Tạp chí Tin học và Điều khiển học, T.28, S.4 (2012), 333–345

MỘT PHƯƠNG PHÁP TIẾN HÓA SINH HỆ LUẬT MỜ CHO BÀI TOÁN PHÂN LỚP VỚI NGỮ NGHĨA THỨ TỰ NGÔN NGỮ*

NGUYỄN CÁT HỒ[1], HOÀNG VĂN THÔNG[2], NGUYỄN VĂN LONG[2]

[1] *Viện Công nghệ thông tin, Viện Khoa học và Công nghệ Việt Nam*
[2] *Trường Đại học Giao thông Vận tải Hà Nội*

Tóm tắt. Bài báo đề xuất một phương pháp thiết kế hệ luật mờ cho bài toán phân lớp có tham khảo ý tưởng đề xuất trong [17] để sinh hệ luật khởi sinh. Việc sinh luật trong nghiên cứu này dựa trên phương pháp học tiến hóa tối ưu đa mục tiêu [10-15] và phương pháp đại số gia tử [1-8] và đề xuất một tiêu chuẩn đánh giá độ thích nghi của các luật dựa trên việc cải tiến tiêu chuẩn đánh giá đề xuất trong [17]. Các từ ngôn ngữ được sử dụng trong việc sinh các luật mờ được thiết kế dựa trên đại số gia tử và thuật giải tiến hóa dựa trên việc tối ưu hóa các tham số mờ của đại số gia tử. Phương pháp đề xuất được thử nghiệm trên 9 bài toán phân lớp điển hình được công bố trên UCI [18] cho hiệu quả phân lớp cao mà vẫn bảo đảm tính dễ hiểu của hệ luật.

Abstract. In this paper, we propose a method to design fuzzy rule-based systems for classification problems with reference to the idea proposed in [17] to generate initial rule-based system. The generation of rules in this method is based on evolutionary multi-objective optimization [10-15] and hedge algebra methods [1-8] and proposed a rule evaluation measure bases on the improvement of the rule evaluation measure proposed in [17]. The linguistic terms used to generate fuzzy rules are designed based on hedge algebra and the generic algorithms based on optimizing the fuzzy parameters of hedge algebra. The proposed method is tested on 9 typical problems published in the UCI [18] with high performance classification while ensuring the rule-based system easy understanding.

1. GIỚI THIỆU

Hệ mờ dựa trên luật đã có những ứng dụng thành công trong trong nhiều lĩnh vực khác nhau, trong đó có lĩnh vực điều khiển, phân lớp trong những năm gần đây. Bài toán phân lớp dữ liệu là một trong những bài toán được nhiều nhà khoa học nghiên cứu giải quyết. Nhiều phương pháp đã đề xuất để giải bài toán này như mạng nơ ron, cây quyết định, hệ mờ dựa trên luật. Trong những thập niên gần đây đã có nhiều đề xuất giải bài toán phân lớp bằng hệ luật mờ [6-8,10-17] và đã cho kết quả khá tốt so với các phương pháp khác. Một số nghiên cứu gần đây tập trung vào xây dựng các thuật toán học tự động để sinh hệ luật mờ dựa trên các thuật giải tiến hóa tối ưu đa mục tiêu. Các thuật toán đề xuất nhằm tạo ra các hệ luật mờ thỏa mãn các mục tiêu: tỉ lệ phân lớp chính xác cao, số luật ít, chiều dài trung bình của

*Nghiên cứu được hoàn thành dưới sự hỗ trợ từ Quỹ phát triển khoa học và Công nghệ quốc gia (NAFOSTED) mã số 102.01-2011.06

các luật ngắn và dễ hiểu với con người. Các nghiên cứu đã đề xuất chủ yếu tập trung vào việc đề xuất các thuật toán sinh hệ luật mờ với số luật ít nhất và tỉ lệ phân lớp đạt cao nhất có thể được.

Nhiều phương pháp đã đề xuất cho kết quả hệ luật có tỉ lệ phân lớp chính xác cao, nhưng hệ luật sinh ra không dễ hiểu đối với con người, do tập mờ sử dụng không phải là ngôn ngữ tự nhiên như trong [20]. Một số phương pháp như trong [10-17] sử dụng một tập các từ ứng với các tập mờ dạng tam giác cố định cho việc sinh luật, các luật sinh ra dễ hiểu với người dùng hơn nhưng vẫn bị hạn chế do giá trị ngôn ngữ chỉ là nhãn được người thiết kế gán cho dựa trên cảm nhận trực giác. Ngoài ra, việc dùng cùng một tập các từ với các tập mờ cố định trước cho các bài toán khác nhau là không phù hợp với thực tế, vì trong thực tế ngữ nghĩa của các từ phụ thuộc vào bài toán cụ thể và chúng được lựa chọn để sử dụng phụ thuộc vào ngữ cảnh dữ liệu cụ thể. Để khắc phục nhược điểm này nhóm tác giả trong [6-8] đã ứng dụng ĐSGT để trích rút các giá trị ngôn ngữ tích hợp cùng các tập mờ của chúng cho từng bài toán trong việc xây dựng hệ luật. Kết quả hệ luật sinh ra đảm bảo được tính dễ hiểu, đảm bảo ngữ nghĩa của các tập mờ phụ thuộc vào các đặc trưng dữ liệu của từng bài toán. Mặc dù phương pháp đề xuất trong [6-8] đã khắc phục được các nhược điểm về ngữ nghĩa của các tập mờ nhưng thuật toán xây dựng hệ luật phải xem xét một lượng lớn các luật ứng cử ($m \times C_n^{l_{\max}}$ luật với bài toán có n thuộc tính, độ dài luật tối đa là $l_{\max}$ và m mẫu dữ liệu).

Trong [17] đề xuất một thuật giải di truyền (SGERD) sinh hệ luật ứng cử nhỏ đồng thời kết hợp với tiêu chuẩn sàng luật để tìm ra hệ luật tối ưu. Phương pháp này có ưu điểm là số lượng các luật ứng cử sinh ra ít, số thế hệ của thuật giải di truyền hữu hạn. Tuy nhiên việc lựa chọn hệ luật tối ưu cuối cùng chỉ theo kinh nghiệm (theo tiêu chuẩn độ thích nghi cao nhất) và đặc biệt là tập các tập mờ tam giác được gán nhãn ngôn ngữ cố định đã được sử dụng cho nhiều công trình nghiên cứu khác nhau để sinh luật. Như vậy, chúng không được điều chỉnh thích nghi để khai thác các đặc trưng riêng biệt của các bài toán khác nhau nên khó có thể nâng cao hiệu quả phân lớp chính xác.

