Tạp chí Tin học và Điều khiển học, T.26, S.2 (2010), 103–117

PHƯƠNG PHÁP TỐI ƯU PARETO HỆ LUẬT MỜ DỰA TRÊN ĐẠI SỐ GIA TỬ SỬ DỤNG GIẢI THUẬT DI TRUYỀN VÀ ỨNG DỤNG VÀO BÀI TOÁN PHÂN LỚP

NGUYỄN CÁT HỒ[1], DƯƠNG THĂNG LONG[2], TRẦN THÁI SƠN[1], TRẦN DUY HÙNG[2]

[1] *Viện Công nghệ thông tin, Viện Khoa học và Công nghệ Việt Nam*
[2] *Khoa Công nghệ Tin học, Viện Đại học Mở Hà Nội*

Abstract. In this paper, we present a method for finding the optimal hedge-algerbas-based fuzzy rules-set based on Pareto optimization algorithm (NSGA-II) proposed by Deb in [5]. The changllenge in methods for constructing classification rules based on fuzzy sets is difficult to represent the rules in linguist, therefore the hedge algebras based methods will overcome this problem. Further, fuzzy rules are extracting directly from the patterns based on the partition of the systems of similarity intervals of terms in AX^2 [15], in which each hyperbox of the partition determins an uniquely fuzzy rule. This may result high performance on computation time and accuracy of classification.

Tóm tắt. Bài báo giới thiệu một phương pháp chọn hệ luật mờ phân lớp dựa trên đại số gia tử bằng phương pháp tìm kiếm tối ưu dựa trên thuật toán NSGA-II của Deb [5], một phương pháp tối ưu Pareto được nhiều tác giả sử dụng. Một trong những thách thức của phương pháp xây dựng luật mờ dựa trên tập mờ là khó diễn tả bằng ngôn ngữ, phương pháp tiếp cận đại số gia tử sẽ khắc phục điều này. Phương pháp sinh hệ luật mờ trực tiếp từ các mẫu dữ liệu, hơn nữa sử dụng hệ phân hoạch các khoảng tính mờ tương tự trong ĐSGT2 [15] sẽ xác định duy nhất mỗi phân hoạch chứa một mẫu dữ liệu. Điều này đem lại hiệu quả lớn về mặt thời gian cũng như tỷ lệ phân lớp đúng của hệ luật.

1. GIỚI THIỆU

Bài toán phân lớp là một trong những bài toán điển hình trong khai phá dữ liệu và nhận dạng mẫu, các mô hình tiếp cận giải dựa trên cây quyết định, mạng nơron hay phương pháp thống kê đã được đề xuất [2, 16]. Trong đó, mô hình dựa trên hệ luật mờ được nhiều tác giả áp dụng [1, 4, 6, 9-11] và đặc biệt hệ luật mờ dựa trên ĐSGT đem lại kết quả khả quan [14]. Bài toán phân lớp được phát biểu như sau: cho một tập các mẫu dữ liệu $D = \{(P; C)\}$, trong đó $P = \{p_i = (d_{i,1}, ..., d_{i,n}) | i = 1, ..., N\}$ là tập dữ liệu, $C = C_1, ..., C_m$ là tập các nhãn của các lớp, $p_i \in U$ là dữ liệu thứ i với $U = U_1 x ... x U_n$ là tích Đề-các của các miền của n thuộc tính $X_1, ..., X_n$ tương ứng, m là số lớp và N là số mẫu dữ liệu, để ý rằng $P \subset U$. Mỗi dữ liệu $p_i \in P$ được gán nhãn phân lớp $c_i \in C$ tương ứng tạo thành từng cặp $(p_i, c_i) \in D$. Thông thường miền của các thuộc tính là miền thực, tức là $U \subset R^n$.

Theo tiếp cận hệ luật mờ [9-11], giải bài toán tức là chúng ta xây dựng một hệ luật mờ

từ tập mẫu trên để đạt hiệu quả phân lớp cao đồng thời hệ luật đơn giản và dễ hiểu đối với người dùng. Thông thường lược đồ xây dựng hệ luật mờ phân lớp cho tập dữ liệu mẫu D gồm hai giai đoạn chính sau:

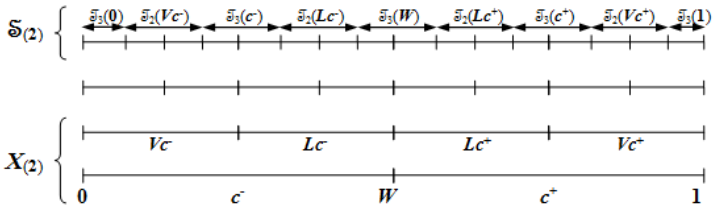
(B1) Phân hoạch mờ (fuzzy partition) trên miền của các thuộc tính dựa trên tập các giá trị ngôn ngữ của các biến ngôn ngữ - $Dom(X_i)$, mỗi giá trị ngôn ngữ được gán một hàm thuộc tương ứng.

(B2) Xác định một hệ các luật mờ từ các phân hoạch ở trên, mỗi luật mờ có dạng sau:

$$If\ X_1\ is\ A_{q1}\ and...and\ X_n\ is\ A_{qn}\ then\ Class\ C_q\ with\ CF_q, \tag{0.1}$$

trong đó $A_{q,j}$ là giá trị ngôn ngữ của các biến ngôn ngữ tương ứng với các thuộc tính, C_q là nhãn phân lớp và CF_q là trọng số của luật, $q = 1, ..., M$ với M là số luật, $j = 1, ..., n$ với n là số biến vào. Thông thường, trọng số của luật là số thực trong khoảng đơn vị, $CF_q \in [0, 1]$.

