

Tipo de artículo: Artículo original
Temática: Inteligencia Artificial
Recibido: 14/03/2014 | Aceptado: 11/04/2014

Comparación de metaheurísticas para obtener predicados difusos: un caso curioso

Comparison of metaheuristics for obtaining fuzzy predicates: a curious case

Taymi Ceruto-Cordovés^{1*}, Orenia Lapeira-Mena¹, Alejandro Rosete-Suárez¹, Rafael Espín-Andrade²

¹ Facultad de Ingeniería Informática, Instituto Superior Politécnico “José Antonio Echeverría”, Calle 114, # 11901, e/
Ciclovía y Rotonda, Marianao, La Habana, Cuba

² Universidad de Occidente (UdeO), Sinaloa, México

*Autor para correspondencia: tceruto@ceis.cujae.edu.cu; rafaelespin@yahoo.com
{[olapeira](mailto:olapeira@ceis.cujae.edu.cu), [rosete](mailto:rosete@ceis.cujae.edu.cu)}@ceis.cujae.edu.cu

Resumen

En este trabajo se presenta un estudio comparativo de tres metaheurísticas en el problema de obtener predicados difusos con alto valor de verdad. Según el Teorema *No Free Lunch* (NFL) no se puede establecer la superioridad general de ninguna metaheurística sobre las otras. Este trabajo demuestra que incluso dentro del mismo tipo de problema, puede ser difícil de establecer la superioridad de una metaheurística. En este caso, cada metaheurística logra ser la mejor al menos en una de las cuatro variantes de operador difuso empleado y de forma normal del predicado obtenido. Este curioso caso, revela la importancia de la comparación experimental de metaheurísticas, antes de asumir la superioridad de una sobre las otras.

Palabras clave: Formas normales, metaheurísticas, operadores de lógica difusa, predicados difusos.

Abstract

This paper presents a comparative study of three metaheuristics on the problem of obtaining fuzzy predicates with high truth value. According to the No Free Lunch Theorem (NFL) cannot establish any general superiority of any metaheuristic over the others. This work demonstrates that even within the same type of problem can be difficult to establish the superiority of a metaheuristic. In this case, each metaheuristic is the best at least in one of the four variants of fuzzy operator employed and normal form of the obtained predicate. This curious case reveals the importance of the experimental comparison of metaheuristics, before assuming the superiority of one over the other.

Keywords: *Metaheuristic, fuzzy logic operators, normal forms, fuzzy predicates.*

Introducción

El término metaheurísticas se obtiene de anteponer a heurística el sufijo “meta” que significa "más allá" o "a un nivel superior". Este término fue introducido por primera vez por Fred Glover, el cual la define como una estrategia de alto nivel que utiliza diferentes métodos para explorar el espacio de búsqueda del problema, transformando de forma iterativa las soluciones de partida (Melian, *et al.*, 2003; Blum & Roli, 2003). Las metaheurísticas generalmente se aplican a problemas que no tienen un algoritmo específico que dé una solución satisfactoria; o bien cuando no es posible implementar un método óptimo. Existen muchas metaheurísticas (Talbi, 2009), y dos de las más conocidas son los Escaladores de Colinas (también llamada Búsqueda Local) y los Algoritmos Genéticos.

Ninguna metaheurística es mejor que otra en la totalidad de los problemas en las que son aplicables; depende de la codificación del problema, de la correcta selección de la función objetivo, así como los operadores que permiten variar de un estado a otro del problema. Según el Teorema NFL, los métodos generales de búsqueda, entre los que se encuentran las metaheurísticas, se comportan exactamente igual cuando se promedian sobre todas las funciones objetivo posibles, de tal forma que si un algoritmo A es más eficiente que un algoritmo B en un conjunto de problemas, debe existir otro conjunto de problemas de igual tamaño para los que el algoritmo B sea más eficiente que el A (Wolpert & Macready, 1997). A partir de la imposibilidad de establecer la ventaja absoluta de una metaheurística sobre otra en todos los problemas, una idea práctica que puede inferirse, es la conveniencia de identificar problemas donde una metaheurística tenga ventaja (Mitchell, *et al.*, 1994).

En particular la extracción de predicados difusos desde una base de datos (*FuzzyPred*) (Ceruto, *et al*, 2013-2014) puede ser visto como un problema de optimización combinatoria donde el objetivo es maximizar el valor de verdad de los predicados en la base de datos. Este método se fundamenta en el paradigma de la lógica de predicados y el cálculo de sus valores de verdad mediante diferentes operadores de lógica difusa. Además utiliza métodos de búsqueda de diferente naturaleza (basados en un punto y poblacionales) en el espacio de los predicados, para mejorar la veracidad de los mismos. La combinación de ambos factores guarda relación y es por ello que este trabajo se enfoca en el análisis experimental de las diferentes combinaciones entre metaheurísticas y operadores difusos para determinar cuál es el mejor en cada caso. Particularmente, este trabajo se enfoca en estudiar el comportamiento de tres metaheurísticas en el problema de la extracción de predicados difusos, con la intención de identificar la metaheurística que tiene mejor comportamiento en este problema.

En la siguiente Sección se describen las metaheurísticas empleadas en el estudio experimental. Los elementos de Lógica Difusa que son relevantes para este trabajo se describen en la sección 3. La sección 4 muestra los resultados obtenidos por *FuzzyPred* sobre dos bases de datos (Economía Mexicana y Diabetes). Finalmente, en la sección 5 se muestran algunas conclusiones a las que se pudo arribar luego de realizados los experimentos.