Bài báo đề xuất một phương pháp với thuật toán OPHA-SGERD được phát triển bằng cách kết hợp phương pháp xây dựng hệ luật SGERD với phương pháp đề xuất trong [6-8], được nghiên cứu dựa trên cách tiếp cận của nhóm H. Ishibuchi. Thuật toán OPHA-SGERD thực hiện việc sinh hệ luật ứng cử, sau đó chúng tôi xây dựng thuật toán tìm hệ luật tối ưu HA-OFRB theo hàm mục tiêu đề xuất trong [15] với việc cải tiến tiêu chuẩn chọn luật trong [17]. Thuật toán đề xuất phải xem xét một số ít các luật hơn so với [6-8] và có khả năng tạo ra các hệ luật có tỉ lệ phân lớp cao, dễ hiểu với con người và ổn định với các bài toán khác nhau. Cũng giống [6-8], việc trích rút các giá trị ngôn ngữ được tích hợp với tập mờ tam giác cũng là một thành phần của phương pháp OPHA-SGERD được đề xuất.

Bài báo được bố cục như sau: Mục 2 tổng quan về thiết kế hệ phân lớp mờ dựa trên các luật mờ và ĐSGT. Mục 3 bàn về các tiêu chuẩn đánh giá luật, Mục 4 đề xuất phương pháp sinh hệ luật dựa trên ĐSGT và Mục 5 phương pháp tối ưu tham số của ĐSGT. Việc nghiên cứu thử nghiệm và đánh giá phương pháp đề xuất đươc trình bày trong Mục 6, và cuối cùng là phần kết luận.

2. TỔNG QUAN THIẾT KẾ HỆ PHÂN LỚP VỚI LUẬT MỜ NGỮ NGHĨA ĐSGT

Các luật mờ dạng if-then cho một bài toán phân lớp mẫu với n thuộc tính có thể viết như sau:

Luật R_j: If x_1 is A_{j1} and ... and x_n is A_{jn} Then C_j với $j = 1..N$ (2.1)

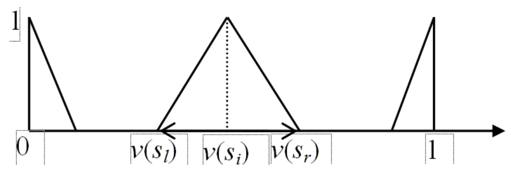
trong đó x_i là biến ngôn ngữ (thuộc tính) và A_{ji} $(i = 1, ..., n)$ là các nhãn ngôn ngữ của các điều kiện trong tiền đề được lấy trong $X_{(k_j)}$ của một ĐSGT AX_j ứng với thuộc tính j, C_j là tên lớp kết luận của R_j và N là số luật mờ. Vấn đề đặt ra là xây dựng một phương pháp cho phép thiết kế hệ phân lớp mờ dựa trên một tập N luật mờ có dạng (2.1) được trích rút từ một tập mẫu phân lớp D có M lớp với m mẫu dữ liệu có n thuộc tính được gán nhãn $d_p = [d_{p1}, d_{p2}, ..., d_{pn}, C_p], p = 1, ..., m$. Trong đó tập nhãn các lớp là $C = \{C_j : j = 1, ..., M\}$.

Một số kết quả nghiên cứu gần đây đều tập trung vào việc xây dựng hệ luật dựa trên việc xem xét tất cả các tổ hợp có thể có của các giá trị ngôn ngữ làm tiền đề [10-15], mỗi một tổ hợp tạo ra một luật. Theo hướng tiếp cận này, khi bài toán có nhiều thuộc tính thì số luật sinh ra có kích cỡ hàm mũ theo số thuộc tính ($\sum_{l=1}^{n} K^l C_n^l$ luật với bài toán có n thuộc tính, sử dụng K giá trị ngôn ngữ cho mỗi thuộc tính). Một hướng tiếp cận khác được đề xuất trong [6-8] sử dụng ĐSGT, phân chia không gian mẫu dựa trên hệ khoảng tương tự để tạo thành các siêu hộp. Thực hiện sinh ra các luật có độ dài $l \le n$ từ mỗi mẫu dữ liệu dựa trên các siêu hộp chứa mẫu dữ liệu đó. Trên cơ sở tập luật có độ dài n sinh ra các luật có độ dài ngắn hơn bằng cách bỏ đi một số điều kiện tiền đề của luật có độ dài n. Khi đó số luật tối đa phải xem xét là $m \times \sum_{l=1}^{l\max} C_n^l$ luật, trong đó m là số mẫu dữ liệu, $l\max$ là chiều dài tối đa của luật. Phương pháp sinh luật này đã làm giảm số luật ứng cử, tuy nhiên số lượng vẫn còn lớn. Trong phần 4 sẽ trình bày một phương pháp xây dựng hệ luật ứng cử với số luật sinh ra giảm đáng kể so với các phương pháp nêu trên.

Lớp kết luận C_j của luật R_j trong (2.1) được xác định bằng cách tính độ đốt cháy của các mẫu dữ liệu huấn luyện. Độ đốt cháy luật R_j với phần tiền đề $A_j = A_{j1} \times A_{j2} \times \cdots \times A_{jn}$ được xác định như sau:

$$\mu_{A_j}(d_p) = \prod_{i=1}^{n} \mu_{ji}(d_{pi}), \tag{2.2}$$

trong đó $\mu_{ji}(\cdot)$ là hàm thuộc của tập mờ của giá trị ngôn ngữ $A_{ji} \in X_{(k_j)}$ (tập tất cả các từ có độ dài $\le k$). Trong ĐSGT với mỗi từ ngôn ngữ s_i có một khoảng tính mờ tương ứng là $[v(s_l), v(s_r)]$, trong đó s_l, s_r lần lượt là các từ có cùng độ dài liền kề bên trái và bên phải của từ $s_i; v(s_l), v(s_r)$ lần lượt là giá trị định lượng ngữ nghĩa của từ ngôn ngữ s_l và $s_r, v(s_i)$ là giá trị định lượng ngữ nghĩa của từ ngôn ngữ s_i. Khi đó ta xây dựng hàm thuộc $\mu_{ji}(s)$ có dạng tam giác như sau:



$$\mu_{ji}(s) = \begin{cases} \max(\dfrac{s - v(s_l)}{v(s_i) - v(s_l)}, 0) & s < v(s_i) \\ \max(\dfrac{s - v(s_r)}{v(s_i) - v(s_r)}, 0) & s \ge v(s_i) \end{cases}$$

Khi đó lớp kết luận C_j của luật R_j được xác định theo công thức (2.3) dưới đây

$$C_j = arg\max\{conf(A_j \Rightarrow C_h)|C_h = 1, 2, ..., M\}, \tag{2.3}$$

trong đó,

$$conf(A_j \Rightarrow C_h) = \frac{\sum_{d_p \in C_h} \mu_{A_j}(d_p)}{\sum_{d_1}^{m} \mu_{A_j}(d_p)}$$

là độ tin cậy của luật.

