamos en presencia de un problema multiobjetivo con dos objetivos bien definidos. En este trabajo se propone como estrategia multiobjetivo a utilizar la Agregación con Pesos Dinámicos cuya principal característica es la forma dinámica de los pesos durante la optimización y el mantenimiento de un conjunto de soluciones pareto en un fichero que asegura no se pierdan soluciones buenas durante la búsqueda.

En el caso de las funciones Beta esta función se determina como:

$$w_1 P_1 + w_2 P_{2a}$$

donde:

$$P_1 = \frac{P}{100} \int f(x, B_f, \gamma_f) - \int_{\alpha_g}^{cut} f(x, B_f, \gamma_f) - \int_{cut}^{b_f} g(x, B_g, \gamma_g)$$

$$P_{2a} = \frac{B_g}{B_f} - \frac{B_{g_0}}{B_{f_0}}$$

p es el porcentaje de solapamiento indicado por el usuario y *cut* es el valor en el eje de las abscisas para el cual se interceptan las curvas.

Los pesos w_1 y w_2 basándonos en la dominación de pareto varían uniforme y dinámicamente durante el proceso evolutivo en un rango de 0 a 1, las siguientes fórmulas establecen el cálculo de los pesos w_1 y w_2 : muestran como se muestra:

$$w_1(t) = \text{sen}(2\pi t / k)$$

$$w_2(t) = 1 - w_1(t)$$

donde *t* indica el número de la generación actual y *k* regula la frecuencia de repetición de los pesos durante el proceso evolutivo.

Los valores de γ_f y γ_g , en el caso de las funciones campanas beta y los valores de b_f y b_g correspondientes a las funciones triangulares, no sufren modificaciones durante el reajuste de las curvas, las curvas mantienen su simetría respecto a sus puntos centrales.

Teniendo en cuenta que la curva campana Beta es asintótica respecto al eje de las abscisas, los valores de a_g y b_f pueden ser determinados a partir de un valor especificado por el usuario y que indique un grado de pertenencia prácticamente nulo, como se muestra a continuación:

$$f(a_f) = f(c_f) = g(a_g) = g(c_g) = 0.001$$

la determinación del valor 0.001 como valor apropiado se presenta en [Piñero00].

La función de evaluación para el caso de las funciones triangulares se muestra a continuación:

$$w_1 P_1 + w_2 P_{2b}$$

donde:

$$P_1 = \frac{P}{50} (c_f - b_f) - f(cut)(c_f - a_g)$$

$$P_{2b} = \frac{c_{g_0} - b_g}{c_{f_0} - b_f} - \frac{c_g - b_g}{c_f - b_f}$$

y los pesos w_1 y w_2 varían de forma similar al caso anterior de las funciones beta.

4.4 Criterio de parada y mecanismo de reemplazo.

Como posibles criterios de parada se tomaron dos criterios:

- Número fijo de iteraciones o de generación de poblaciones. Se condicionó la parada a las 200 generaciones.
- El error en los cálculos de las funciones de pertenencia no sea mayor que un valor de umbral $E = 0.02$.

En el caso de las campanas beta que:

$$\frac{P}{100} \int f(x, B_f, \gamma_f) - \int_{\alpha_g}^{cut} f(x, B_f, \gamma_f) - \int_{cut}^{b_f} g(x, B_g, \gamma_g) \leq E$$

y en el caso de las funciones de pertenencia triangulares que:

$$\frac{P}{50} (c_f - b_f) - f(cut)(c_f - a_g) \leq E$$

Como mecanismo de reemplazo utilizamos la estrategia “Poblaciones de tamaño fijo”. La nueva población se forma a partir de la población vieja siguiendo una estrategia de reemplazo. El usuario puede especificar el porcentaje de individuos a reemplazar en cada generación aunque el sistema propone por defecto el reemplazo de del 50 % de la población.

Como último parámetro a tener en cuenta en la resolución del problema se fijó como tamaño de las poblaciones 200 individuos.

5. Experimentos y Resultados

Como se discutió anteriormente existen numerosas formas de construir automáticamente funciones de pertenencia asociadas a conjuntos borrosos. En esta sección se discuten las ventajas y desventajas de algunos de los métodos citados anteriormente. Se muestran los resultados de comparar las funciones de pertenencia obtenidas por diversos métodos por medio de su aplicación en la clasificación automatizada de bases de casos internacionales publicadas en UCI Repository of Machine Learning databases, University of California Irving.