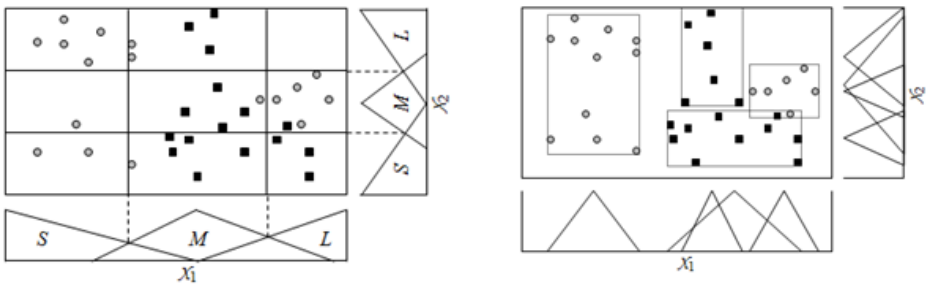
Bước phân hoạch mờ thường áp dụng 2 phương pháp gồm *grid-partition* và *scatter-partition* (hình vẽ 1.2). Trong AX^2 (là đại số gia tử hạn chế gồm 2 gia tử [15]), hệ khoảng tính mờ tương tự của một tập các hạng từ có độ dài từ 1 đến giới hạn $k_j, X_{(kj)}$, được xây dựng dựa trên hệ khoảng tính mờ mức k_{j+2}. Mỗi hạng từ $x \in X_{(kj)}$ sẽ xây dựng một khoảng tính mờ tương tự gồm hai khoảng tính mờ kề nhau mức $k_{j+2}, \Im_{kj+2}(z), \Im_{kj+2}(z') \in I_{k+2}$ chứa $\upsilon(x)$ làm điểm trong (hình vẽ 1.1). Chúng ta áp dụng phân hoạch mờ trên miền của mỗi thuộc tính dựa trên hệ các khoảng tính mờ tương tự này và nó là một phân hoạch rõ nên việc xây dựng hệ luật trong giai đoạn tiếp theo sẽ dễ dàng và nhanh.



Hình 1.1. Hệ khoảng tính mờ tương tự mức 2

Giai đoạn xác định hệ luật mờ thường rất khó khăn để đảm bảo tính hiệu quả cũng như tính đơn giản, dễ hiểu của hệ luật. Một số tác giả áp dụng các phương pháp tối ưu để xác định các luật [10, 14, 15], các tác giả trong [14, 4, 6] thiết kế các thuật toán tìm kiếm tối ưu tham số mờ để phân hoạch với mục tiêu đạt hiệu quả cao. Tuy nhiên các phương pháp tìm kiếm tối ưu tham số mờ thường làm mất tính ngữ nghĩa của các tập mờ, các tác giả đã phải đưa ra những ràng buộc nhất định. Với những tính chất của ĐSGT [7, 8, 13, 20] sẽ khắc phục được nhược điểm này và đặc biệt không gian các tham số mờ trong ĐSGT2 [15] sẽ giảm đi rất nhiều so với không gian các tập mờ, điều này làm tăng tốc độ tìm kiếm tối ưu của phương pháp.

Nói chung hai giai đoạn trên tương ứng là hai bài toán tối ưu và chúng không tách rời nhau vì mỗi kết quả phân hoạch sẽ sinh ra một hệ luật tối ưu khác nhau, hay bước xác định hệ luật mờ phụ thuộc vào bước phân hoạch mờ.



Hình 1.2. Phân hoạch mờ dạng grid và dạng scatter

Để giải bài toán tối ưu nhiều tác giả áp dụng phương pháp thích nghi dựa trên giải thuật di truyền (GA) [3, 4, 10]. Hơn nữa, bài toán tìm kiếm tham số mờ hay tìm kiếm hệ luật mờ đều là bài toán tối ưu đa mục tiêu. Bây giờ chúng ta xét bài toán tối ưu k mục tiêu phát biểu như sau:

$$F(x) = (f_1(x), f_2(x), ..., f_k(x)) \rightarrow max, \forall x \in X, \tag{0.2}$$

trong đó $F(x)$ là véc-tơ mục tiêu, $f_i(x)$ là mục tiêu thứ i cần cực đại, x là véc-tơ lời giải, và X là không gian lời giải của bài toán.

Trong [15], chúng tôi đã tiếp cận ĐSGT2 để thiết kế một thuật toán tìm kiếm hệ luật tối ưu trực tiếp trên lưới phân hoạch mờ. Do đó không gian tìm kiếm khá lớn với cả số lượng các luật và các thuộc tính tham gia vào điều kiện luật. Hơn nữa, công thức (1.2) được chuyển về dạng một mục tiêu bằng phép kết nhập các mục tiêu theo trọng số để áp dụng giải thuật di truyền đã làm thu hẹp khả năng tìm kiếm lời giải tối ưu, dẫn đến giảm hiệu quả có thể không cao. Trong bài này chúng tôi sẽ thiết kế một thuật toán tìm kiếm tối ưu Pareto nhằm khắc phục nhược điểm trên.

Theo phương pháp tối ưu Pareto dựa trên giải thuật di truyền NSGA-II [5, 18], một lời giải $x \in X$ được gọi trội hơn (dominate) lời giải $y \in X$ (tức là x tốt hơn y), ký hiệu $x \succ y$, nếu:

$$\forall i : f_i(y) \leq f_i(x), \exists j : f_j(y) < f_j(x). \tag{0.3}$$

Trong một tập lời giải, nếu không có bất kỳ một lời giải y trội hơn x thì ta gọi x là một lời giải tối ưu Pareto. Tập tất cả các lời giải như vậy gọi là tập tối ưu Pareto, ký hiệu S_p. Tập ảnh của lời giải tối ưu Pareto trên không gian mục tiêu được gọi là mặt Pareto (frontier). Một lời giải tối ưu Pareto x không thể nói tốt hơn một lời giải Pareto y [18], do đó chúng ta càng tìm ra nhiều lời giải tối ưu Pareto cho bài toán càng tốt. Các phương pháp tối ưu

cổ điển thực hiện chuyển bài toán đa mục tiêu về một mục tiêu bằng việc tập trung vào một mục tiêu tại một thời điểm và phải thực hiện nhiều lần phương pháp này để tìm tập các lời giải. Tuy đã có nhiều thuật toán tiến hóa giải bài toán tối ưu đa mục tiêu được giới thiệu [18], nhưng thuật toán NSGA-II được đề xuất bởi Deb và cộng sự trong [5] là một trong những thuật toán nổi tiếng và được các tác giả sử dụng [10]. NSGA-II có hai đặc trưng nổi bật làm tăng hiệu quả thực hiện cho bài toán tối ưu đa mục tiêu. Thứ nhất, đánh giá độ phù hợp của mỗi lời giải của bài toán (tức là cá thể trong quần thể) dựa trên xếp hạng Pareto và độ đo đám đông, thứ hai một thủ tục cập nhật thế hệ các cá thể tiên tiến được áp dụng.

Trong bài này chúng tôi áp dụng thuật toán NSGA-II để thiết kế phương pháp tìm kiếm tối ưu tham số mờ gia tử và tìm kiếm tối ưu hệ luật mờ phân lớp dựa trên ĐSGT2. Phần 2 trình bày một phương pháp sinh hệ luật mờ khởi đầu cho bài toán phân lớp dựa trên ĐSGT2, phần 3 và 4 sẽ thiết kế phương pháp tối ưu tham số mờ gia tử và tối ưu hệ luật mờ dựa trên thuật toán NSGA-II. Phần 5 áp dụng mô hình vào giải bài toán phân lớp đối với tập dữ liệu mẫu Yeast tại [19] và so sánh với kết quả của các phương pháp trong [11, 16].