Materiales y métodos

Metaheurísticas

Una metaheurística es un método heurístico para resolver un tipo de problema computacional general, usando los parámetros dados por el usuario sobre unos procedimientos genéricos y abstractos de una manera que se espera eficiente. La mayoría de las metaheurísticas tienen como objetivo los problemas de optimización combinatoria. El objetivo de la optimización combinatoria es encontrar una solución que maximice (o minimice, dependiendo del problema) una función especificada por el usuario. A estos objetos se les suele llamar estados, y al conjunto de todos los estados candidatos se le llama espacio de búsqueda. La naturaleza de los estados y del espacio de búsqueda son usualmente específicos del problema (Talbi, 2009).

FuzzyPred (Ceruto *et al*, 2013-2014) es un método híbrido de reciente creación que se enmarca dentro del proceso de Descubrimiento de Conocimiento en Bases de Datos y tiene como objetivo fundamental descubrir predicados difusos con alto valor de verdad que caractericen los datos. Para la representación de los predicados se utilizan formas normales, modificadores (muy, algo) y se emplea el cuantificador universal para el cálculo del valor de verdad.

Cuando se utilizan formas normales se pueden representar varias cláusulas combinando tres tipos de conectivas lógicas (Bruno, 1998): Conjunción, Disyunción y Negación (sólo afectando a los átomos). Si la conectiva principal es la conjunción, entonces es una forma normal conjuntiva (FNC). Si por el caso contrario, la conectiva principal es la disyunción, entonces es una forma normal disyuntiva (FND). Cada variable en cada cláusula puede tomar diferentes valores, como pueden ser 0 (no está en el predicado), 1 (está en el predicado), 2 (aparece negada) y 3 (aparece con el modificador muy). Este sería un ejemplo de un predicado que se puede obtener $(X_1 \wedge \neg X_2) \vee (X_0 \wedge \neg X_1)$ con la correspondiente codificación de sus dos cláusulas $(x_0=0, x_1=1, x_2=2); (x_0=1, x_1=2, x_2=0)$.

Dada la codificación que se utiliza en *FuzzyPred*, el espacio de búsqueda de las soluciones puede llegar a ser grande, teniendo en cuenta los aspectos que determinan la complejidad del problema: la cantidad de variables involucradas, que a su vez depende de la cantidad de cláusulas del predicado; y los posibles valores que puede tomar cada variable. Si se tuviera un problema que utilizara un rango de 3 valores posibles valores por variable, 3 cláusulas y 5 variables, entonces el espacio de soluciones: $3^{15} = 14348907$. Independientemente de que la cantidad de combinaciones no fuera tan grande, de todos modos no es factible desde el punto de vista computacional hacer el cálculo de la función objetivo para todas las combinaciones posibles, por la dimensión que pueda tener la base de datos. Teniendo en cuenta que no existe un algoritmo exacto que permita su resolución, se decidió utilizar metaheurísticas para guiar el proceso de búsqueda de los predicados.

Entre las metaheurísticas más conocidas se encuentran la Búsqueda Aleatoria, Escalador de Colinas, Recocido Simulado, Búsqueda Tabú, Algoritmos Genéticos, Estrategias Evolutivas, Algoritmo de Estimación de Distribuciones, etc (Blum & Roli, 2003; Talbi, 2009). Actualmente está demostrado que estos algoritmos se comportan de la misma forma al ser promediados sobre todas las posibles funciones a optimizar, por lo tanto no existe un algoritmo mejor que otro. Este razonamiento está planteado en el *Teorema No Free Lunch* (NFL), demostrado por David Wolpert y William Macready en (Wolpert & Macready, 1997). En particular para los experimentos se decidió utilizar la Búsqueda Aleatoria (muy simple), Escalador de Colinas (basado un punto) y Algoritmo Genético (poblacional). Estos tres métodos son bastante diferentes, y pueden verse como ejemplos de paradigmas de búsqueda. Por esta razón, han sido estudiados antes con la intención de identificar cuándo gana cada uno (Mitchell, *et al*, 1994). Las características que los distinguen se exponen a continuación.

Búsqueda Aleatoria (*Random Search, RS*)

La Búsqueda Aleatoria es la metaheurística más simple. No utiliza soluciones anteriores para guiar la búsqueda, solo considera la solución que devuelve. En cada iteración sigue una estrategia de generación aleatoria de un elemento del espacio de estados, siguiendo generalmente una distribución uniforme. Todos los puntos son equiprobables (Melián *et al.*, 2003; Marti, 2003; Talbi, 2009).

Escalador de Colinas (*Hill Climbing, HC*)

El Escalador de Colinas se usa frecuentemente cuando se tiene una buena función heurística para evaluar los estados y no se dispone de otro conocimiento útil. La idea es desplazarse en el espacio en la dirección de mejoramiento del valor de la función objetivo, con la intención de llegar a un estado óptimo. Para ello utiliza la información brindada por las soluciones anteriores que han sido visitadas.