El algoritmo expuesto en [Hong98] propone un método para derivar automáticamente reglas borrosas y funciones de pertenencia a partir de un conjunto dado de casos de entrenamiento. Como principales desventajas del mismo podemos citar:

- Requiere que los valores del rasgo objetivo sean ordenables. En muchos problemas reales, los rasgos son de naturaleza variada, y no son ordenables.
- No se permite el uso de datos simbólicos, solo numéricos, lo que restringe su aplicabilidad porque generalmente en problemas reales se presentan datos de los dos tipos.
- Trabajan con un solo tipo de función de pertenencia, la triangular. Los modelos borrosos son raramente sensibles a las descripciones de los conjuntos borrosos, tal insensibilidad los hace bastante robustos y elásticos. A pesar de esto, en problemas reales no todos los rasgos son descritos con un mismo tipo de función y esta posibilidad no es ofertada por el algoritmo. Para algunos rasgos las funciones triangulares no son apropiadas y el uso de este tipo de funciones en estos casos puede provocar que los valores de la función de pertenencia no sean los deseados. Además, en la modelación de sistemas dinámicos el uso de funciones triangulares puede aproximar su comportamiento escasamente a un grado de precisión. Generalmente, cuando hay que representar variables borrosas en finanzas, mercadeo, transportación, y otros modelos de negocio, la combinación de curvas con forma de campana y de curvas sigmoideas, para representar los puntos intermedios y extremos respectivamente, proveen una mejor forma para representar regiones borrosas individuales.
- El método propuesto por Hong, parte además de la construcción de un número elevado de funciones de pertenencia iniciales en dependencia de los datos suministrados en la base de aprendizaje. Esta característica puede provocar que en la mayoría de los problemas reales este algoritmo sea imposible de usar por su alto costo computacional.

Muchos de los métodos citados en la bibliografía tales como los citados en [Narazaki94], [Mann90] y [Chen95] no tienen en cuenta en la construcción de funciones de pertenencia los datos del tipo simbólicos. Sin embargo por medio de los algoritmos genéticos se pueden construir funciones de pertenencia para cualquier tipo de dato. Solo es necesario diseñar correctamente el cromosoma y la función a optimizar como se explicó anteriormente. Por tanto por su flexibilidad los algoritmos

genéticos son fácilmente adaptables para la resolución de este tipo de problemas.

5.1 Aplicación de funciones de pertenencia a bases de casos internacionales

En esta subsección se muestran los resultados de comparar dos métodos de obtención de funciones de pertenencia por medio de la clasificación de bases de casos internacionales usando las funciones de pertenencia obtenidas por dichos métodos.

Para la comparación respecto a los resultados de clasificación se implementó el sistema MLClassif versión 1.0, sistema de inferencia borroso del tipo Sugeno grado cero. Este sistema genera reglas de inferencia borrosas a partir de las funciones de pertenencia, obtenidas por medio de los métodos a comparar. Luego con las reglas generadas clasifica nuevos casos que le sean suministrados.

Se utilizan dos bases de casos publicadas en UCI Repository of Machine Learning databases, University of California [BLA98]. La base de casos de tiroides suministrada por el Garvan Institute of Medical Research, Sydney y la base de casos de enfermedades del corazón suministradas por el European Statlog project, Dept. Statistics and Modelling Science, Strathclyde University.

En nuestras pruebas a partir de cada base de datos se seleccionaron de forma aleatoria 10 pares de particiones donde en cada partición se toma el 75 % de los casos para el entrenamiento² y el resto para prueba. En la tabla 1 se muestran los resultados de comparar el AG diseñado con el algoritmo de Hong al aplicarlos sobre la partición 1 generada para las pruebas realizadas.

Partición1 Resultados	Tiroides		Corazón	
	Entrena- miento	Prueba	Entrena- miento	Prueba
AG	96.39	95.26	95.26	88.6
Hong	87.4	83.9	71.8	88.1

Tabla 1: Partición 1 Resultados experimentales

Para comparar los dos algoritmos se utilizó el test de comparación de rangos U-Man-Whitney, utilizándose para computar el nivel de significación el método de Monte Carlo con consideración de un intervalo del confianza para la significación del

² El proceso de entrenamiento se refiere a la construcción automática de las funciones de pertenencia y la generación de reglas a partir de ellas.

99%. En la aplicación de las pruebas estadísticas se utilizó el sistema SPSS versión 8.0.