2. PHƯƠNG PHÁP SINH HỆ LUẬT MỜ KHỞI ĐẦU DỰA TRÊN ĐSGT2

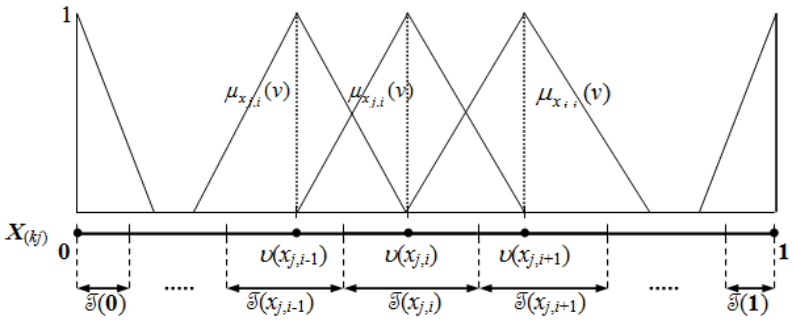
Theo tiếp cận ĐSGT và áp dụng phương pháp lưới phân hoạch mờ [14, 17], miền của mỗi thuộc tính sẽ được phân hoạch mờ dựa trên tập các giá trị ngôn ngữ trong ĐSGT2. Ta ký hiệu ĐSGT2 cho mỗi thuộc tính X_j là AX_j. Hệ khoảng tính mờ tương tự $S_{(kj)}$ là một phân hoạch của $[0,1]$, bằng cách chọn mức phân hoạch thích hợp đối với mỗi thuộc tính khi đó miền của mỗi thuộc tính được phân hoạch bởi $S_{(kj)}$ và tương ứng là tập giá trị ngôn ngữ $X_{(kj)}$. Mặt khác miền của các thuộc tính là miền thực, $U_j = [a_j, b_j] \in R$, chúng ta chuẩn hóa về miền đơn vị $[0,1]$ bằng các hàm chuyển như sau:

$$f_j(v) = \frac{v - a_j}{b_j - a_j}, j = 1, ..., n. \tag{0.4}$$

Trong ĐSGT2, sắp thứ tự ngữ nghĩa tập $X_{(kj)} = \{x_{j,0}, x_{j,1}, ..., x_{j,i-1}, x_{j,i}, x_{j,i+1}, ..., x_{j,1+2^{kj+1}}\}$, chúng ta gán hàm định lượng ngữ nghĩa cho mỗi giá trị ngôn ngữ $x_{j,i} \in X_{(kj)}$ với mong muốn càng gần tâm thì hàm định lượng càng lớn và đạt đỉnh tại tâm $\upsilon(x_{j,i})$, hàm này sẽ bằng 0 nếu vượt ra ngoài tâm của hai giá trị ngôn ngữ láng giềng của $x_{j,i}$ trong tập $X_{(kj)}$. Để đơn giản và tường minh khi ứng dụng xây dựng hệ luật, chúng ta thiết kế hàm dưới dạng tam giác như sau (hình vẽ 2.1):

$$\mu_{x_{j,i}}(v) = min(max(\frac{v - \upsilon(x_{j,i-1})}{\upsilon(x_{j,i}) - \upsilon(x_{j,i-1})}, 0), max(\frac{\upsilon(x_{j,i+1}) - v}{\upsilon(x_{j,i+1}) - \upsilon(x_{j,i})}, 0)), \tag{0.5}$$

trong đó $\upsilon(x)$ là giá trị định lượng của hạng từ ngôn ngữ x.



Hình 2.1. Hàm định lượng dạng tam giác của các hạng từ trong ĐSGT2

Phương pháp phân hoạch mờ dưới dạng lưới sẽ tạo nên một không gian phân hoạch trên miền của các thuộc tính gồm các siêu hộp, $H_S = \{(T^g(x_1, h_1), ..., T^g(x_n, h_n))|T^g(x_j, h_j) \in S_{(kj)}, j = 1, ..., n, h_j \in \{0, 1, ..., 1 + 2^{kj+1}\}\}$. Theo tính phân hoạch của hệ khoảng tính mờ tương tự, mỗi mẫu dữ liệu $p_i \in P$ xác định duy nhất một siêu hộp $B_i \in H_S$. Chúng ta chỉ xem xét sinh các luật từ những siêu hộp có chứa mẫu dữ liệu, do đó số luật tối đa được sinh là N trong trường hợp cực đoan, tức là bất kỳ hai mẫu dữ liệu đều không cùng thuộc một siêu hộp trong H_S. Trong ĐSGT2, mỗi siêu hộp trong H_S xác định một tập các giá trị ngôn ngữ trong $X_{(kj)}$ và luật mờ tương ứng được sinh ra với điều kiện vế trái là các giá trị ngôn ngữ này, $_q = (A_{q,1}, ..., A_{q,n}), A_{q,j} \in X_{(kj)}, j = 1, ..., n$. Phần kết luận vế phải của luật là nhãn phân lớp C_q xác định dựa trên độ tin cậy của luật như sau:

$$C_q = argmax_{C_h}\{c(A_q \Rightarrow C_q)|s(A_q \Rightarrow C_q) > 0, h = 1, ..., m\} \tag{0.6}$$

trong đó c và s là độ tin cậy và độ hỗ trợ của luật được tính theo luật kết hợp như sau:

$$c(A_q \Rightarrow C_q) = \frac{\Sigma_{p_i \in Class C_q}\mu_{A_q}(p_i)}{\Sigma_{i=1}^{N}\mu_{A_q}(p_i)}, s(A_q \Rightarrow C_q) = \frac{\Sigma_{p_i \in Class C_q}\mu_{A_q}(p_i)}{N}, \tag{0.7}$$

và mức độ đáp ứng đầu vào (đốt cháy) của luật đối với dữ liệu là

$$\mu_{A_q}(p_i) = \mu_{A_{q,1}}(d_{i,1}).\mu_{A_{q,2}}(d_{i,2})...\mu_{A_{q,n}}(d_{i,n}), \tag{0.8}$$

với $\mu_{A_{q,j}}(d_{i,j})$ được tính theo công thức (2.2).

