ĐẠI SỐ GIA TỬ HẠN CHẾ AX^2 VÀ ỨNG DỤNG CHO BÀI TOÁN PHÂN LỚP MỜ

NGUYỄN CÁT HỒ, TRẦN THÁI SƠN, DƯƠNG THĂNG LONG

1. GIỚI THIỆU

Đại số gia tử (*HA*) đã được nhiều tác giả nghiên cứu và ứng dụng [2, 3, 11, 12, 13], điều này cho thấy tiềm năng ứng dụng rất lớn của *HA*. Cấu trúc một đại số gia tử gồm $\boldsymbol{AX} = (\boldsymbol{X}, \boldsymbol{G}, \boldsymbol{C}, \boldsymbol{H}, \boldsymbol{\Sigma}, \boldsymbol{\Phi}, \leq)$. Trong đó $\boldsymbol{X}$ là tập các hạng từ (terms) của đại số, $\boldsymbol{G}=\{c^-,c^+\}$ là tập phần tử sinh, $\boldsymbol{C}=\{\boldsymbol{0},\boldsymbol{1},\boldsymbol{W}\}$ các giá trị hằng, $\boldsymbol{H}$ là tập các gia tử (hedges), $\boldsymbol{\Phi}$ gia tử tới hạn min, $\boldsymbol{\Sigma}$ gia tử tới hạn max, $\leq$ quan hệ thứ tự ngữ nghĩa giữa các hạng từ.

Thực tế, các tác giả áp dụng ***HA*** trong các mô hình giải các bài toán sử dụng ít các gia tử [3, 12, 13] với những nghiên cứu về ***HA*** trong ứng dụng các bài toán phân lớp, chúng tôi tiếp cận đại số gia tử hạn chế chỉ với 2 gia tử (kí hiệu $\boldsymbol{AX^2}$). $\boldsymbol{AX^2}$ gồm một gia tử dương và một gia tử âm, không mất tính tổng quát, đặt $H^-=\{L\}$ và $H^+=\{V\}$. Các khái niệm và kết quả liên quan đến $\boldsymbol{AX^2}$ được trình bày ở phần sau, khẳng định tính hữu dụng của $\boldsymbol{AX^2}$ trong ứng dụng các mô hình phân lớp mờ. Trên cơ sở đó, chúng tôi đề xuất một mô hình tìm kiếm tối ưu tập luật mờ phân lớp sử dụng giải thuật di truyền (GA) và thử nghiệm trên bài toán GLASS [28].

Phân lớp là bài toán rất đặc trưng của khai phá dữ liệu, được nhiều tác giả nghiên cứu và phát triển các mô hình ứng dụng [3, 4, 9, 10, 14, 15 – 19, 20, 22 - 24, 27]. Bài toán cho một tập dữ liệu mẫu $\boldsymbol{P}=\{p_i=(d_{i1},\ldots d_{in};\ C_i) \mid i=1,\ldots,M\}$, p_i là mẫu dữ liệu thứ i, $C_i \in \{C_1,\ldots,C_m\}$ là nhãn phân lớp tương ứng, M là số mẫu dữ liệu, n là số thuộc tính, m là số lớp. Giải quyết bài toán tức phải xây dựng mô hình dựa trên tập mẫu này để phân lớp cho các dữ liệu với mục tiêu đạt hiệu quả phân lớp cao nhất, hơn nữa mô hình phải đơn giản, dễ hiểu đối với người dùng. Các phương pháp và kỹ thuật như lí thuyết học máy, trí tuệ nhân tạo, mạng nơron,... đã được sử dụng nhằm đạt mục đích trên [6, 21, 25].

Phương pháp tiếp cận theo hệ mờ được nhiều tác giả đề cập trong các bài báo [3, 10, 15 – 19, 25], trong đó mô hình dựa trên cơ sở hệ luật mờ sử dụng khá phổ biến. Luật mờ ở dạng này gồm tập các điều kiện và phần kết luận là nhãn phân lớp được biểu diễn như sau:

$$r_q : IF\ X_1\ is\ A_{q,1}\ AND\ldots AND\ X_n\ is\ A_{q,n}\ \ THEN\ \ C_q\ with\ CF_q, \tag{1}$$

trong đó, $X=(X_1,\ldots,X_n)$ là véctơ dữ liệu mẫu kích thước n (n là số thuộc tính), C_q là nhãn phân lớp đầu ra của luật và CF_q là trọng số (*weight*) của luật. Theo tiếp cận ***HA*** [Long-09], miền ngôn ngữ của mỗi thuộc tính được cấu trúc bởi đại số gia tử $\boldsymbol{AX^2}$, kí hiệu $\underline{\boldsymbol{X_j}}$, do đó $A_{q,j} \in \underline{\boldsymbol{X_j}}$ là một hạng từ (linguistic term) làm điều kiện của luật tương ứng với thuộc tính j.