Phương pháp lập luận trên hệ luật được sử dụng phổ biến nhất là phương pháp single-winer rule, do nó dễ hiểu với con người. Theo phương pháp này một mẫu dữ liệu $d_p = [d_{p1}, d_{p2}, ..., d_{pn}]$ được phân lớp theo luật có độ đốt cháy cao nhất R_w. Nó được phát biểu như sau:

$$\mu_w(d_p) = \max\{\mu_{A_j}(d_p),\ j = 1, 2, ..., N\} \tag{2.4}$$

trong đó $\mu_{A_j}(d_p)$ là mức độ đốt cháy luật R_j của mẫu dữ liệu d_p được xác định theo (2.2).

3. CÁC TIÊU CHUẨN ĐÁNH GIÁ LUẬT

Các thuật toán xây dựng hệ mờ phân lớp, thường sử dụng các tiêu chuẩn đánh giá luật với mục đích chọn ra được các luật có khả năng làm tăng hiệu quả phân lớp của hệ luật, do đó làm giảm bớt số luật của hệ cho phù hợp với yêu cầu đọc dễ hiểu của người dùng. Một số tiêu chuẩn được thừa kế từ lĩnh vực khai phá luật kết hợp, đó là độ hỗ trợ và độ tin cậy của luật.

Tiêu chuẩn dựa trên độ tin cậy và độ hỗ trợ của luật

$$f_{cs}(A_j \Rightarrow C_j) = \frac{(\sum_{d_p \in ClassC_j} \mu_{A_j}(d_p))^2}{\sum_{p=1}^{m} \mu_{A_j}(d_p)}. \tag{3.1}$$

Một số thuật toán học máy đơn giản trong môi trường mờ (SLAVE) sử dụng tiêu chuẩn đánh giá

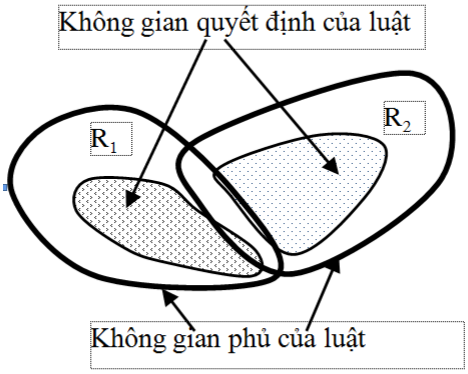
$$f_F(A_j \Rightarrow C_j) = \sum_{d_p \in ClassC_j} \mu_{A_j}(d_p) - \sum_{d_p \notin ClassC_j} \mu_{A_j}(d_p). \tag{3.2}$$

trong đó $\sum_{d_p \in ClassC_j} \mu_{A_j}(d_p)$, $\sum_{d_p \notin \mu_{A_j}(d_p)}$ lần lượt là tổng độ đốt cháy luật của những mẫu dữ liệu mà luật đoán nhận đúng và không đúng.

Mỗi một luật mờ luôn xác định 2 không gian con, không gian phủ và không gian quyết định của luật. Không gian phủ của luật bao gồm tất cả các mẫu dữ liệu đốt cháy luật, không gian quyết định bao gồm những mẫu dữ liệu được phân lớp bởi luật. Không gian phủ của các luật kề nhau có thể giao nhau. Vì vậy một số mẫu dữ liệu có thể đốt cháy một số luật kề nhau, khi đó không thể xác định được chính xác không gian quyết định của luật. Để giải quyết tình huống này trong [16,17] đề xuất ngưỡng τ_j cho luật R_j, khi đó các mẫu dữ liệu có độ đốt cháy luật lớn hơn τ_j thì nó được cho là thuộc về không gian quyết định của R_j. Từ đó trong [16,17] đề xuất tiêu chuẩn đánh giá

$$f'_F(A_j \Rightarrow C_j) = f_F(A_j \Rightarrow C_j) + \gamma_j(1 - \tau_j), \tag{3.3}$$

trong đó γ_j là số mẫu dữ liệu có độ đốt cháy luật R_j cao hơn τ_j, τ_j là tham số ngưỡng được xác định như sau $\tau_j = 0.5^{l_j}$ với l_j là chiều dài của luật R_j.



Hình 3.1. Minh họa không gian phủ và không gian quyết định của luật

Ta thấy việc xây dựng hàm xác định ngưỡng $\tau_j = 0.5^{l_j}$ là không phù hợp với các bài toán khác nhau, vì sự phân bố của các mẫu dữ liệu là khác nhau trong mỗi bài toán. Do đó tỉ lệ giao nhau giữa các không gian phủ của các luật trong các bài toán khác nhau là khác nhau. Vì vậy, tham số ngưỡng để quyết định một mẫu dữ liệu thuộc về không gian quyết định của luật nào đó phải phụ thuộc vào từng bài toán. Chúng tôi đề xuất tiêu chuẩn đánh giá luật (3.4) dựa trên (3.3) với tham số ngưỡng là một hàm phụ thuộc tham số α, trong đó $\alpha \in (0, 1)$ và được xác định tùy theo từng bài toán, hay tập mẫu, bằng giải thuật di truyền.

$$f'_F(A_j \Rightarrow C_j) = f_F(A_j \Rightarrow C_j) + \gamma_j(1 - \alpha^{l_j}), \tag{3.4}$$

Trong (3.4), γ_j là số mẫu dữ liệu có độ đốt cháy luật R_j lớn hơn α^{l_j}. Ta thực hiện xác định tham số α tối ưu cho từng bài toán được tiến hành đồng thời với quá trình xác định các tham số mờ của ĐSGT đối với bài toán đó.