Đối với các bài toán có số các thuộc tính lớn, để đảm bảo tính đơn giản và dễ hiểu đối với hệ luật mờ sinh ra và hơn nữa thực tế các thuộc tính có những vai trò khác nhau quyết định đến việc phân lớp, do đó chúng ta mong muốn các luật phân lớp sinh ra chỉ chứa điều kiện của một số ít các thuộc tính có vai trò lớn hơn. Theo tiếp cận của H. Ishibuchi và các cộng sự [Ish04], chúng ta sử dụng thêm một giá trị ngôn ngữ *"Don't Care (DC)"* trong phân hoạch để sinh luật, hàm thuộc của giá trị ngôn ngữ này đồng nhất bằng 1 trên miền của thuộc tính ($\mu_{DC}(v) = 1, \forall v$).

Lược đồ sinh luật dựa trên ĐSGT2 này giảm thiểu sự tính toán và xem xét đến các khả năng sinh luật từ không gian các phân hoạch H_S vì theo tính phân hoạch mỗi dữ liệu chỉ

xem xét duy nhất một lần để sinh luật, nhỏ hơn nhiều so với phương pháp của Ishibuchi có số khả năng sinh các luật là $|H_S|$. Tuy nhiên với sự bổ sung giá trị ngôn ngữ DC thì số khả năng sinh luật là khá lớn. Trong bài này chúng tôi sử dụng tiêu chuẩn để chọn luật là c.s cũng như hạn chế độ dài luật (số điều kiện tham gia ở vế trái) $L = 3$ và trọng số luật được xác định là:

$$CF(A_q \Rightarrow C_q) = c_q - c_{q,2nd}, \tag{0.9}$$

trong đó $c_{q,2nd}$ là độ tin cậy lớn nhất của các luật có cùng điều kiện A_q nhưng kết luận khác C_q:

$$c_{q,2nd} = max\{c(A_q \Rightarrow C_q)|h = 1, ..., m, C_h \neq C_q\}. \tag{0.10}$$

Thuật toán **IFRG(D, PAR, N0, L)** [17]: sinh hệ luật khởi đầu từ tập dữ liệu mẫu dựa trên ĐSGT2

Vào:

+ Tập mẫu $D = \{(p_i; c_i) : i = 1, ..., N\}, p_i = (d_{i,1}, ..., d_{i,n})$ là mẫu dữ liệu, $c_i \in \{C_1, ..., C_m\}$ là nhãn của mẫu dữ liệu tương tứng, m là số lớp và n là số thuộc tính,

+ Các tham số mờ gia tử và mức khoảng tính mờ tương tự cho mỗi thuộc tính: $PAR = \{fm_j(c^-), fm_j(c^+), m_j(L), m_j(V), k_j : j = 1, ..., n\}$,

+ Độ dài tối đa (số điều kiện tham gia trong vế trái luật) là L,

+ Số luật cần sinh N_0,

(để ý tiêu chuẩn sàng luật là $c.s$, trọng số luật xác định bởi công thức (2.6))

Ra: Tập các luật mờ S_0.

Các bước:

Step1) Tính hệ các khoảng tính mờ tương tự $S_{(kj)}$ để phân hoạch mờ trên miền của mỗi thuộc tính, xác định tập các giá trị ngôn ngữ từ độ dài 1 đến k_j cho $X_{(kj)}$,

Step2) Lặp trên mỗi mẫu dữ liệu $(p_i; c_i) \in D$ và thực hiện:

Step2.a) Với mỗi giá trị của thuộc tính đầu vào $d_{i,j} \in p_i$, xác định khoảng tính mờ tương tự trong $S_{(kj)}$ chứa $d_{i,j}, d_{i,j} \in \Im_{kj}(x_{kj,i*})$ và giá trị ngôn ngữ tương ứng $A_{i,j} = x_{kj,i*}$, với $j = 1, 2, ..., n$,

Step2.b) Tạo một tuyển điều kiện vế trái của luật gồm các giá trị ngôn ngữ vừa xác định ở trên

$$A_i = (A_{i,1}, A_{i,2}, ..., A_{i,n}), \tag{0.11}$$

Step2.c) Sinh các luật có điều kiện vế trái A_q lấy từ A_i với độ dài 1 đến L đưa, xác định phần kết luận vế phải theo công thức (2.3) và thêm vào tập luật khởi đầu nếu chưa có

$$S_0 = S_0 \cup \{A_q \Rightarrow C_q\}, \tag{0.12}$$

trong đó $A_q \subset A_i$ và $|A_q| \leq 3$ và

$$C_q = argmax_{C_k}\{c(A_q \Rightarrow C_h)|s(A_q \Rightarrow C_h) > 0, h = 1, ..., m\}. \quad (0.13)$$

Step3) Sắp tập luật S_0 theo thứ tự giảm của tiêu chuẩn sàng $(c.s)$ theo nhóm các luật có nhãn phân lớp giống nhau và chọn $\lfloor N_0/m \rfloor$ luật đầu tiên trong mỗi nhóm ($\lfloor \bullet \rfloor$ là phép lấy phần nguyên của $\bullet$).

Mệnh đề 2.1. [17] *Độ phức tạp của thuật toán IFRG là đa thức đối với kích thước và số chiều của tập dữ liệu mẫu D.*

Chứng minh. Dễ dàng nhận thấy thời gian tính toán phân hoạch mức kj cho các thuộc tính ở bước 1 là $O(n.|X_{(k*)}|)$, trong đó $k* = max\{k_j : j = 1, ..., n\}$. Thông thường chúng ta có $|X_{(k*)}| \ll |D|$ (*).

Trong bước 2, thời gian tính toán để sinh tuyển các điều kiện vế trái là $O(n.|D|)$. Trường hợp cực đoan kích thước của tập tuyển các điều kiện vế trái là $|D|$, tức mỗi mẫu dữ liệu xác định một tuyển vế trái. Mỗi tuyển vế trái được chọn để sinh luật với độ dài từ 1 đến L, mỗi luật được tính toán độ tin cậy và độ hỗ trợ theo lần lượt mỗi kết luận là một nhãn phân lớp trong tập dữ liệu. Do đó thời gian tính toán là $O(|D|^2.\sum_{\lambda=1...L} C_n^\lambda) = O(nL.|D|^2)$ (**).