Mỗi luật mờ phân lớp như trên có thể được xem như một dạng của luật kết hợp trong khái phá luật kết hợp (*association rules*), kí hiệu $r_q = (A_q \Rightarrow C_q)$. Một số định nghĩa về trọng số luật được đề xuất bởi Ishibuchi trong [15, 18], trong bài này chúng tôi sử dụng theo dạng $CF3$ sau:

$$CF3 = c(A_q \Rightarrow C_q) - c_{2nd}(A_q \Rightarrow C_q), \tag{2}$$

trong đó $c(A_q \Rightarrow C_q)$ là độ tin cậy của luật được tính theo luật kết hợp [18], $c_{2nd}(A_q \Rightarrow C_q)$ là độ tin cậy lớn nhất của các luật có vế trái là A_q còn kết luận khác C_q, chúng được tính bởi các công thức:

$$c(A_q \Rightarrow C_q) = \frac{\sum_{p_i \in class\,Cq} \mu_{Aq}(p_i)}{\sum_{i=1}^{M} \mu_{Aq}(p_i)} \tag{3}$$

$$c_{2nd} = \max\{\ c(A_q \Rightarrow C_h) \mid h=1,2...,m;\ class_h \neq C_q\ \} \tag{4}$$

với m là số lớp của tập dữ liệu mẫu, M là số mẫu dữ liệu, hàm tính độ phù hợp của một dữ liệu thực với hạng từ A_q, $\mu_{Aq}(p_i)$, được xem xét trong phần sau (phần 2, định nghĩa 2.3).

Hệ mờ dạng luật trên (1), kí hiệu S, có hai phương pháp lập luận các tác giả dùng nhiều là *single-winner-rule* và *weighted-vote* [18], theo cách *single-winner-rule* mỗi mẫu dữ liệu sẽ được tính mức độ thích hợp của nó với vế trái tất cả các luật, chọn luật nào có mức độ thích cao nhất và phân lớp đầu ra theo luật đó:

$$Classify(p) = \text{argmax}_{Cq}\ \{\ \mu_{Aq}(p).CF_q \mid \forall A_q \Rightarrow C_q \in\ S\ \}. \tag{5}$$

Trường hợp có nhiều luật cùng đạt giá trị lớn nhất nhưng kết luận phân lớp khác nhau, khi đó ta chọn ngẫu nhiên một trong số chúng.

2. ĐẠI SỐ GIA TỬ HẠN CHẾ AX^2 VÀ TIẾP CẬN ĐẾN BÀI TOÁN PHÂN LỚP

Kí hiệu $\boldsymbol{X_k} \subset \boldsymbol{X}$ là tập các hạng từ có độ dài đúng k, $\boldsymbol{I_k} = \{\mathfrak{I}_k(x) \mid x \in \boldsymbol{X_k}\}$ là tập các khoảng tính mờ mức k *(k-intervals)* xác định bởi các hạng từ trong $\boldsymbol{X_k}$ [2, 3] (có thể viết gọn $\mathfrak{I}_k$ hoặc $\mathfrak{I}(x)$ thay cho $\mathfrak{I}_k(x)$). Gọi $\boldsymbol{X_{(k)}}$ là tập các hạng từ có độ dài không quá k, như vậy $\boldsymbol{X_{(k)}} = \boldsymbol{X_1} \cup \ldots \cup \boldsymbol{X_k}$, tương tự kí hiệu $\boldsymbol{I_{(k)}} = \boldsymbol{I_1} \cup \ldots \cup \boldsymbol{I_k}$. Sau đây là một số khái niệm và kết quả liên quan đến $\boldsymbol{AX^2}$.