Una mejor idea para entender este método es describir la situación de la vida real a la cual se asemeja. Un alpinista (escalador de colinas) el cual padece de amnesia y se encuentra cerca del Monte Everest en medio de una espesa neblina que sólo le permite ver a unos pocos metros de su ubicación actual y su objetivo es alcanzar la cima. La estrategia que sigue el escalador para alcanzar su meta es irse moviendo hacia los lugares que desde su posición actual le parecen que indican un ascenso hacia la cima (Blum, 2003; Talbi, 2009).

El HC es un algoritmo de mejoramiento iterativo. Comienza a partir de un punto (punto actual) en el espacio de búsqueda. En cada iteración, un nuevo punto es seleccionado de la vecindad del punto actual. Si el nuevo punto es mejor, se transforma en un punto actual, sino otro punto vecino es seleccionado y evaluado. El método termina cuando no hay mejorías, o cuando se alcanza un número predefinido de iteraciones.

Los HC son métodos de propósito general muy sencillos. Su limitación esencial es la propiedad de converger al óptimo local más cercano, sin garantizar de ningún modo que este sea el óptimo global. Sin embargo, muchas veces esta limitación se exagera. El problema real está en que puede quedarse en un óptimo local que no sea global y que además esté lejos de serlo.

Algoritmos Genéticos (*Genetic Algorithms, GA*)

En una búsqueda basada en poblaciones (o en grupo) se sustituye la solución actual que recorre el espacio de soluciones, por un conjunto de soluciones que lo recorren conjuntamente interactuando entre ellas. Además de los movimientos aplicables a las soluciones que forman parte de este conjunto, se contemplan otros operadores para

generar nuevas soluciones a partir de las ya existentes. El resultado final proporcionado por este tipo de algoritmos depende fuertemente de la forma en que manipula la población (Blum & Roli, 2003; Talbi, 2009).

Para llevar a cabo la selección de los individuos que se van a reproducir, en GA existen diversos métodos, entre los que se encuentran (Talbi, 2009): Método de la ruleta, Por Truncamiento, Elitismo, Por Rango y Por Torneo. Una vez seleccionados los individuos, estos se intercambian con una determinada probabilidad para producir la nueva descendencia. Para ello se utilizan diversos métodos, entre los que se encuentran: Cruce en un punto, Cruce en dos puntos, Cruce uniforme. Si el cruce tiene éxito entonces uno de los descendientes, o ambos, se muta con cierta probabilidad, utilizando diversos métodos: Inversión de genes, Cambio de orden, Modificación de genes, Mutación uniforme, etc. Después de realizar cruce y mutación en los individuos de la población, se debe decidir si se aceptan los hijos generados; para lo cual existen varias técnicas: Aleatorio, Reemplazo generacional, Reemplazo de similares, Reemplazo de los peores (Talbi, 2009).

Obtención de Predicados Difusos

La lógica difusa (Zadeh, 1965; Zadeh, 1983; Zadeh, 1989) constituye una herramienta de representación del conocimiento muy importante. La misma permite modelar conocimiento impreciso y cuantitativo; así como transmitir, manejar incertidumbre y soportar, en una extensión razonable, el razonamiento humano de una forma natural. Se ha aplicado en muchas áreas de la investigación fundamentalmente por su cercanía al razonamiento humano y por proporcionar una forma efectiva para capturar la naturaleza aproximada e inexacta del mundo real.

El éxito de la lógica difusa está directamente asociado a los Sistemas Difusos y en particular a los Controladores Difusos, que han sido ampliamente utilizados en el control automático de procesos. No obstante, la afinidad de la lógica difusa con la forma en que se representa el conocimiento humano, hace que su uso se haya hecho extensivo en el campo de la minería de datos (Hernández *et al*, 2004), fundamentalmente en las reglas de asociación y particularmente en las reglas de asociación cuantitativas (son las que se generan a través del análisis de datos numéricos). Este enfoque ha sido desarrollado por un conjunto amplio de autores: (Hong & Kuo, 1999; Mata, 2002; Alcalá & *et al*, 2009; Venugopal, 2009) y también ha estado vinculado al uso de algoritmos metaheurísticos como los Algoritmos Genéticos (*Genetic Algorithm*, GA) y la Colonia de Hormigas (*Ant Colony*, ACO). En general, las investigaciones en que mezcla la lógica difusa y las metaheurísticas es un campo muy activo, formando parte de lo que hoy se conoce como *SoftComputing* (Verdegay, 2005).

Una característica a destacar en la lógica difusa es que no existe una definición única de algunas de las operaciones clásicas como la unión o la intersección de conjuntos, sino que existen múltiples formas de desarrollar estas operaciones (Mitsubishi *et al.*, 2001). Para la evaluación de cada predicado en *FuzzyPred* (Ceruto *et al.*, 2013; Ceruto *et al.*, 2014) se puede elegir entre diferentes operadores de lógica difusa existentes, como por ejemplo: el max-min propuesto por (Zadeh, 1965) o la media geométrica y su dual que son operadores compensatorios (Espin & *et. al.*, 2006). La elección del par de operadores de lógica difusa es responsabilidad del usuario y para lograr una correcta elección, el usuario debe tener en cuenta las características de cada uno y su comportamiento.

En la intersección (conjunción) se utiliza una norma triangular o T-norma. Algunas de las funciones asociadas frecuentemente a la operación de intersección son el Mínimo y el Producto, aunque existen otras. Algo similar sucede con la unión (disyunción), en la cual se utiliza una conorma triangular, T-conorma ó S-norma. Aquí las funciones asociadas con mayor frecuencia son el Máximo y la Suma Algebraica, aunque también existen muchas más. Para el caso del complemento (negación) la función es una C-norma, y aunque tiene igualmente varias funciones asociadas, existe un mayor consenso para el uso de $n(x) = 1 - x$.