Bước 3 sắp các luật theo thứ tự giảm của tiêu chuẩn, trường hợp cực đoan tất cả các luật cùng thuộc một nhóm nên thời gian bị giới hạn bởi:

$$O(|P|.\sum_{1\leq\lambda\leq L} C_n^\lambda.log_2(|P|.\sum_{1\leq\lambda\leq L} C_n^\lambda)) =$$
$$O(|P|.log_2|P|.\sum_{1\leq\lambda\leq L} C_n^\lambda) + O(|P|.\sum_{1\leq\lambda\leq L} C_n^\lambda.log_2(\sum_{1\leq\lambda\leq L} C_n^\lambda)\ (***).$$

Từ (*), (**) và (***) ta có độ phức tạp của thuật toán $IFRG(D, PAR, N_0, L)$ là:

$$O(n^L.|P|^2) + O(n^L.log_2 n.|P|) \Rightarrow \text{đpcm.}$$

3. THIẾT KẾ THUẬT TOÁN TỐI ƯU PARETO TẬP LUẬT MỜ PHÂN LỚP

Với bộ tham số mờ gia tử PAR được cho hoặc đã được tối ưu đối với một bài toán phân lớp, sử dụng thuật toán $IFRG$ sinh tập luật S_0 kích thước đủ lớn. Tuy nhiên tập S_0 có thể chứa nhiều luật dư thừa và tồn tại các luật mẫu thuẫn nhau vì tiêu chuẩn sàng được chọn chưa thể đảm bảo yếu tố này. Một phương pháp nhiều tác giả sử dụng là tìm kiếm hệ luật tối ưu dựa trên tập S_0 [10, 11, 14], sử dụng thuật toán di truyền. Trong bài này chúng tôi áp dụng thuật toán tối ưu Pareto dựa trên giải thuật di truyền (GA) - NSGA-II của Deb [5]. Thuật toán được thiết kế để chọn một tập luật S S0 sao cho đạt được những mục tiêu hiệu quả phân lớp cao, số luật ít và luật càng đơn giản càng tốt. Rõ ràng đây là bài toán tối ưu có 3 mục tiêu (MOP3), có thể phát biểu như sau:

$$f_p(S) \to max, f_n(S), f_a(S) \to min, \quad (0.14)$$

với ràng buộc $S \in S_0, |S| \leq N_{opt}$.

Trong đó $f_p(S)$ là tỷ lệ phân lớp đúng của hệ S trên tập mẫu luyện (training patterns), $f_n(S)$ là số luật và $f_a(S)$ là độ dài luật trung bình của hệ S. Nopt là số luật chọn tối đa và được cho trước.

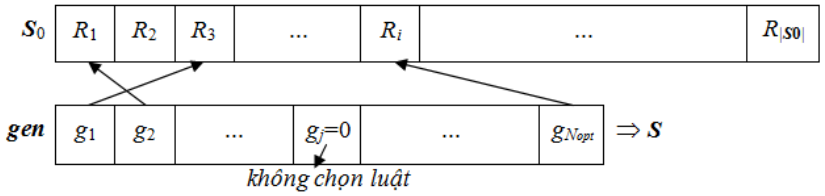
Mục tiêu cực đại hiệu quả phân lớp là yêu cầu tự nhiên đối với bất kỳ hệ luật mờ nào được xây dựng. Hai mục tiêu cực tiểu số luật và độ dài các luật nhằm mục đích làm đơn giản hệ luật được xây dựng, tạo tính dễ hiểu đối với người dùng. Rõ ràng khi $f_n(S)$ giảm thì nguy cơ hiệu quả phân lớp $f_p(S)$ cũng giảm, độ dài luật $f_a(S)$ giảm tức số điều kiện quyết định đến phân lớp giảm dẫn đến miền quyết định tạo bởi phân hoạch mờ của các điều kiện đó được mở rộng (xem hình 1.2) và có nguy cơ phân lớp sai đối với các mẫu dữ liệu không thuộc lớp. Hơn nữa, phương pháp sinh luật theo cách lấy tổ hợp số các điều kiện vế trái (thuật toán IFRG) thì số điều kiện giảm có nghĩa số luật tăng theo số tổ hợp được lấy. Như vậy, việc đạt được ba mục tiêu trên một cách đồng thời là không thể, chúng ta sẽ phải thỏa hiệp trong các phương pháp tối ưu.

Hai mục tiêu sau đối với $f_n(S)$ và $f_a(S)$ của bài toán được chuyển về dạng cực đại bằng phép lấy nghịch đảo, điều này luôn đảm bảo vì $f_n(S) > 0$ và $f_a(S) > 0$. Do đó mục tiêu của bài toán trở về dạng sau:

$$F(S) = (f_p(S), f_n(S)^{-1}, f_a(S)^{-1}) \rightarrow max. \tag{0.15}$$

Sử dụng phương pháp mã hóa số thực, mỗi lời giải ứng với một tập luật S được chọn từ S_0 được biểu diễn bởi một chuỗi gen được gọi là một cá thể $s_i = (g_1, ..., g_{N_{opt}}), g_j \in [0, 1]$. Giá trị gen g_j xác định luật có chỉ số là $g_j.|S_0|$ trong tập $S_0 (0 \leq g_j.|S_0| \leq |S_0|)$, nếu chỉ số này bằng 0, tức $g_j = 0$ thì luật tương ứng không chọn, điều này dẫn đến $|S| \leq N_{opt}$ (hình vẽ 3.1).

$$S = \{R_i \in S_0 | i = \lfloor g_j.|S_0| \rfloor, i > 0\} \tag{0.16}$$



Hình 3.1. Sơ đồ mã hóa cá thể lựa chọn tập luật S từ S_0

Trong NSGA-II, mỗi quần thể gồm một tập các lời giải $P = \{S_i : i = 1, ..., N_{pop}\}, N_{pop}$ là kích thước quần thể, sẽ được sắp Pareto vào các tập frontier F_k dựa trên quan hệ chi phối giữa các cá thể ($\succ$, xem công thức (1.3)). Tập F_k gồm các cá thể bị chi phối bởi $k - 1$ cá thể khác trong P. Ký hiệu D_p là tập các cá thể bị chi phối bởi cá thể p, M_p là số lượng các

cá thể chi phối cá thể p trong quần thể. Thủ tục tính toán và sắp Pareto được trình bày dưới dạng ngôn ngữ giả lập trình như sau:

Thủ tục NDS(P) [5]: sắp Pareto tập các cá thể trong pop vào các tập frontier F_k.

$D_p = \emptyset, M_p = 0$ for all $p \in P$

for each $p \in P$
  for each $q \in P$
    if ($p \succ q$) then
      $D_p = D_p \cup \{q\}$
    else if ($q \succ p$) then
      $M_p = M_p + 1$
    end if
    end if
  end for
  if ($M_p = 0$) then
    $F_1 = F_1 \cup \{p\}$
  end if
end for
$k = 1$
while ($F_k \neq \emptyset$)
  $H = \emptyset$
  for each $p \in F_k$
    for each $q \in D_p$
      $M_q = M_q - 1$
      if ($T_q = 0$) then
        $H = H \cup \{q\}$
      end if
    end for
  end for
  $k = k + 1$
  $F_k = F_k \cup H$
end while.