La tabla 2 muestra los resultados de la aplicación de las técnicas estadísticas citadas anteriormente. Superíndices diferentes en los algoritmos indican grupos significativamente diferentes; a los mejores grupos les corresponde una letra mayor y más los más malos una menor. Por tanto se muestra en la tabla que el algoritmo del grupo 1 es mejor que el algoritmo del grupo 2 indicándose que el uso de los algoritmos genéticos reporta resultados significativamente mejores que el uso del algoritmo de Hong.

Grupos	Tiroides (entrenamiento y prueba)	Corazón (entrenamiento)
1	AG ^b	AG ^b
2	Hong ^a	Hong ^a

Tabla 2. Comparación por medio del test U Man Whitney.

6. Conclusiones

Analizamos en este trabajo diferentes enfoques representativos de funciones de pertenencia, particularizando en las funciones campana Beta y las Triángulo. Así como diversos métodos para la construcción de funciones de pertenencia.

Como caso particular se analizó el uso de los Algoritmos Genéticos en la obtención de funciones de pertenencia campana Beta y Triángulo. Se mostró además las potencialidades de los AG en la obtención de funciones de pertenencia tanto en datos numéricos como simbólicos. El diseño del algoritmo genético desarrollado posibilita la resolución y el planteamiento del problema de construcción automática de las funciones de pertenencia como un problema multiobjetivo.

Se discutieron las potencialidades y desventajas de diversos métodos reportados en la bibliografía para la construcción de funciones de pertenencia. Se compararon además los algoritmos genéticos con el método propuesto en [Hong98] por medio de la comparación de los resultados de aplicar ambos métodos en la clasificación de bases de casos internacionales.

Referencias

- [Baldwin98] J. Baldwin, J. Lawry, T. Martin. 'The application of generalised fuzzy rules to machine learning and automated knowledge discovery'. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 5. Vol. 6. pp. 459-487. (1998).
- [Bilgic97] T. Bilgic, B. Türksen. 'Measurement of Membership Functions: Theoretical and Empirical Work'. *Handbook of Fuzzy Sets and Systems*, 1. Foundations. (1997).
- [Blake98] C. L. Blake, C. J. Merz, UCI Repository of Machine Learning databases. University of California, Department of Information and Computer Science. January 2002.
<http://www1.ics.uci.edu/~mlearn/MLRepository.html>
- [Cox98] E. Cox, R. Taber, M. O'Hagan. 'The Fuzzy Systems Handbook'. AP Professional, Paperback, 2nd Bk&Cd edition. (1998).
- [Chen00] Y. Chen, C. Chiu. 'New estimation method for the membership values in fuzzy sets'. *Fuzzy Sets and Systems*, 112. pp. 521-525. (2000).
- [Chen95] J. Chen, K. Otto. 'Constructing membership functions using interpolation and measurement theory'. *Fuzzy Sets and Systems*, 73. pp. 313-327. (1995).
- [Giles88] R. Giles. 'The Concept of Grade of Membership'. *Fuzzy Sets and Systems*, 25. (1988).
- [Golberg89] D. Golberg. 'Genetic Algorithms in Search, Optimization, and Machine Learning'. Addison-Wesley, Reading. U.S.A. (1989).
- [Holland75] J. Holland. 'Adaptation in Natural and Artificial Systems'. Ann Arbor: The University of Michigan Press. (1975).
- [Hong98] T. Hong, C. Lee. 'Learning Fuzzy Knowledge from Training Examples'. *CIKM*. pp. 161-166. (1998).
- [Jaochu01] J. Jaochu. *Dynamic Weighed Agregation for Evolucionary Multi-Objective Optimization*. Futury tecnology Research, Honda R&D Europe, Alemania, 2001.
- [Mann90] S. Mann. 'An evaluation of the eigenvalue approach for determining the membership values in fuzzy sets'. *Fuzzy Sets and Systems*, (1990).
- [Man99] K. Man, K. Tang, S. Kwong. 'Genetic Algorithms'. Springer-Verlag London Limited. England. (1999).
- [Michalewics94] Z. Michalewics, *Genetic Algorithms + Data Structures = Evolution Programs*, Second Edition (1992).
- [Narazaki94] H. Narazaki, A. Ralescu. 'Iterative induction of a category membership function'. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 1. Vol. 2. pp. 91-100. (1994).
- [Nauck00] D. Nauck. 'Data Analysis with Neuro – Fuzzy Methods'. (Dr.rer.nat.habil.). (2000).
- [Piñero00] P. Piñero. 'GACom: Biblioteca de componentes para el trabajo con Algoritmos Genéticos'. Universidad Central "Marta Abreu" de Las Villas. (2000).