Los operadores que se utilizan con mayor frecuencia tanto en las normas, como en las conormas, tienen como limitaciones la asociatividad ($p \vee (q \vee r) = (p \vee q) \vee r$). Si un sistema es asociativo o no es sensible (cambios de los valores de verdad de los predicados básicos no se reflejan en el valor de verdad obtenido del predicado compuesto) o no es idempotente ($p \wedge p = p$; $p \vee p = p$). Es por ello que algunos autores (Espin & *et. al.*, 2006) defienden la idea de utilizar sistemas que no sean asociativos y para ello proponen el uso de la Lógica Difusa Compensatoria (LDC). La misma es sensible ante cambios en cualquiera de las variables, es idempotente, permite la compensación de los valores de unas variables con otras y facilita la interpretación de los resultados. A continuación se muestran dos tablas comparativas (ver Tabla 1 y 2) entre diferentes operadores estudiados, la cual permite reafirmar las propiedades antes mencionadas. No obstante, en *FuzzyPred* se brinda la posibilidad de que el usuario elija el par de operadores que desea utilizar.

Tabla 1. Comportamiento del operador de conjunción.

Lógica	Idempotencia	Sensibilidad	Compensación
Zadeh (Mínimo)	Si $0.5 \wedge 0.5 = 0.5$	No $0.5 \wedge 0.8 = 0.5$	No Mínimo
Probabilística (Producto)	No $0.5 \wedge 0.5 = 0.25$	Si $0.5 \wedge 0.8 = 0.4$	No Debajo Mínimo
Compensatoria (Media Geométrica)	Si $0.5 \wedge 0.5 = 0.5$	Si $0.5 \wedge 0.8 \sim 0.63$	Si El valor se encuentra entre el mínimo y el máximo

Tabla 2. Comportamiento del operador de disyunción.

Lógica	Idempotencia	Sensibilidad	Compensación
Zadeh (Máximo)	Si $0.5 \vee 0.5 = 0.5$	No $0.5 \vee 0.3 = 0.5$	No Máximo
Probabilística (Suma algebraica)	No $0.5 \vee 0.5 = 0.75$	Si $0.5 \vee 0.3 = 0.65$	No Encima Máximo
Compensatoria (Dual media Geométrica)	Si $0.5 \vee 0.5 = 0.5$	Si $0.5 \vee 0.3 \sim 0.40$	Si El valor se encuentra entre el mínimo y el máximo

Resultados y discusión

En esta sección se muestra un estudio comparativo de tres metaheurísticas (RS, HC, GA) en el problema de la obtención de predicados difusos. Se realiza la comparación variando el operador difuso que se utiliza para calcular el valor de verdad de la conjunción y de la disyunción. En los estudios se emplearán los operadores de Zadeh (Min-Max) y los operadores compensatorios.

Para comprobar la relación que puede existir entre el operador de lógica difusa seleccionado y las metaheurísticas, se realizaron experimentos sobre dos bases de datos: la primera es una pequeña base de ejemplos del mundo real referente a la economía mexicana (21 tuplas) utilizada por (Espin & *et al*, 2011) y la segunda es referida a pacientes de la comunidad de Jaruco que padecen o están en riesgo de padecer diabetes (7672 tuplas) utilizada por (Acosta & *et*

al, 2009). En el primer caso, el proceso estuvo encaminado a que el algoritmo descubriera relaciones en corto tiempo entre las siguientes variables: Inflación (I), Producto Interno Bruto (PIB) y la Paridad (P). En el caso de la segunda base de datos las variables utilizadas fueron las siguientes: Edad avanzada (EA), Piel blanca (PB), Presión alta (PA), Obeso (O) y Diabético Detectado (DD).

La ejecución de cada experimento estuvo determinada por:

- Estructura del predicado: Forma normal conjuntiva (FNC) y Forma normal disyuntiva (FND) sólo para dos cláusulas.
- Algoritmo: RS, HC (con Primer Ascenso y aceptando sólo las soluciones mejores) y GA (Selección por truncamiento, Cruzamiento uniforme, Mutación en un punto). Y en particular el GA fue configurado de la siguiente manera: tamaño de la población (20), truncamiento (10), la probabilidad de cruzamiento (0.9), la probabilidad de mutación (0.5), cruzamiento en un punto.
- Operador de lógica difusa: Zadeh (Min-Max), LDC (Media Geométrica y su Dual).
- Cada algoritmo se repitió 30 veces para una cantidad máxima de 500 iteraciones.

Para el experimento con FNC-Zadeh como promedio el algoritmo que mejor se comportó fue la Búsqueda Aleatoria. En cambio para el experimento con FNC-LDC el mejor fue el Escalador de Colinas (ver figura 1).