Trong mỗi tập F_k, chúng ta tính toán giá trị ước lượng độ trù mật của các thể dựa trên khoảng cách của giá trị mỗi mục tiêu giữa các cá thể đó. Ký hiệu $F_k[i]$ là cá thể thứ i trong tập $F_k, F_k[i].m$ là giá trị hàm mục tiêu thứ m trong các mục tiêu của $F(S)$ tại (3.2), $F_k[i].dist$ là giá trị ước lượng độ trù mật cần tính.

Thủ tục DE(F_k) [5]: tính toán giá trị ước lượng độ trù mật các cá thể trong mỗi frontier.

$l = |F_k|$

$F_k[i].dist = 0$, for all i
for each objective m in $F(S)$
sort(F_k, m)
$F_k[1].dist = F_k[l].dist = \infty$
for $i = 2$ to $l - 1$
$F_k[i].dist = F_k[i].dist + (F_k[i+1].m - F_k[i-1].m)$
end for
end for.

Hàm sort(F_k, m) để sắp tập các cá thể trong F_k theo giá trị hàm mục tiêu thứ m.

Trong một quần thể chúng ta định nghĩa quan hệ thứ tự bộ phận ($\geq_d$) của các cá thể dựa trên chỉ số tập frontier của cá thể, với mỗi $p \in F_k$ ta ký hiệu $p.rank = k$, và giá trị ước lượng độ trù mật là $p.dist$. Khi đó với hai cá thể p và q thì

$$p \geq_d q \; if \; (p.rank < q.rank) \; or \; ((p.rank = q.rank) \; and \; (p.dist > q.dist)). \tag{0.17}$$

Thuật toán FROPT(S_0, N_{opt}): tìm kiếm hệ luật tối ưu Pareto dựa trên giải thuật di truyền (GA)

Vào:
+ Hệ luật khởi đầu S_0 sinh bởi thuật toán IFRG,
+ Số luật tối đa cần chọn tối ưu N_{opt},
+ Các tham số cho GA: xác suất lai ghép β_c, xác suất đột biến β_m, tham số lai ghép α_c, tham số đột biến α_m, kích thước quần thể mỗi thế hệ N_{pop} và số thế hệ cần tiến hóa G_{max}.

Ra: Tập tối ưu các luật mờ S_{opt}.

Các bước:

Step 1) Khởi tạo một quần thể ngẫu nhiên $P_0 = \{p_i | i = 1, ..., N_{pop}\}$, N_{pop} là độ lớn của quần thể ở mỗi thế hệ, đặt chỉ số thế hệ $g = 0$,

Step 2) Xác định hệ luật S_i tương ứng với mỗi cá thể $p_i \in P_g$.

Step 3) Tính các giá trị của các mục tiêu trong (3.2), $f_p(S_i), f_n(S_i), f_a(S_i)$ cho các cá thể $p_i \in P_g$.

Step 4) Sắp Pareto các cá thể trong P_g vào các tập F_k (sử dụng thủ tục $NDS(P_g)$),

Step 5) Sinh quần thể con Q_g từ P_g bằng các phép toán di truyền [14], sử dụng giá trị hàm đánh giá độ phù hợp theo chỉ số tập $F_k, fitness = rank^{-1}$, để áp dụng phép chọn lọc, lai ghép và đột biến.

Step 6) Tính hợp hai quần thể Q_g và P_g ở thế hệ hiện tại, $R = Q_g \cup P_g$.

Step 7) Sắp Pareto các cá thể trong R vào các tập F_k (sử dụng thủ tục $NDS(R)$),

Step 8) Lặp từ $k = 1$ trên các tập F_k cho đến khi $|P_{g+1}| \geq N_{pop}$, thực hiện:
(i) Tính toán độ trù mật của các cá thể trong F_k theo thủ tục $DE(F_k)$,
(ii) Tính hợp $P_{g+1} = P_{g+1} \cup F_k$.

Step 9) Sắp giảm dần thứ tự các cá thể trong quần thể P_{g+1} theo quan hệ ed.

Step 10) Chọn Npop cá thể đầu trong quần thể $P_{g+1}, P_{g+1} = P_{g+1}[0, ..., N_{pop}]$.

Step 11) Đặt $g = g + 1$ và lặp lại Step 2) cho đến khi đạt thế hệ G_{max}.

Step 12) Trả về kết quả là hệ luật S xác định bởi cá thể có $f_p(S)$ lớn nhất, nếu cùng $f_p(S)$ thì $f_n(S)$ nhỏ nhất, nếu cùng $f_p(S)$ và $f_n(S)$ thì $f_a(S)$ nhỏ nhất.

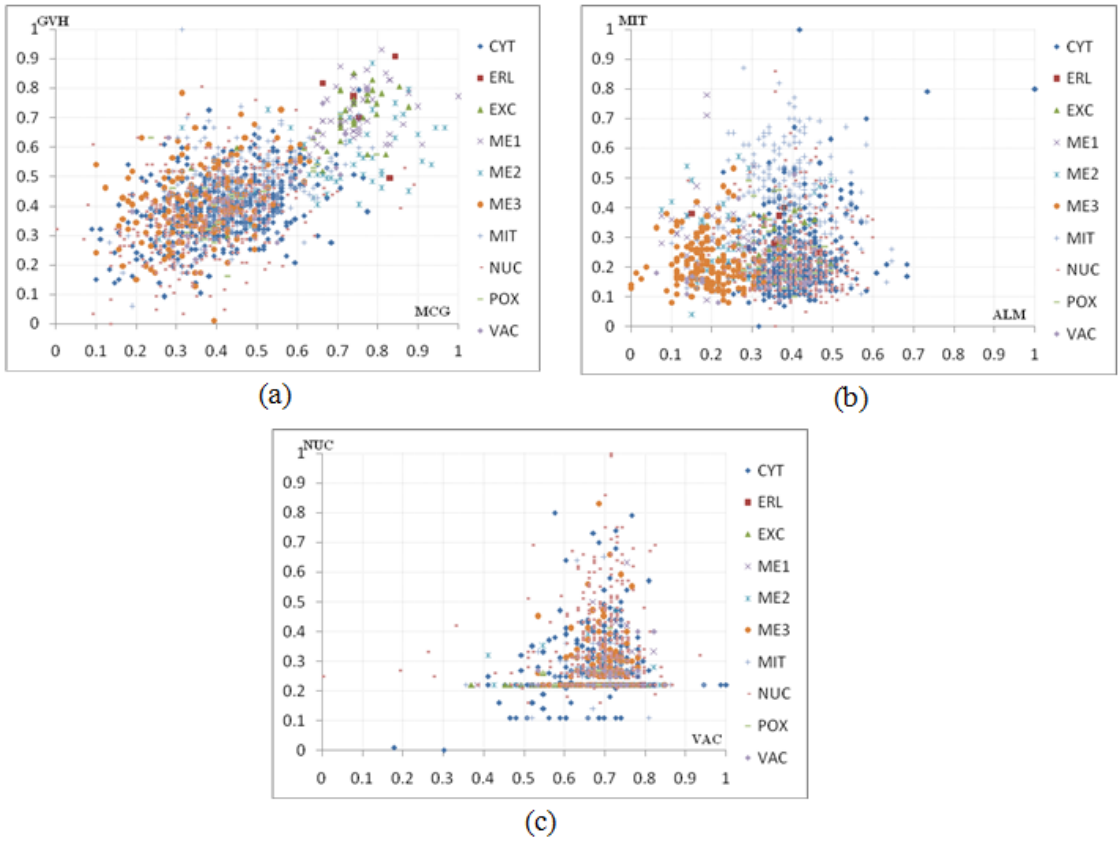
Tiếp theo chúng ta áp dụng phương pháp xây dựng hệ luật mờ phân lớp trên để giải bài toán phân lớp các loại men sinh học (Yeast).