4. THUẬT TOÁN TOÁN TIẾN HÓA SINH HỆ LUẬT OPHA-SGERD

Trong phần này sẽ mô tả thuật toán OPHA-SGERD để sinh hệ luật. Với mỗi thuộc tính thứ j của bài toán ta sử dụng một tập $X_{(k_j)} \cup \{Don'tcare\}$ các từ ngôn ngữ để sinh ra điều kiện tiền đề thứ j của luật, trong đó $Don'tcare$ là tập mờ có $\mu_{Don'tcare}(x) = 1$.

Thuật toán được thiết kế dựa trên ý tưởng của thuật toán tiến hóa trong [17]. Mỗi cá thể trong quần thể là một luật mờ R_r có n biến điều kiện tiền đề và lớp kết luận trong $C, R_r : A_r \rightarrow C_r$, trong đó $A_r = (A_r[1], ..., A_r[n]), A_r[j]$ là biến điều kiện tiền đề thứ j của luật và nhận giá trị trong tập $X_{(k_j)} \cup \{Don'tcare\}$. Gọi các biến điều kiện tiền đề có giá trị $Don'tcare$ là các biến không hoạt động, các biến điều kiện tiền đề có giá trị thuộc $X_{(k_j)}$ là các biến hoạt động. Tại mỗi thế hệ, từ tập luật ứng cử chọn ra Q luật có độ đo thích nghi cao nhất trên mỗi lớp làm quần thể hiện tại. Từ quần thể hiện tại tạo sinh ra các luật con có chiều dài lớn hơn cha mẹ của chúng một đơn vị. Những luật con có độ đo thích nghi cao hơn cha mẹ của chúng cùng với các luật cha mẹ được giữ lại làm tập luật ứng cử cho thế hệ kế tiếp. Quần thể khởi tạo là tập các luật có độ dài 1 được sinh ra từ tổ hợp tất cả các khả năng của các giá trị ngôn ngữ $X_{(k_j)}$.

Môt số ký hiệu: R là tập luật ứng cử cho quần thể kế tiếp trong quá trình tiến hóa; R là tập các luật của quần thể hiện tại, R' là tập luật bổ trợ cho quá trình tạo sinh; $R(C_j), \Re(C_j), R'(C_j)$

là các tập luật có cùng lớp kết luận C_j; $|R(C_j)|, |\Re(C_j)|$ lần lượt là số luật của tập luật $R(C_j)$ và $\Re(C_j)$; $R_r.fitness$ và $R_r.Class$ là độ đo thích nghi và lớp kết luận của luật R_r.

Hàm SORT($\Re(C_j)$) thực hiện sắp xếp các luật trong tập luật $\Re(C_j)$ theo độ đo thích nghi của luật giảm dần.

Algorithm OPHA-SGERD (D, M, n, Q, X, f)
Input
- Tập mẫu dữ liệu $D = \{(d_p, C_p) : p = 1..m\}$, số lớp kết luận M, số thuộc tính n
- Số luật Q được chọn ứng với mỗi lớp, tập các tập từ ngôn ngữ $X = \{X_{(k_j)} : j = 1..n\}$
- Hàm độ đo thích nghi f của luật (chọn trước 1 trong 4 hàm (3.1), (3.2), (3.3) hoặc (3.4))

Output
- Tập luật mờ R có tối đa $M \times Q$ luật

Method Phương pháp tiến hóa sinh hệ luật
Begin

```
Khởi tạo ℜ bằng tập luật rỗng;
for j ← 1 to n do
  for each x in X_(k_j) do
    Sinh luật R_r có A_r[i] = Don'tcare với i = 1..n, i ≠ j và A_r[j] = x;
    R_r.Class ⟵ arg max{conf(A_r ⟹ C_h)|} ; //xác định lớp kết luận
    R_r.fitness ⟵ f(R_r); //tính độ đo thích nghi của luật R_r
    ℜ ⟵ ℜ ∪ {R_r};
  endfor
endfor
g ⟵ 1;
Khởi tạo R' bằng tập luật rỗng; //R' là tập bổ trợ sử dụng trong quá trình tạo sinh
repeat
  for j ⟵ 1 to M do
    Khởi tạo ℜ(C_j) bằng tập luật rỗng;//ℜ(C_j) để lưu các luật trong R có vế phải
    là C_j;
    for each R_r in ℜ do
      if R_r.Class = C_j then ℜ(C_j) ⟵ ℜ(C_j) ∪ {R_r};
     endfor
    SORT(ℜ(C_j)); //sắp xếp tập luật ℜ(C_j) theo độ đo thích nghi của luật giảm dần
  endfor
  Khởi tạo R bằng tập luật rỗng;
  for j ⟵ 1 to M do
    if g > 1 then
      Đặt R(C_j) gồm min{Q, |ℜ(C_j)|} luật đầu tiên của ℜ(C_j) đã được sắp;
    else
      if |ℜ(C_j)| ≥ 2 * Q then
        Đặt R(C_j) gồm Q luật đầu tiên của ℜ(C_j);
        ℜ(C_j) ⟵ R(C_j) \ R(C_j);
        Đặt R'(C_j) gồm Q luật đầu tiên của ℜ(C_j); //xác định tập bổ trợ R' ứng
        với lớp C_j //
      else
        Đặt R(C_j) gồm |ℜ(C_j)|/2 luật đầu tiên của ℜ(C_j);
```

$\Re(C_j) \longleftarrow \Re(C_j) \setminus R(C_j)$;
$R'(C_j) \longleftarrow \Re(C_j)$; //xác định tập bổ trợ R' ứng với lớp C_j
endif
$R' \longleftarrow R' \cup R'(C_j)$;
endif
$R \longleftarrow R \cup R(C_j)$;
endfor
$g \longleftarrow g+1$;
if $g \leq n$ **then** $\Re \longleftarrow REPRODUCE(R, R', P, M, n, X, f)$; //thực hiện thuật toán tạo sinh
until $(g > n)$ **or** ($\Re$ **trùng** R)**;**
return R;
End

Thuật toán tạo sinh $REPRODUCE(R, R', D, M, n, X, f)$, thực hiện tạo sinh trên các luật của quần thể R và quần thể bổ trợ R' để tạo ra tập luật ứng cử của thế hệ kế tiếp bao gồm các luật của R và các luật thế hệ con có độ đo thích nghi cao hơn cha mẹ sinh ra chúng.