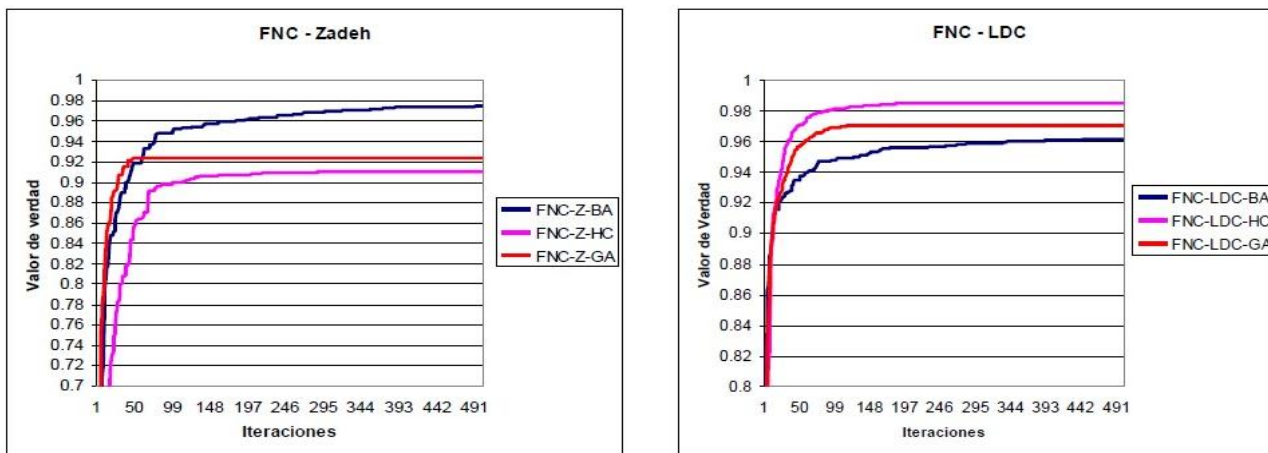


Figura 1. Economía Mexicana- FNC - (Zadeh- LDC).

Para los experimentos FND-Zadeh y FND-LDC como promedio el algoritmo que mejor se comportó fue el Algoritmo Genético (ver Figura 2).

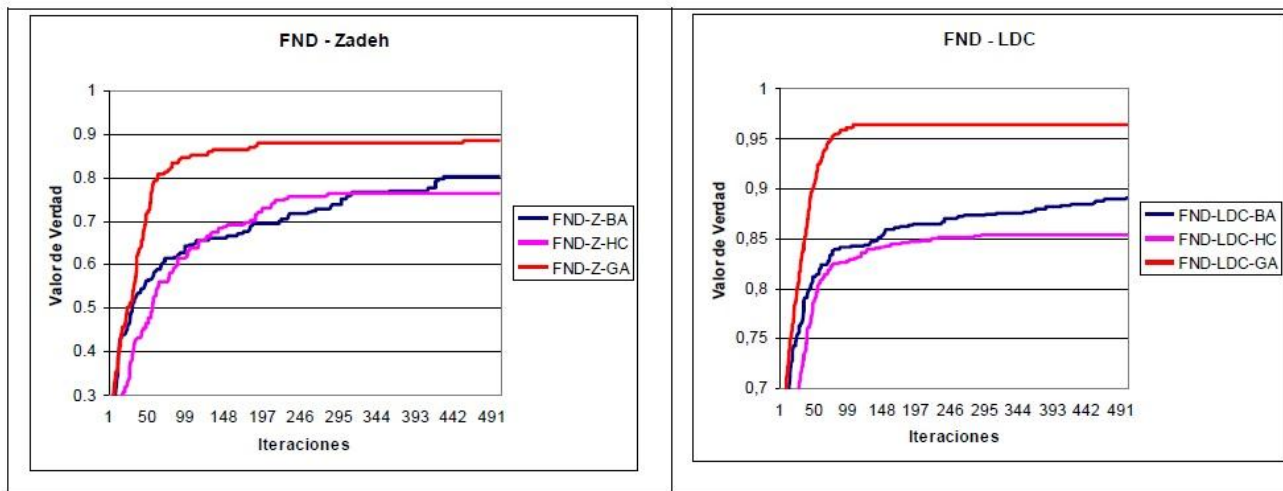


Figura 2. Economía Mexicana- FNC - (Zadeh- LDC).

Para el experimento con FNC-Zadeh como promedio el algoritmo que mejor se comportó fue la Búsqueda Aleatoria, en cambio FNC-LDC el mejor fue el Escalador de Colinas (ver Figura 3).

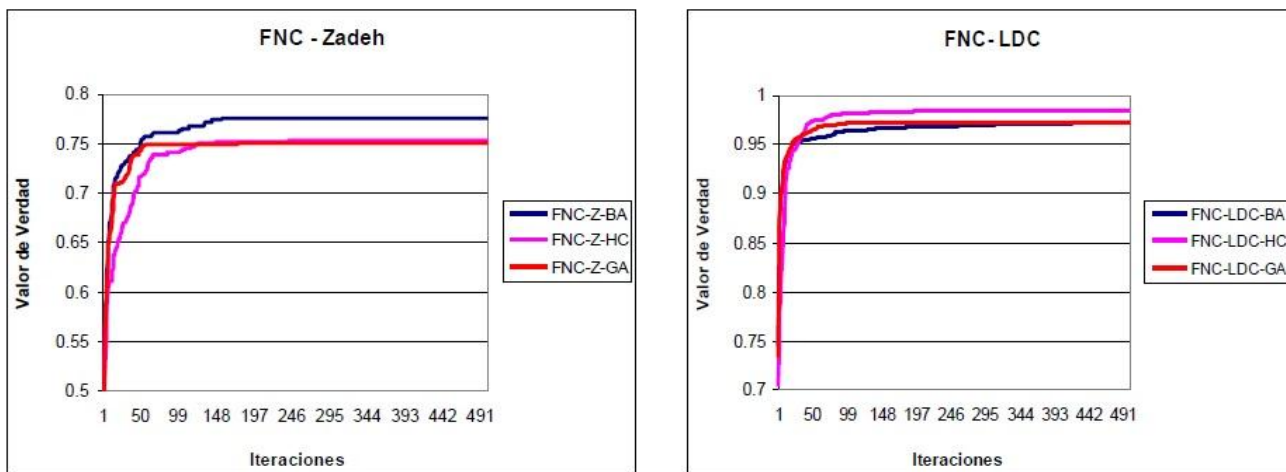


Figura 3. Diabetes - FNC- (Zadeh-LDC).