Hai khoảng tính mờ $\mathfrak{I}(x) \neq \mathfrak{I}(y)$ được gọi là kề nhau cùng mức nếu chúng được xác định bởi hai hạng từ x,y có cùng độ dài ($x,y \in \boldsymbol{X_k}$), hay $\mathfrak{I}(x),\mathfrak{I}(y) \in \boldsymbol{I_k}$ và chúng có một *điểm mút* chung, tức là $lmp(\mathfrak{I}(x)) = rmp(\mathfrak{I}(y)) \vee rmp(\mathfrak{I}(x)) = lmp(\mathfrak{I}(y))$, trong đó *lmp* và *rmp* là điểm mút trái và mút phải của khoảng tính mờ.

Kí hiệu $\mathfrak{I}(x) \mid\!< \mathfrak{I}(y)$ là khoảng tính mờ $\mathfrak{I}(x)$ kề bên trái $\mathfrak{I}(y)$, hay $rmp(\mathfrak{I}(x)) = lmp(\mathfrak{I}(y))$, tương tự $x \mid\!< y$ là hạng từ x kề bên trái hạng từ y: $\forall z \in \boldsymbol{X_{(k)}}$, $z \neq x$, $z \neq y$, $(z < x < y) \vee (z > x > y)$.

Định nghĩa 2.1. Cho $\boldsymbol{AX^2}$, $\forall x \in \boldsymbol{X}$ ($k = l(x)$ là độ dài hạng từ x), khoảng tính mờ tương tự bậc g ($g \geq 1$) của x là khoảng được tạo bởi hai khoảng tính mờ kề nhau cùng mức $k+g$ chứa $\upsilon(x)$ làm điểm trong, kí hiệu $\mathfrak{I}_g(x)$, được xác định như sau:

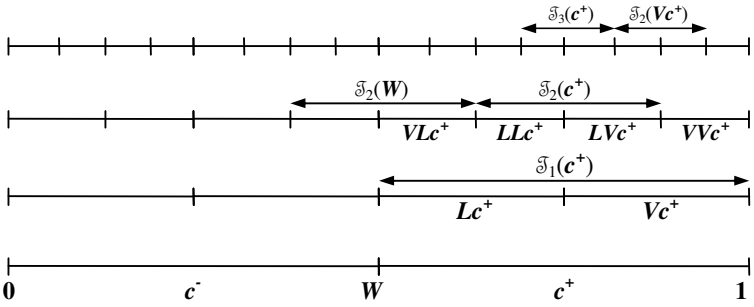
i) nếu $\upsilon(x)=0$ hoặc $\upsilon(x)=1$ thì $\mathfrak{I}_g(x) = \mathfrak{I}_{k+g}(y)$, sao cho $y \in \boldsymbol{X_{k+g}}$, $\upsilon(x) \in \mathfrak{I}_{k+g}(y)$.

ii) ngược lại,

$\mathfrak{I}_g(x) = \mathfrak{I}_{k+g}(y_i) \oplus \mathfrak{I}_{k+g}(y_j)$, sao cho $y_i, y_j \in \boldsymbol{X_{k+g}}$, $rmp(\mathfrak{I}_{k+g}(y_i)) = lmp(\mathfrak{I}_{k+g}(y_j)) = \upsilon(x)$.

trong đó $\oplus$ là phép nối hai khoảng tính mờ kề nhau.

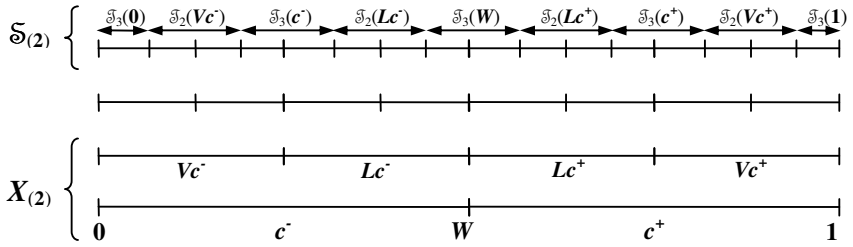
Định nghĩa này cho thấy một hạng từ có khoảng tính mờ tượng tự bậc càng cao sẽ càng bị thu hẹp về tâm của hạng từ đó: ta có $\forall x \in \boldsymbol{X}$, $g1 > g2$, $\mathfrak{I}_{g1}(x) \subset \mathfrak{I}_{g2}(x)$. Ví dụ hình 2.1, $\mathfrak{I}_3(\boldsymbol{c^+}) \subset \mathfrak{I}_2(\boldsymbol{c^+}) \subset \mathfrak{I}_1(\boldsymbol{c^+})$.