Với mỗi luật R_j có lớp kết luận C_j của R thực hiện tao sinh trên luật R_j như sau: chọn ngẫu nhiên một luật R_p trong số các luật có cùng lớp kết luận C_j của R. Nếu luật R_p trùng với luật R_j thì chọn lại luật R_p bằng cách chọn ngẫu nhiên một luật trong các luật có cùng lớp kết luận C_j của R'. Tiếp theo chọn ngẫu nhiên một biến x_i trong số các biến hoạt động của của luật R_p. Kiểm tra, nếu biến x_i tương ứng trong luật R_j có giá trị là $Don'tcare$ thì lần lượt sinh ra các luật con của R_j bằng cách thay thế giá trị $Don'tcare$ của biến x_i của luật R_j bằng một giá trị ngôn ngữ trong tập $X_{(k_i)}$. Xác định lớp kết luận của các luật mới sinh theo (2.3) và giá trị độ đo thích nghi theo hàm (3.1), (3.2), (3.3) hoặc (3.4) được chọn. Các luật có độ đo thích nghi cao hơn độ đo thích nghi của luật R_j được giữ lại làm luật ứng cử cho thế hệ kế tiếp. Nếu biến x_i tương ứng trong luật R_j có giá trị khác $Don'tcare$ thì không thực hiện tạo sinh trên luật R_j.

Hàm $RANDOM(R, C_j)$ thực hiện chọn ngẫu nhiên một luật trong các luật có cùng lớp kết luận C_j của tập luật R, hàm $RANDOMACTIVE(R_p)$ thực hiện chọn ngẫu nhiên chỉ số của biến trong số các biến hoạt động của luật R_p.

Algorithm $REPRODUCE(R, R', D, M, n, X, f)$
Input:
- Tập luật mờ R, tập luật mờ bổ trợ R', tập mẫu dữ liệu $D = \{(d_p, C_p) : p = 1..m\}$
- Số lớp kết luận M, số thuộc tính n, tập các tập từ ngôn ngữ $X = \{X_{(k_j)} : j = 1..n\}$
- Hàm độ đo thích nghi f của luật (chọn 1 trong 4 hàm (3.1), (3.2), (3.3) hoặc (3.4))

Output:
- Tập luật mờ $\Re$

Method: Tạo sinh hệ luật ứng cử cho quần thể kế tiếp
Begin
$\Re \longleftarrow R$;
for each R_j **in** R **do**
$R_p \longleftarrow RANDOM(R, C_j)$;
if R_p **trùng** R_j **then** $R_p \longleftarrow RANDOM(R', C_j)$;
$i \longleftarrow RANDOMACTIVE(R_p)$;
if $A_j[i] = Don'tcare$ **then** // A_j là vế trái của luật R_j

for each x **in** $X_{(k_i)}$ **do**
Sinh luật R_r có $A_r[t] = A_j[t]$ với $t = 1..n$;
$A_r[i] \longleftarrow x$;//A_r là vế trái của luật R_r
$R_r.Class \longleftarrow arg\max\{conf(A_r \Longrightarrow C_h)|C_h = 1, 2, ..., M\}$
$R_r.fitness \longleftarrow f(R_r)$; //tính độ đo thích nghi của luật
if $R_r.fitness > R_j.fitness$ **then** $\Re \leftarrow \Re \cup \{R_r\}$;//bổ sung luật R_r vào $\Re$
endfor
endif
endfor
return $\Re$;
End

5. TỐI ƯU THAM SỐ MỜ ĐSGT VÀ TỐI ƯU HỆ LUẬT

5.1. Tối ưu tham số mờ ĐSGT

Để nâng cao hiệu quả phân lớp của hệ luật thì các tập mờ phải được điều chỉnh thích nghi với từng bài toán. Trong nghiên cứu này, các tập mờ được xây dựng dựa trên các từ ngôn ngữ được thiết kế dựa trên ĐSGT. Vì vậy để điều chỉnh các tập mờ thì ta phải thực hiện điều chỉnh các tham số mờ của ĐSGT tức là đi tìm các tham số mờ tối ưu để làm tăng hiệu quả phân lớp của hệ luật.

Sử dụng ĐSGT có 2 gia tử (V, L) cho tất cả các bài toán, vì vậy việc tối ưu tham số của ĐSGT cho mỗi bài toán là đi tìm các bộ tham số tối ưu $op = \{(o\mu^j_{fmc-}, o\mu^j_L, ok_j, o\alpha) : j = 1..n\}$ với hàm độ đo thích nghi (3.4) hoặc $op = \{(o\mu^j_{fmc-}, o\mu^j_L, ok_j, o\alpha) : j = 1..n\}$ với các hàm còn lại.

Để tìm tham số mờ tối ưu của ĐSGT, ta thực hiện thiết kế một thuật giải di truyền dựa trên sơ đồ mã hóa nhị phân với hàm muc tiêu $perf(R, D)$ là hiệu quả phân lớp của hệ luật R được sinh ra từ thuật toán OPHA-SGERD trên tập mẫu dữ liệu D với phương pháp Test-All. Các toán tử đột biến, lại ghép và lựa chọn quần thể kế tiếp được thừa kế trong [9].

Algorithm $OP - PARHA(D, M, n, Q, f, PAR, Npop, Gmax, Pmu, Pcross, lchrom)$
Input:
- Tập mẫu dữ liệu $D = \{(d_p, Cp) : p = 1..m\}$, số lớp kết luận M, số thuộc tính n
- Số luật Q được chọn ứng với mỗi lớp
- Hàm độ đo thích nghi f của luật (chọn 1 trong 4 hàm (3.1), (3.2), (3.3) hoặc (3.4))
- Tập các ràng buộc tham số mờ của ĐSGT

$$PAR = \{(\min\mu^j_{fmc-}, \max\mu^j_{fmc-}, \min\mu^j_L, \max\mu^k_L, \min k_j, \max k_j) : j = 1..n\}$$

- $Npop$ là số cá thể trong quần thể, $Gmax$ là số thế hệ tiến hóa, Pmu là xác xuất đột biến, $Pcross$ là xác xuất lai ghép, $lchrom$ là chiều dài gen

Output:
- Bộ tham số tối ưu $op = \{(o\mu^j_{fmc-}, o\mu^j_L, ok_j, o\alpha) : j = 1..n\}$ với hàm độ đo thích nghi (3.4) hoặc $op = \{(o\mu^j_{fmc-}, o\mu^j_L, ok_j) : j = 1..n\}$ với các hàm còn lại.