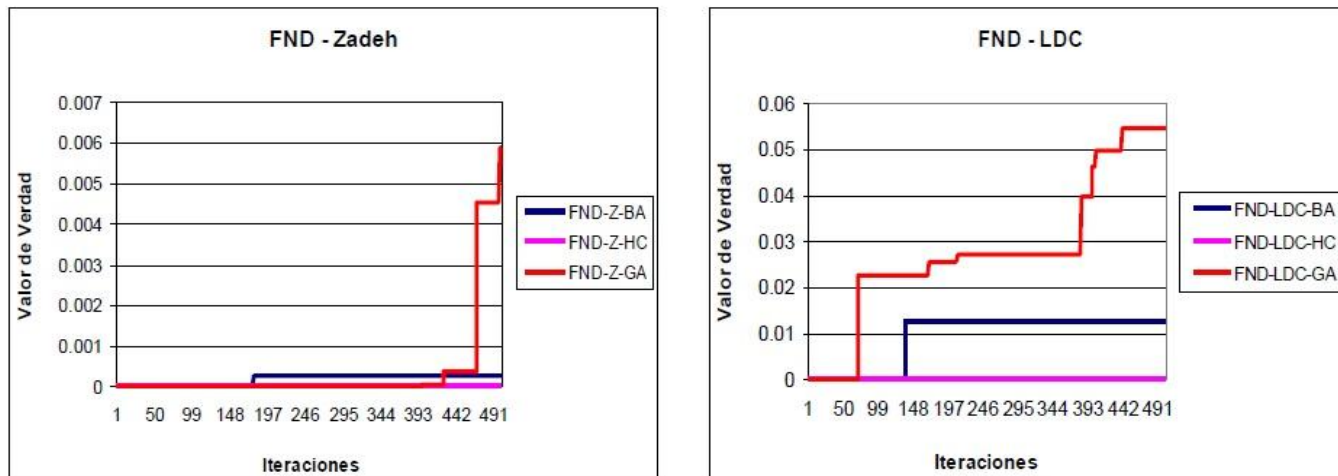


Figura 4. Diabetes - FND- (Zadeh-LDC).

Los experimentos realizados con FND - Zadeh y FND - LDC, arrojaron como resultado predicados más falsos que verdaderos (ver figura 4). Es curioso notar que en este caso, ninguno de los algoritmos logró encontrar predicados con alto valor de verdad. Podría pensarse que los malos resultados en este caso están influidos por el mayor tamaño del espacio, debido a la existencia de más variables. En Economía Mexicana el tamaño del espacio de búsqueda es de $9^6 = 531441$ y en el de Diabetes $9^{10} = 3486784401$. Sin embargo, puede notarse que en la obtención de FNC (figura 3) este problema no se observa. Este es un ejemplo concreto de lo difícil que puede resultar la predicción del comportamiento de las metaheurísticas. A pesar de este comentario, los resultados de la Figura 4 muestran que los Algoritmos Genéticos obtienen los mejores resultados y de habersele dado más iteraciones a lo mejor hubiera podido encontrar mejores soluciones.

De forma curiosa, en ambos casos de estudio el mejor algoritmo de acuerdo al tipo de experimento siempre fue el mismo (ver Tabla 3).

Tabla 3. Mejores algoritmos.

Tipo de Experimento	Mejor Algoritmo
FNC-Zadeh	Búsqueda Aleatoria
FNC-LDC	Escalador de Colinas
FND-Zadeh	Algoritmo Genético
FND-LDC	Algoritmo Genético

En el caso de (FNC o FND) utilizando Zadeh, el Escalador de Colinas puede tender a comportarse mal. Esto se debe a que la vecindad posee mesetas (soluciones con igual valor de verdad que la solución actual) de las cuales le es difícil escapar, porque la configuración actual del algoritmo no le permite aceptar soluciones iguales, solo las mejores. Esto le impide moverse en la meseta y tratar de acercarse a una solución, que le permita dar un salto cualitativo. Este tipo de problema es fácilmente evitado por la búsqueda aleatoria.

La existencia de mesetas usando los operadores de Zadeh se deben a la falta de sensibilidad de este operador, que hace que muchos predicados tengan el mismo valor de verdad. Un cambio en una variable importante debería provocar un cambio del valor de verdad del predicado y no siempre ocurre así. Esta falta de sensibilidad afecta a HC y a GA, lo que hace que estos no superen a la Búsqueda Aleatoria en este caso.