4. ÁP DỤNG THỬ NGHIỆM VÀO BÀI TOÁN PHÂN LỚP

Tập dữ liệu mẫu cho bài toán phân lớp các loại men (Yeast) do giáo sư K. Nakai thu thập tại Viện phân tử và tế bào sinh học, Đại học Osaka, Nhật Bản, và được công bố trong [19]. Nhiều tác giả nghiên cứu đã sử dụng tập dữ liệu này để thử nghiệm các mô hình cho bài toán phân lớp [16, 12, 11]. Tập dữ liệu gồm 1484 mẫu chia thành 10 lớp và có 8 thuộc tính là MCG, GVH, ALM, MIT, ERL, POX, VAC, NUC. Để đơn giản ta ký hiệu X_j là các thuộc tính, j = 1,...,8. Bảng 4.1 thể hiện phân bố số lượng các mẫu dữ liệu theo từng lớp, hình 4.1 thể hiện sự phân bố dữ liệu của các lớp theo từng cặp thuộc tính: (4.1-a) cho cặp thuộc tính MCG và GVH, (4.1-b) cho cặp thuộc tính ALM và MIT, (4.1-c) cho cặp thuộc tính VAC và NUC. Đối với cặp thuộc tính ERL và POX có hầu hết các mẫu dữ liệu bằng 0 hoặc 1. Đồ thị phân bố dữ liệu cho thấy bài toán rất phức tạp, các mẫu dữ liệu ở các lớp chồng chéo lên nhau, hầu như không có thuộc tính nào thể hiện tính trội hơn hẳn để phân lớp. Hơn nữa, số lượng mẫu trong tập dữ liệu khá lớn cùng với sự phân bố các mẫu dữ liệu không cân bằng nhau, tỷ số chênh lệch phân bố này lên đến 463/5. Những thách thức không nhỏ đối với bất kỳ mô hình phân lớp nào.

Bảng 4.1. Phân bố số lượng các mẫu dữ liệu trong mỗi lớp

Lớp	Mô tả	Số mẫu dữ liệu
CYT	Cytosolic or cytoskeletal	463
NUC	Nuclear	429
MIT	Mitochondrial	244
ME3	Membrane protein, no N-terminal signal	163
ME2	Membrane protein, uncleaved signal	51
ME1	Membrane protein, cleaved signal	44
EXC	Extracellular	37
VAC	Vacuolar	30
POX	Peroxisomal	20
ERL	Endoplasmic reticulum lumen	5



Hình 4.1. Biểu đồ phân bố dữ liệu của các thuộc tính theo lớp

Áp dụng cấu trúc ĐSGT2 cho miền các giá trị ngôn ngữ của các thuộc tính X_j, ký hiệu là $AX_j, j = 1, ..., 8$. Bộ tham số gia tử cùng với mức phân hoạch k_j hệ các khoảng tính mờ tương tự trong AX_j được thiết lập trong bảng sau (để ý rằng $fm(c^+) = 1 - fm(c^-), m(V) = 1 - m(L)$):

Bảng 4.2. Các tham số gia tử và mức phân hoạch hệ các khoảng tính mờ tương tự

Thuộc tính	MCG (X_1)	GVH (X_2)	ALM (X_3)	MIT (X_4)	ERL (X_5)	POX (X_5)	VAC (X_7)	NUC (X_8)
$fm_j(c^-)$	0.64	0.73	0.52	0.39	0.51	0.32	0.47	0.48
$\mu_j(L)$	0.21	0.68	0.46	0.68	0.55	0.28	0.51	0.55
k_j	1	1	2	1	2	1	2	1

Chúng tôi thử nghiệm mô hình cho bài toán theo sơ đồ *10-folds*, chia ngẫu nhiên tập dữ liệu mẫu thành 10 phần bằng nhau, lấy ra một phần để kiểm tra còn lại 9 phần dùng để xây dựng hệ luật bằng cách áp dụng liên tiếp thuật toán IFRG để sinh hệ luật khởi đầu, thuật toán FROPT để tìm hệ luật tối ưu. Lặp lại quá trình chia tập mẫu và xây dựng hệ luật này 20 lần. Thời gian trung bình sinh hệ luật khởi đầu của thuật toán IFRG nhỏ hơn nhiều so với [11], giảm 86.67%, tức thuật toán này chỉ mất 2 giây trong khi của [11] mất 15 giây.

Bảng 4.3. Kết quả thử nghiệm mô hình trong trường hợp 10-folds

Mô hình	PN	PL	PA	PT
Ishibuchi [11]	37.85	2.83	63.77	56.93
Pavlidis [16]	-	-	-	58.26
Our method	34.70	2.45	62.25	59.12

Kết quả cuối cùng của các lần thử nghiệm sau khi tìm kiếm hệ luật tối ưu được tính trung bình đối với số luật sinh ra (PN), độ dài trung bình hệ luật (PL), tỷ lệ phân lớp đúng trong tập mẫu học (PA) và tập kiểm tra (PT), thể hiện trong bảng 4.3 (ký hiệu '-' là không có kết quả thử nghiệm). So sánh với phương pháp của [11], kết quả của chúng tôi có số luật giảm 8.32%, độ dài trung bình của luật giảm 13.57%, tỷ lệ phân lớp đúng trên tập huấn luyện giảm 2.38% trong khi tỷ lệ phân lớp đúng trên tập kiểm tra tăng 3.85%. So với [16] thì tỷ lệ phân lớp đúng trên tập kiểm tra của chúng tôi tăng 1.52%.