Hình 2.1. Minh họa khoảng tính mờ tương tự bậc g của các hạng từ trong $\boldsymbol{AX^2}$

Định nghĩa 2.2. Cho $\boldsymbol{AX^2}$, $k \geq 1$, hệ khoảng tính mờ tương tự của tập $\boldsymbol{X_{(k)}}$, kí hiệu $\mathfrak{S}_{(k)}$, là tập các khoảng tính mờ tương tự của tất cả các hạng từ trong $\boldsymbol{X_{(k)}}$, sao cho $\forall x \in \boldsymbol{X_{(k)}}$, $\mathfrak{I}_g(x) \in \mathfrak{S}_{(k)}$, $g+l(x) = k^*$ không đổi (tức là $\forall x \in \boldsymbol{X_{(k)}}$, $\mathfrak{I}_g(x)$ được tạo bởi các khoảng tính mờ cùng mức k^*) và $\mathfrak{S}_{(k)}$ là một phân hoạch của [0,1].

Ta gọi $\mathfrak{S}_{(k)}$ là hệ khoảng tính mờ tương tự mức k, tương ứng với tập $\boldsymbol{X_{(k)}}$.



Hình 2.2. Minh họa hệ khoảng tính mờ tương tự mức 2, tương ứng tập $\boldsymbol{X_{(2)}}$ trong $\boldsymbol{AX^2}$

Mệnh đề 2.1. Cho $\boldsymbol{AX^2}$, kích thước các tập $\boldsymbol{X_k}$, $\boldsymbol{X_{(k)}}$, $\boldsymbol{I_k}$, $\boldsymbol{I_{(k)}}$ được tính như sau:

i) Với $k = 1$, $|\boldsymbol{X_1}| = 5$.

ii) Với $k > 1$, $|\boldsymbol{X_k}| = 2^k$.

iii) $\forall k \geq 1$, $|\boldsymbol{X_{(k)}}| = 3+\sum_{i=1\cdots k} 2^k = 3+(2.2^k-2) = 1+2^{k+1}$.

iv) $|\boldsymbol{X_{(k+2)}}| = 2.(|\mathfrak{S}_{(k)}|-2)+2$.

Để ý rằng $|\boldsymbol{I_k}| = |\boldsymbol{X_k}|$, $|\boldsymbol{I_{(k)}}| = |\boldsymbol{X_{(k)}}|$.

Bổ đề 2.1. Cho $\boldsymbol{AX^2}$, $k \geq 1$, giả sử tập $\boldsymbol{X_{(k)}}$, $\boldsymbol{X_{k+1}}$ được sắp thứ tự, $\boldsymbol{X_{(k)}} = \{x_0 < \cdots < x_{i-1} < x_i < \cdots < x_{2^{k+1}}\}$, $\boldsymbol{X_{k+1}} = \{y_1 < y_2 < \cdots < y_{2^{k+1}}\}$. Với $\forall y_i, y_{i+1} \in \boldsymbol{X_{k+1}}$, $\exists x_i \in \boldsymbol{X_{(k)}}$: $y_i < x_i < y_{i+1}$, i=1,2,$\cdots$,2^{k+1}. Khi đó, tập $\boldsymbol{X_{(k+1)}} = \boldsymbol{X_{(k)}} \cup \boldsymbol{X_{k+1}} = \{ x_0 < y_1 < x_1 < \cdots < x_{i-1} < y_i < x_i < \cdots < y_{2^{k+1}} < x_{2^{k+1}} \}$.