El siguiente ejemplo evidencia la falta de sensibilidad de los operadores de Zadeh:

$$(\neg I^3 \vee \neg PIB^3 \vee P^2) \wedge (\neg I^2 \vee \neg PIB \vee P^2) \quad \text{Valor de verdad} = 0.25$$

$$(I^{0.5} \vee \neg PIB^3 \vee P^2) \wedge (\neg I^2 \vee \neg PIB \vee P^2) \quad \text{Valor de verdad} = 0.25.$$

Esto no sucede en LDC:

$$(\neg I^3 \vee \neg PIB^3 \vee P^2) \wedge (\neg I^2 \vee \neg PIB \vee P^2) \quad \text{Valor de verdad} = 0.64$$

$$(I^{0.5} \vee \neg PIB^3 \vee P^2) \wedge (\neg I^2 \vee \neg PIB \vee P^2) \quad \text{Valor de verdad} = 0.80$$

Peor aún es cuando el HC encuentra un óptimo local. Por ejemplo la siguiente solución: $x_0=7, x_1=7, x_2=3, x_3=6, x_4=2, x_5=3 \rightarrow (\neg I^3 \vee \neg PIB^3 \vee P^2) \wedge (\neg I^2 \vee \neg PIB \vee P^2)$ de la Economía Mexicana fue obtenida en FNC con 2 cláusulas utilizando Zadeh. Este predicado tiene un valor de verdad de 0.25, que no es superado por ninguna solución de su vecindario. En la siguiente figura se muestra la vecindad de este predicado (48 soluciones), donde las columnas representan las variables involucradas en el predicado y las filas los posibles valores de cada variable (ver figura 5). Los elementos en azul representan los valores de verdad de los vecinos que constituyen mesetas y los rojos son los predicados con menor valor de verdad que el original. Por ejemplo el predicado que está en la intersección de la primera fila y la primera columna tendría como valor de verdad 0.25 y sería: $x_0=0, x_1=7, x_2=3, x_3=6, x_4=2, x_5=3 \rightarrow (\neg PIB^3 \vee P^2) \wedge (\neg I^2 \vee \neg PIB \vee P^2)$ donde la mutación con respecto al predicado original sería en la variable x_0 que cambia de $x_0=7$ (Hiper Negado) a $x_0=0$ (No está). Las intersecciones que están vacías representan los valores originales de las variables del predicado al cual se le está analizando la vecindad.

	I	G	P	I	G	P
0 (No está)	0.25	0.04	9.94E-4	0.25	0.04	0.09
1 (Esta)	0.25	0.04	0.25	0.25	0.04	0.25
2 (Negada)	0.25	0.25	0.25	0.25		0.25
3 (Muy)	0.25	0.04		0.25	0.04	
4 (Hiper)	0.25	0.04	0.12	0.25	0.04	0.12
5 (Algo)	0.25	0.12	0.25	0.25	0.12	0.25
6 (Muy Negado)	0.25	0.25	0.24		0.25	0.24
7 (Hiper Negado)			0.12	0.25	0.25	0.12
8 (Algo Negado)	0.25	0.25	0.25	0.25	0.25	0.25

Figura 5. Análisis de un óptimo local.

Si al menos el HC aceptara las soluciones iguales, podría caer en uno de sus vecinos como por ejemplo: $x_0=5$, $x_1=7$, $x_2=3$, $x_3=6$, $x_4=2$, $x_5=3 \rightarrow (I^{0.5} \vee \neg PIB^3 \vee P^2) \wedge (\neg I^2 \vee \neg PIB \vee P^2)$. Después de estar en esta solución, se podría llegar a una solución mejor en este nuevo vecindario (zonas verdes de la Figura 6).

	I	G	P	I	G	P
0 (No está)	0.25	0.25	0.25	0.25	0.04	0.09
1 (Esta)	0.25	0.25	0.25	0.89	0.04	0.5
2 (Negada)	0.25	0.25	0.25	0.25		0.49
3 (Muy)	0.25	0.25	0.25	0.89	0.04	
4 (Hiper)	0.25	0.25	0.25	0.89	0.04	0.12
5 (Algo)		0.25	0.25	0.89	0.12	0.71
6 (Muy Negado)	0.25	0.25	0.25		0.25	0.24
7 (Hiper Negado)	0.25	0.25	0.25	0.25	0.25	0.12
8 (Algo Negado)	0.25	0.25	0.25	0.25	0.31	0.70

Figura 6. Análisis de la vecindad.

A otra conclusión importante a la que se puede arribar es que en los experimentos de FNC, la convergencia de los algoritmos fue temprana (iteración 150), por lo que no puede esperarse que los resultados mejoren si se permitieran

más iteraciones. En general, estos resultados son muy relevantes desde el punto de vista didáctico, porque muestran como en un mismo problema (obtención de predicados difusos) pueden ganar y perder las tres metaheurísticas en comparación. Esto reafirma la necesidad de estudios experimentales entre los algoritmos metaheurísticos, y no extrapolar de manera simplista los resultados de un problema a otro.