Method: Phương pháp tối ưu tham số mờ của ĐSGT
Begin
$i \longleftarrow 0$;
Khởi tạo quần thể $POP_0 = \{p_{0,1}, ..., p_{0,Npop}\}$;

$op.objvalue \longleftarrow 0$; //khởi tạo giá trị hàm mục tiêu của biến lưu bộ tham số tối ưu
repeat
for each p **in** POP_i **do**
Xây dựng tập các tập từ ngôn ngữ $X = \{X_{(k_j)} : j = 1..n\}$ bằng ĐSGT với bộ tham số mờ p;
$R \longleftarrow OPHA - SGERD(D, M, n, Q, X, f)$; //xây dựng hệ luật R với tập các tập từ ngôn ngữ X
$p.objvalue \longleftarrow perf(R, D)$; //tính giá trị hàm mục tiêu của cá thể p
if $op.objvalue < p.objvalue$ **then** $op \longleftarrow p$**;**
endfor
$i \longleftarrow i + 1$;
if $i \leq gmax$ **then**
Thực hiện lai ghép, đột biến;
Lựa chọn quần thể thế hệ thứ i (thế hệ kế tiếp): POP_i;
endif
until $i > Gmax$;
return *op*; //trả lại bộ tham số tối ưu
End

5.2. Thuật toán HA-OFRB tối ưu hệ luật

Thuật toán OPHA-SGERD sinh ra hệ luật có tối đa $M \times Q$ luật theo phương pháp Test-All. Thuật toán sinh ra hệ luật với số luật khá lớn $M \times Q$ luật, để đảm bảo tính dễ hiểu của hệ luật thì ta cần tìm một hệ luật con S tối ưu trong $M \times Q$ luật, sao cho S có hiệu quả phân lớp chính xác cao nhất có thể, có số luật ít nhất và tổng chiều dài các luật của hệ luật là nhỏ nhất (tức là S thỏa mãn hàm đa mục tiêu: $f_{mo}(S) = \{\max f_p(S), \min f_n(S), \min f_l(S)\}$, trong đó $f_p(S)$ là hiệu quả phân lớp chính xác của hệ luật S, $f_n(S)$ là số luật trong S, $f_l(S)$ là tổng chiều dài của các luật trong S). Đây là bài toán tối ưu đa mục tiêu, để giải quyết vấn đề này, trong [15] H.Ishibuchi và các đồng nghiệp đã đề xuất giải pháp thay thế hàm đa mục tiêu $f_{mo}(S)$ bằng hàm một mục tiêu $f(S) = \max\{w_p f_p(S) - w_n f_n(S) - w_l f_l(S)\}$ với w_p, w_n, w_l là các tham số cho trước có giá trị trong khoảng $(0, 1)$ và thỏa mãn $w_p + w_n + w_l = 1$. Như vậy việc tìm kiếm hệ luật S tối ưu trở thành bài toán tối ưu hàm một mục tiêu $f(S)$.

Để tìm hệ luật tối ưu, ta xây dựng một giải thuật di truyền để làm việc này, với sơ đồ mã hóa nhị phân và hàm mục tiêu $f(S)$. Với số luật tối đa $R_{\max}$ cho trước khi đó mỗi cá thể của quần thể có $R_{\max}$ biến mục tiêu là $r_1, ..., r_{R\max}$. Giá trị của các biến mục tiêu được giới hạn trong khoảng [0,1]. Khi đó luật có chỉ số $[r_i \times |R|]$ với $i = 1..R_{\max}$ trong hệ luật R sẽ được chọn vào hệ luật tối ưu, trong đó $|R|$ là số luật của hệ luật R. Ở đây, ta sử dụng toán tử đột biến, lại ghép và lựa chọn quần thể kế tiếp trong [9]. Do thuật giải di truyền tương tự như trong mục 5.1 nên không trình bày lại ở đây.

6. THỬ NGHIỆM VÀ ĐÁNH GIÁ

Trong phần này sẽ thử nghiệm và so sánh hiệu quả phân lớp của thuật toán OPHA-SGERD trên 9 bài toán được lấy từ UCI [18] với thuật toán SGERD trong[17]. Phương pháp thử nghiệm Ten-fold, với các tiêu chuẩn đánh giá luật (3.2), (3.3) và (3.4). Kết quả thực nghiệm lần lượt trình bày trong bảng 6.3, 6.4 và 6.5.

Quá trình thử nghiệm được thực hiện với hai pha. Pha thứ nhất tìm các tham số tối ưu của ĐSGT, thực nghiệm với các tham số của giải thuật di truyền như sau: xác suất lai ghép $Pcross = 0.85$, xác suất đột biến $Pmu = 0.05$, giới hạn các biến mục tiêu $0.3 \leq \mu^j_{fmc-} \leq 0.7$, $0.3 \leq \mu^j_L \leq 0.7$, $1 \leq k_j \leq 3$ với $j = 1..n, 0.3 \leq \alpha \leq 0.7$, chiều dài mỗi gen $lchrom = 24$, số cá thể 250 và số thế hệ tiến hóa 150, tham số của thuật toán OPHA-SGERD $Q = 20$, chiều dài tối đa của luật ≤ 3, phương pháp thử nghiệm sinh luật Test-All. Pha thứ hai tìm hệ luật tối ưu từ hệ luật R được sinh ra từ thuật toán OPHA-SGERD. Thực hiện thuật toán HA-OFRB với các tham số: $R_{\max}$ được giới hạn với giá trị xấp xỉ bằng số luật của hệ luật công bố trong [17], chi tiết mô tả trong bảng 6.2, xác suất lai ghép $Pcross = 0.85$, xác suất đột biến $Pmu = 0.05$, giới hạn các biến mục tiêu $0 \leq r_i \leq 1$ với $i = 1..R_{\max}$, chiều dài mỗi gen $lchrom = 24$, số cá thể 250, số thế hệ tiến hóa 500, tham số hàm mục tiêu $w_p = 0.99, w_n = 0.001, w_l = 1 - w_p - w_n$.

Kí hiệu: $\#Na$ số thuộc tính, $\#Nc$ số lớp, $\#Np$ số mẫu dữ liệu, $\#Nar$ trung bình số luật, $\#Nal$ trung bình chiều dài luật, $Perf$ tỉ lệ phân lớp chính xác.