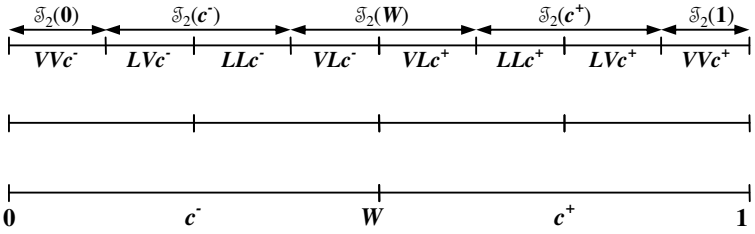
Chứng minh. Theo Mệnh đề 2.1, $|\boldsymbol{X_{(k)}}| = 1+2^{k+1}$, $|\boldsymbol{X_{k+1}}| = 2^{k+1}$, điều này thỏa mãn mỗi cặp theo thứ tự trong $\boldsymbol{X_{(k)}}$ tương ứng với một phần tử trong $\boldsymbol{X_{k+1}}$. Bây giờ ta chứng minh $\forall y_i \in \boldsymbol{X_{k+1}}$, $\exists x_{i-1}$, $x_i \in \boldsymbol{X_{(k)}}$: $x_{i-1} < y_i < x_i$, i=1,2,$\cdots$,2^{k+1}. Bằng phương pháp quy nạp, với trường hợp k=1, $\boldsymbol{X_{(1)}} = \{\boldsymbol{0}, \boldsymbol{c^-}, \boldsymbol{W}, \boldsymbol{c^+}, \boldsymbol{1}\}$ và $\boldsymbol{X_2} = \{\boldsymbol{Vc^-}, \boldsymbol{Lc^-}, \boldsymbol{Lc^+}, \boldsymbol{Vc^+}\}$, rõ ràng $\boldsymbol{X_{(2)}} = \{\boldsymbol{0} < \boldsymbol{Vc^-} < \boldsymbol{c^-} < \boldsymbol{Lc^-} < \boldsymbol{W} < \boldsymbol{Lc^+} < \boldsymbol{c^+} < \boldsymbol{Vc^+} < \boldsymbol{1}\}$, vậy k=1 được khẳng định.

Giả sử trường hợp $k > 1$ đã khẳng định, ta chứng minh cho trường hợp k+1. Theo giả thiết, $\boldsymbol{X_{(k)}} = \{x_0 < \cdots < x_{i-1} < x_i < \cdots < x_{2^{k+1}}\}$, $\boldsymbol{X_{k+1}} = \{y_1 < y_2 < \cdots < y_{2^{k+1}}\}$ và $\boldsymbol{X_{(k+1)}} = \boldsymbol{X_{(k)}} \cup \boldsymbol{X_{k+1}} = \{ x_0 < y_1 < x_1 < \cdots < x_{i-1} < y_i < x_i < \cdots < y_{2^{k+1}} < x_{2^{k+1}} \}$ (*). Trong $\boldsymbol{AX^2}$, tính $\boldsymbol{X_{k+2}} = \{Ly_i, Vy_i : \forall y_i \in \boldsymbol{X_{k+1}}\}$ (bằng cách dùng hai gia tử $\{L,V\}$ tác động lên mỗi hạng từ trong $\boldsymbol{X_{k+1}}$). Dựa trên tính cấu trúc dàn của ***HA*** [11, 12], $\forall y_i \in \boldsymbol{X_{k+1}}$, ta có hoặc $\Phi y_i < Ly_i < y_i < Vy_i < \sum y_i$, hoặc $\Phi y_i < Vy_i < y_i < Ly_i < \sum y_i$, hơn nữa trong $\boldsymbol{X_{(k+1)}}$ ta cũng có $y_{i-1} < \Phi y_i < y_i < \sum y_i < y_{i+1}$, để ý rằng $y_0 = \mathbf{0}$, $y_{2^{k+1}} = \mathbf{1}$ (**). Từ (*) và (**) ta suy ra $\forall z, z' \in \boldsymbol{X_{k+2}}$, $\exists y_{i-1}, y_i, y_{i+1} \in \boldsymbol{X_{(k+1)}}$, $y_{i-1} < z < y_i < z' < y_{i+1}$, trong đó $z = Ly_i$, $z' = Vy_i$, hoặc $z' = Vy_i$, $z = Ly_i$ (i=1,2,$\cdots$,2^{k+1}). Vậy k+1 được khẳng định $\Rightarrow$ *đpcm*. ■

Định lí 2.1. Cho $\boldsymbol{AX^2}$, $k \geq 1$, với tập $\boldsymbol{X_{(k)}}$, tồn tại duy nhất một hệ các khoảng tính mờ tương tự mức k, $\mathfrak{S}_{(k)}$ được tạo bởi phân hoạch các khoảng tính mờ mức k+2, $\boldsymbol{I_{k+2}}$.