Conclusiones

En este trabajo se ha presentado un estudio experimental entre tres metaheurísticas en el problema de obtener predicados difusos con alto valor de verdad. Para la evaluación del valor de verdad de los predicados se han utilizado dos variantes de operadores (Zadeh y compensatorios), y se ha estudiado la obtención de predicados en forma normal conjuntiva y en forma normal disyuntiva. Los resultados en ambas bases de datos (Economía Mexicana y Diabetes) fueron consistentes, resultando que los algoritmos ganadores para cada combinación de tipo de forma normal y de operador fueron iguales en ambas bases de datos. Sin embargo, lo curioso de los resultados es que las tres metaheurísticas resultaron ganadoras en alguna de las combinaciones de operador difuso con forma normal. En la combinación FNC-Zadeh el mejor algoritmo fue la Búsqueda Aleatoria, para FNC-LDC fue el Escalador de Colinas y para FND (independientemente del operador de lógica difusa utilizado) fue el Algoritmo Genético. El Escalador de Colinas con los operadores de Zadeh tuvo como tendencia un mal comportamiento, debido a la presencia de mesetas y óptimos locales. Este resultado, y la explicación de los mismos que se incluye en el trabajo subrayan la importancia de no asumir la superioridad de una metaheurística sobre otra, sin tener una base experimental que lo sustente.

Referencias

- ACOSTA R., ROSETE A., RODRÍGUEZ A. *Predicción de pacientes diabéticos. Preprocesado para Minería de Datos*, Revista Cubana de Informática Médica (RCIM), 2009, 1(9), ISSN: 1684-1859.
- ALCALA-FDEZ, J., ALCALA, R., GACTO, M. J., HERRERA, F. *Learning the membership function contexts for mining fuzzy association rules by using genetic algorithms*, Fuzzy Sets and Systems, 2009, 160(7): p. 905–921.

- BLUM C., ROLI A. *Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison*, ACM Computing Surveys, 35(3): p. 268–308, 2003.
- BRUNO, A.D. *Normal forms*, Mathematics and Computers in Simulation, 1998, 45: p. 413-427.
- CERUTO T., LAPEIRA O., ROSETE A. & ESPIN, R. *Discovery of fuzzy predicates in database*, Atlantis Press, 2013, p. 45-54, ISSN 1951-6851.
- CERUTO, T., SUÁREZ, A., ESPIN R. *Knowledge Discovery by Fuzzy Predicates*. In Soft Computing for Business Intelligence, 2014, 537: p. 187-196, ISSN 1860-949X, Springer Berlin Heidelberg.
- ESPIN R., FERNANDEZ E., MAZCORRO G., MARX-GÓMEZ J. & LECICH M. *Compensatory Logic: A fuzzy normative model for decision making*, Investigación Operacional, 2006, 27 (2): p. 178-193.
- ESPIN R., CHAO A. MARX-GÓMEZ J. & RACET A. *Fuzzy Semantic Transdisciplinary Knowledge Discovery Approach for Business Approach* in Towards a Transdisciplinary Technology for Business Intelligence: Gathering Knowledge Discovery, Knowledge Management and Decision Making, 2011, Germany, ISBN 3832297189, p. 13-34.
- HERNÁNDEZ ORALLO, J.; RAMÍREZ QUINTANA, M. J.; FERRI RAMÍREZ, C. *Introducción a la minería de datos*. Ed. PEARSON EDUCACIÓN, S.A, 2004. ISBN 84-205-4091-9.
- HONG, T.-P., C.-S. KUO. *Mining association rules from quantitative data*, Intelligent Data Analysis, 1999, 3: p. 363-376.
- MATA, J., ÁLVAREZ J.-L. *Discovering Numeric Association Rules via Evolutionary Algorithm*. PAKDD 2002, LNAI 2336., Springer-Verlag Berlin Heidelberg: p. 40-51.
- MELIÁN, B., MORENO PÉREZ, J.A., MORENO VEGA, J.M. *Metaheurísticas: una visión global*, Revista iberoamericana de inteligencia artificial 19 (2): p. 7-28. 2003.
- MITCHELL, M., HOLLAND, J. H., FORREST, S.: *When Will a Genetic Algorithm Outperform Hill Climbing?* In Advances in Neural Information Processing Systems 6. LNCS, 1994 San Mateo, CA: Morgan Kaufmann.
- MITSUISHI T., ENDOU N., SHIDAMA, Y. *The Concept of Fuzzy Set and Membership Function and Basic Properties of Fuzzy Set Operation*, Formalized mathematics, 2001, 9 (2): p. 351-356.
- TALBI E. *Metaheuristics: From Design to Implementation*, Ed. John Wiley & Sons, ISBN 978-0-470-27858-1, 2009, p. 18-29.
- ZADEH, L. *Fuzzy Sets*, Information and Control, 1965, p. 338-353.

- ZADEH, L. *The role of fuzzy logic in the management of uncertainty in expert systems*. Fuzzy Sets and Systems, 1983 11: p. 199-227.
- ZADEH, L. Knowledge representation in fuzzy logic. IEEE Transactions on Knowledge and Data Engineering, 1989, 1(1): p. 89-100.
- VENUGOPAL K., SRINIVASA K. AND PATNAIK L. *Soft Computing for Data Mining Applications*. Studies in Computational Intelligence, Springer-Verlag, Berlin Heidelberg, 2009, 190: pp. 1-15, ISBN 978-3-642-00192-5.
- VERDEGAY J. L. *Una revisión de las metodologías que integran el Softcomputing*. Actas del Simposio sobre Lógica Fuzzy y Soft Computing LFSC 2005 (LFSC2005), 2005, Granada, Spain, p. 151-156.
- WOLPERT D., MACREADY W., *No Free Lunch Theorems for Optimization*, IEEE Transactions on Evolutionary Computation, 1997, 1: p. 67-82.