Bảng 6.1. Các bài toán thử nghiệm

Bài toán	*#Na*	*#Nc*	*#Np*
Cancer	9	2	684
Glass	9	6	214
Iris	4	3	178
Pima	8	2	768
Sonar	60	2	208
Wine	13	3	178
Image	19	7	210
Vowel	10	11	990
Yeast	8	10	1484

Bảng 6.2. Các giá trị của $R_{\max}$ trong quá trình tối ưu hệ luật

Bài toán	Hàm đánh giá luật		
	(3.2)	(3.3)	(3.4)
Cancer	6	5	5
Glass	12	12	12
Iris	5	5	5
Pima	7	8	8
Sonar	6	6	6
Wine	8	7	7
Image	12	14	14
Vowel	30	34	34
Yeast	23	22	22

Bảng 6.3. So sánh kết quả thử nghiệm thuật toán OPHA-SGERD và thuật toán SGERD với hàm đánh giá (3.2)

Bài toán	*Perf (%)*		*#Nar*		*#Nal*	
	SGERD	OPHA-SGERD	SGERD	OPHA-SGERD	SGERD	OPHA-SGERD
Cancer	96.29	**96.42**	5.38	6	1.17	1.33
Glass	62.90	**68.22**	11.52	12	**1.85**	2.83
Iris	**96.93**	**96.67**	4.00	4	**1.01**	1.25
Pima	74.64	**77.34**	6.12	7	**1.42**	1.43
Sonar	77.20	**82.21**	4.29	6	1.14	1.50
Wine	95.52	**96.07**	7.12	8	1.39	2.13
Image	83.52	**86.19**	11.44	11	2.18	2.45
Vowel	49.68	**51.72**	30	29	**3.04**	**3.03**
Yeast	49.84	**53.77**	22.36	20	2.85	2.90

Từ các bảng kết quả cho thấy, tỉ lệ phân lớp chính xác cao hơn của thuật toán OPHA-SGERD cao hơn so với thuật toán SGERD. Bảng 6.3 cho thấy thuật toán OPHA-SGERD có tỉ lệ phân lớp chính xác cao hơn thuật toán SGERD trên 8 bài toán, trong đó có một số bài toán có tỉ lệ cao hơn khá nhiều như bài toán Glass là 5.32%, Sonar là 5.0% và Image là 2.67%.

Bảng 6.4 So sánh kết quả thử nghiệm thuật toán OPHA-SGERD và thuât toán SGERD với hàm đánh giá (3.3)

Bài toán	*Perf (%)*		*#Nar*		*#Nal*	
	SGERD	OPHA-SGERD	SGERD	OPHA-SGERD	SGERD	OPHA-SGERD
Cancer	**97.02**	96.42	**3.96**	5	2.31	**2.40**
Glass	63.38	**73.36**	10.22	11	**2.13**	2.45
Iris	96.40	**97.33**	4.30	5	1.95	**1.80**
Pima	73.08	**76.95**	7.76	8	7.18	**2.50**
Sonar	75.20	**79.81**	5.96	5	5.17	**3.80**
Wine	**96.19**	**96.63**	6.14	7	3.56	**2.43**
Image	86.10	**86.76**	9.28	14	4.56	**2.57**
Vowel	**58.53**	55.25	33.78	30	3.88	**2.57**
Yeast	**56.53**	54.18	21.50	20	5.50	**2.95**

Bảng 6.5. So sánh kết quả thử nghiệm thuật toán OPHA-SGERD với hàm đánh giá (3.3) và (3.4)

Bài toán	*Perf (%)*		*#Nar*		*#Nal*	
	(3.3)	**(3.4)**	**(3.3)**	**(3.4)**	**(3.3)**	**(3.4)**
Cancer	96.42	96.42	**5**	6	2.40	**1.83**
Glass	71.96	**73.83**	12	**11**	**2.58**	3.73
Iris	97.33	97.33	5	**4**	1.80	**1.50**
Pima	76.95	**77.34**	8	**7**	2.50	**2.29**
Sonar	**79.81**	78.85	5	5	**3.80**	4.40
Wine	96.63	**97.19**	7	8	**2.43**	2.75
Image	**86.76**	86.19	14	**12**	**2.57**	4.00
Vowel	55.25	**57.37**	30	34	**2.57**	3.44
Yeast	54.18	**55.73**	20	**19**	**2.95**	3.53

Bảng 6.6. So sánh kết quả thử nghiệm thuật toán OPHA-SGERD với các tiểu chuẩn đánh giá khác nhau

Bài toán	*Perf (%)*		
	(3.2)	**(3.3)**	**(3.4)**
Cancer	96.42	96.42	96.42
Glass	68.22	73.36	**73.83**
Iris	96.67	97.33	**97.33**
Pima	**77.34**	76.95	77.34
Sonar	**82.21**	79.81	78.85
Wine	96.07	96.63	**97.19**
Image	86.19	**86.76**	86.19
Vowel	51.72	55.25	**57.37**
Yeast	53.77	54.18	**55.73**

Chỉ có duy nhất bài toán Iris có tỉ lệ thấp hơn, tuy nhiên tỉ lệ thấp hơn rất nhỏ chỉ 0.26%.

Bảng 6.4 cho thấy thuật toán OPHA-SGERD có tỉ lệ phân lớp chính xác cao hơn thuật toán SGERD trên 6 bài toán, đặc biệt là bài toán Glass có tỉ lệ cao hơn nhiều là 9.98%, Pima là 3.87%, Sonar là 4.6%. Các bài toán có tỉ lệ thấp hơn không đáng kể như bài toán Cancer là 0.6%, Vowel là 3.28% và Yeast là 2.35%. Với tiêu chuẩn đánh giá (3.3) cho thấy thuật toán OPHA-SGERD tạo ra hệ luật có tỉ lệ phân lớp chính xác cao đồng thời dễ hiểu với người sử dụng nhờ vào các điều kiện tiền đề là các từ ngôn ngữ được sinh ra bởi ĐSGT và chiều dài luật ngắn. Bảng 6.4 cho thấy chiều dài hệ luật sinh ra bởi thuật toán OPHA-SGERD thấp hơn thuật toán SGERD trên 8 bài toán và chỉ cao hơn trên 1 bài toán. Bảng 6.5 cho thấy tiêu chuẩn đánh giá (3.4) cho ta hệ luật có tỉ lệ phân lớp chính xác cao hơn tiêu chuẩn đánh giá (3.3), tuy nhiên sự khác biệt không được rõ ràng, do chỉ có 5 trong 9 bài toán có tỉ lệ phân lớp chính xác cao hơn, cao nhất trên bài toán Vowel là 2.12%. Từ bảng 6.6 cho thấy tiêu chuẩn đánh giá (3.4) tạo ra hệ luật có tỉ lệ phân lớp chính xác cao nhất trong 3 tiêu chuẩn đánh giá (3.2), (3.3) và (3.4).