5. KẾT LUẬN

Trong bài này chúng tôi đã xây dựng một phương pháp tìm kiếm hệ luật mờ tối ưu sử dụng thuật toán NSGA-II, một phương pháp tối ưu Pareto đa mục tiêu bằng giải thuật di truyền. Tập luật khởi đầu được sinh từ dữ liệu mẫu bằng cách áp dụng thuật toán IFRG dựa trên hệ phân hoạch các khoảng tính mờ tương tự của tập các giá trị ngôn ngữ trong ĐSGT2, các giá trị ngôn ngữ này tạo nên phân hoạch mờ trên miền của các thuộc tính. Thuật toán IFRG áp dụng để sinh hệ hệ luật khởi đầu với số lượng đủ lớn nhằm tránh sự mất mát thông tin đối với bài toán. Hơn nữa, các luật mờ được sinh trực tiếp từ các mẫu dữ liệu nên không gian các luật cần xem xét trong quá trình tính toán giảm. Trong khi đó phương pháp trong [11] sẽ xem xét hết mọi khả năng sinh luật có thể, sau đó sử dụng tiêu chuẩn giới hạn độ tin cậy và độ hỗ trợ để loại bớt. Điều này đòi hỏi khối lượng thời gian tính toán rất lớn.

Thuật toán tìm kiếm tối ưu Pareto dựa trên giải thuật di truyền (FROPT) được thiết kế với mục tiêu tìm hệ luật sao cho càng đơn giản, dễ hiểu và hiệu quả phân lớp cao. Chúng tôi sử dụng thuật toán khá nổi tiếng của Deb [5] là NSGA-II. Ứng dụng vào bài toán phân lớp các loại men sinh học, một bài toán rất khó đối với các mô hình phân lớp, đạt kết quả khả quan khi so sánh với một số phương pháp khác (bảng 4.3), minh họa cho tính hiệu quả của phương pháp này. Ngoài ra, chúng ta có thể thiết kế thuật toán tìm kiếm bộ tham số tối ưu trước khi thực hiện xây dựng và tìm kiếm hệ luật để tăng kết quả của mô hình (xem [14]).

TÀI LIỆU THAM KHẢO

[1] Johannes A. Roubos, Magne Setnes, Janos Abonyi, Learning fuzzy classification rules from labeled data, *Information Sciences* **150** (2003) 77–93.

[2] Diyar Akay, M. Ali Akcayol, Mustafa Kurt, NEFCLASS based extraction of fuzzy rules and classification of risks of low back disorders, *Expert Systems with Applications* **35** (2008) 2107–2112.

[3] Ulrich Bodenhofer, *Genetic Algorithms: Theory and Applications, Lecture Notes*, Third Edition-Winter 2003/2004.

[4] Chia-Chong Chen, Design of PSO-based fuzzy classification systems, *Tamkang Journal of Science and Engineering* **9** (1) (2006) 63–70.

[5] Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and T Meyarivan, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, *Parallel Problem Solving from Nature - PPSN VI*, Springer, 2000 (849–858).

[6] A. Fernandez, M. Calderon, F. Herrera, Enhancing fuzzy rule based systems in multi-classification using pairwise coupling with preference relations, *EUROFUSE Workshop Preference Modelling and Decision Analysis*, Public University of Navarra, 2009.

[7] Nguyen Cat Ho, A topological completion of refined hedge algebras and a model of fuzziness of linguistic terms and hedges, *Fuzzy Sets and Systems* **158** (2007) 436–451.

[8] Nguyễn Cát Hồ, CSDL mờ với ngữ nghĩa đại số gia tử, "Lectures on the Fuzzy Systems & Applications Autumn School", Viện toán học Việt Nam, 9/2008.

[9] Hisao Ishibuchi and Takashi Yamamoto, Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining, *Fuzzy Sets and Systems* **141** (1) (2004) 59–88.

[10] Hisao Ishibuchi, Yusuke Nojima, Analysis of interpretability-accuracy tradeo of fuzzy systems by multiobjective fuzzy genetics-based machine learning, *International Journal of Approximate Reasoning* **44** (1) (2007) 4–31.

[11] H. Ishibuchi, Y. Nojima and I. Kuwajima, Parallel distributed genetic fuzzy rule selection, *Soft Computing - A Fusion of Foundations, Methodologies and Applications, SpringerLink* **13** (5) (2009) 511–519.

[12] Huan Liu, Rong Jin, A novel approach to model generation for heterogeneous data classification, *19th International Joint Conference on AI (IJCAI-05)*, Edinburgh, Scotland, 2005.

[13] Nguyen Cat Ho, Nguyen Van Long, Fuzziness measure on complete hedge algebras and quantifying emantics of terms in linear hedge algebras, *Fuzzy Sets and Systems* **158** (2007) 452–471.

[14] Nguyễn Cát Hồ, Trần Thái Sơn, Dương Thăng Long, Tiếp cận đại số gia tử cho phân lớp mờ, *Tạp chí Tin học và Điều khiển học*, **25** (1) (2009) 53–68.

[15] Nguyễn Cát Hồ, Trần Thái Sơn, Dương Thăng Long, Đại số gia tử hạn chế AX2 (ĐSGT2) và ứng dụng cho bài toán phân lớp mờ , *Tạp chí Khoa học và Công nghệ* (2010).

[16] N.G. Pavlidis, V.L. Georgiou, K.E. Parsopoulos, Alevizos, M.N. Vrahatis, Optimizing the Performance of Probabilistic Neural Networks in a Bionformatics Task, *Proceedings of the EUNITE 2004 Conference*, Aachen, Germany, 2004 (pages 34–40).

[17] Witold Pedrycz, Nguyen Cat Ho, Duong Thang Long, Tran Thai Son, Fuzzy Rule Extraction for Classification Problems Using Hedge Algebra-Based Semantics of Vague Terms, manuscript for submitting to *Information Sciences*, 2010.

[18] C. R. Rao, O. A. Jadaan, L. Rajamani, Non-dominated ranked genetic algorithm for solving multi-objective optimization problems: NRGA, *Journal of Theoretical and Applied Information Technology* **4** (1) (2008).

[19] UC Irvine Machine Learning Repository, http://archive.ics.uci.edu/ml.

[20] Nguyen Cat Ho, Vu Nhu Lan, Le Xuan Viet, Optimal hedge-algebras-based controller: Design and application, *Fuzzy Sets and Systems* **159** (2008) 968–989.

Nhận bài ngày 2 - 3 - 2010
Nhận lại sau sửa ngày 24 - 4 -2010