Từ những so sánh đánh giá ở trên cho thấy thuật toán OPHA-SGERD tạo ra các hệ luật có tỉ lệ phân lớp chính xác cao hơn thuật toán SGERD, dễ hiểu với người sử dụng do có chiều dài luật ngắn và các điều kiện tiền đề là các từ ngôn ngữ có ngữ nghĩa.

7. KẾT LUẬN

Bài báo đã đề xuất một phương pháp xây dựng hệ luật mờ cho bài toán phân lớp dựa trên việc cải tiến phương pháp SGERD bằng việc ứng dụng ĐSGT và thuật toán tối ưu hệ luật, đồng thời đề xuất tiêu chuẩn đánh giá các luật dựa trên việc cải tiến tiêu chuẩn đánh giá đề xuất trong [17].

Phương pháp sinh hệ luật ứng cử có ý tưởng khác so với các phương pháp đề xuất [6-8, 10-15, 20]. Từ các luật có chiều dài 1 với không gian mờ lớn có độ đo thích nghi nhỏ. Sau mỗi bước thuật toán sinh ra các luật có chiều dài tăng thêm một đơn vị. Các luật thế hệ con có không gian mờ giảm và độ đo thích nghi tăng lên. Do đó tăng tính chính xác khi phân lớp và đảm bảo hệ luật sinh ra đơn giản bởi các luật có độ dài ngắn được xem xét trước. Với việc áp dụng ĐSGT hệ luật được tạo ra đảm bảo tính dễ hiểu với con người. Đặc biệt thuật giải di truyền với số thế hệ hữu hạn, nhiều nhất bằng số thuộc tính của luật, tốn ít thời gian cho việc xây dựng hệ luật ứng cử và vì vậy có thể giải quyết được các bài toán với số mẫu dữ liệu và số chiều lớn.

Kết quả thử nghiệm cho thấy, hệ luật xây dựng dựa trên ĐSGT cho kết quả phân lớp chính xác cao hơn 8 trong 9 bài toán với tiêu chuẩn đánh giá (3.2) và 6 trong 9 bài toán với tiêu chuẩn đánh giá (3.3). Nhìn chung hệ luật sinh ra dễ hiểu với người sử dụng do chiều dài của luật giảm. Tiêu chuẩn đánh giá (3.4) đề xuất trong bài báo cho kết quả phân lớp chính xác cáo hơn các tiêu chuẩn đánh giá còn lại trong hầu hết các bài toán.

TÀI LIỆU THAM KHẢO

[1] Nguyen Cat Ho, Fuzziness in structure of linguistic truth values: A foundation for development of fuzzy reasoning, *Proc. of ISMVL' 87*, Boston, USA, IEEE Computer Society Press, New York, 1987 (326–335).

[2] Nguyen Cat Ho and W. Wechler, Extended hedge algebras and their application to fuzzy logic, *Fuzzy Sets and Systems* **52** (1992) 259–281.

[3] Nguyen Cat Ho, H. Van Nam, Ordered structure-based semantics of linguistic terms of linguistic variables and approximate reasoning, *AIP conference proceedings on Computing Anticipatory Systems, CASYS'99 Third International Conference* **157** (1) (2000) 98–116.

[4] Nguyen Cat Ho, T. Đinh Khang, L.Xuan Viet, Fuzziness measure, quantified semantic mapping and interpolative method of approximate reasoning in medical expert systems, *Tạp chí Tin học và Điều khiển học* **18** (3) (2002) 237–252.

[5] Nguyen Cat Ho, Nguyen Van Long, Đại số gia tử đầy đủ tuyến tính, *Tạp chí Tin học và Điều khiển học* **19** (3) 274–280.

[6] Dương Thăng Long, Nguyễn Cát Hồ, Trần Thái Sơn, Một phương pháp xây dựng hệ luật mờ có trọng số để phân lớp dựa trên đại số gia tử, *Tạp chí Tin học và Điều khiển học* **26** (1) (2010) 55–72.

[7] Nguyễn Cá 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.

[8] 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*, năm 2010.

[9] Hoàng Kiếm, Lê Hoàng Thái, *Giải thuật di truyền - Cách giải tự nhiên các bài toán trên máy tính*, Nhà Xuất bản giáo dục, năm 2000.

[10] H. Ishibuchi, T. Nakashima, T Murata, Three-objective genetics-based machine learning for linguistic rule extraction, *Information Scienc* **136** (2001) 109–133.

[11] 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.

[12] H. Ishibuchi, Multiobjective genetic fuzzy systems: Review and future research directions, *Proc. of 2007 IEEE International Conference on Fuzzy Systems*, London, UK, July 23-26, 2007 (913–918)

[13] H. Ishibuchi, T. Murata, and I.B.Turksen, Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classication problems, *Fuzzy Sets and Syst.* **89** (2) (1997) 135–150.

[14] H. Ishibuchi and T. Yamamoto, Rule weight specication in fuzzy rule-based classication systems, *IEEE Trans. Fuzzy Syst.* **13** (4) (Aug. 2005) 428–435.

[15] 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** (2004) 59–88.

[16] E. G. Mansoori, M. J. Zolghadri, and S. D. Katebi, A weighting function for improving fuzzy classication systems performance, *Fuzzy Sets Syst.* **158** (5) (2007) 583–591.

[17] Eghbal G. Mansoori, Mansoor J. Zolghadri, and Seraj D. Katebi, SGERD: A steady-sate generatic algorithm for extracting fuzzy classification rules from data, *IEEE Transactions on Fuzzy Systems* **16** (4) (August 2008).

[18] The Machine Learning Repository of University of California - Irvine, http://archive.ics.uci.edu/ml/datasets.html.

[19] J. H. Holland, *Adaptation in Natural and Articial Systems.AnnArbor*, MI: Univ. Michigan Press, 1975.

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

Ngày nhận bài 16 - 7 - 2012
Nhận lại sau sửa ngày 5 - 12 - 2012