Chứng minh. Rõ ràng tập $\boldsymbol{I_{k+2}}$ là một phân hoạch của [0,1] [11, 12], vậy nếu $\mathfrak{S}_{(k)}$ được tạo từ $\boldsymbol{I_{k+2}}$ thỏa mãn là một phân hoạch của [0,1] (thỏa định nghĩa 2.2). Ta chỉ cần chứng minh $\mathfrak{S}_{(k)}$ được tạo duy nhất từ phân hoạch $\boldsymbol{I_{k+2}}$.

(*i*) Trước hết, chứng minh $\mathfrak{S}_{(k)}$ được tạo từ phân hoạch $\boldsymbol{I_{k+2}}$. Bằng phương pháp quy nạp, với trường hợp k=1, ta có $\boldsymbol{X_{(1)}} = \{\mathbf{0}, \boldsymbol{c^-}, \boldsymbol{W}, \boldsymbol{c^+}, \mathbf{1}\}$ và $\boldsymbol{I_3} = \{\mathfrak{I}(VVc^-), \mathfrak{I}(LVc^-), \mathfrak{I}(LLc^-), \mathfrak{I}(VLc^-), \mathfrak{I}(VLc^+), \mathfrak{I}(LLc^+), \mathfrak{I}(LVc^+), \mathfrak{I}(VVc^+)\}$. Theo định nghĩa 2.1, hệ khoảng tính mờ tương tự của $\boldsymbol{X_{(1)}}$ được xây dựng như sau, $\mathfrak{S}_{(1)} = \{\mathfrak{I}_2(\mathbf{0}) = \mathfrak{I}(VVc^-), \mathfrak{I}_2(\boldsymbol{c^-}) = \mathfrak{I}(LVc^-) \oplus \mathfrak{I}(LLc^-), \mathfrak{I}_2(\boldsymbol{W}) = \mathfrak{I}(VLc^-) \oplus \mathfrak{I}(VLc^+), \mathfrak{I}_2(\boldsymbol{c^+}) = \mathfrak{I}(LLc^+) \oplus \mathfrak{I}(LVc^+), \mathfrak{I}_2(\mathbf{1}) = \mathfrak{I}(VVc^+)\}$ (hình vẽ 2.2), vậy k=1 được khẳng định.



Hình 2.2. Hệ khoảng tính mờ tương tự $\mathfrak{S}_{(1)}$ của $\boldsymbol{X_{(1)}}$

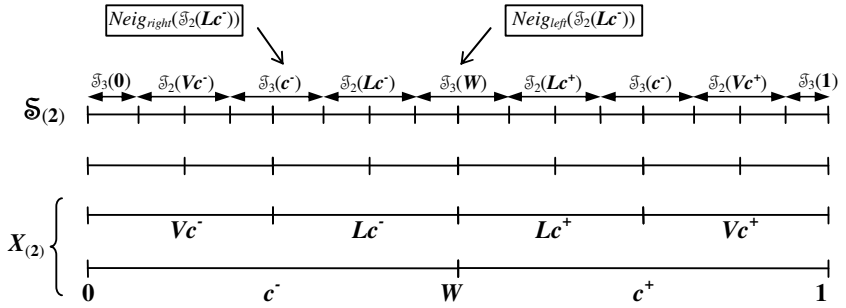
Giả sử trường hợp $k > 1$ đã khẳng định, tức là $\mathfrak{S}_{(k)}$ của $\boldsymbol{X_{(k)}}$ được tạo bởi $\boldsymbol{I_{k+2}}$. Ta chứng minh cho trường hợp k+1. Để ý rằng $\boldsymbol{I_k} = \{\mathfrak{I}(y_i) \mid \forall y_i \in \boldsymbol{X_k}\}$. Không mất tính tổng quát, theo tính chất của ***HA***, giả sử các tập $\boldsymbol{X_k}$, $\boldsymbol{X_{(k)}}$ và $\boldsymbol{X_{k+2}}$ được sắp thứ tự. Hệ khoảng tính mờ tương tự mức k+2 của $\boldsymbol{X_{(k)}}$ được xây dựng như sau, $\mathfrak{S}_{(k)} = \{ \mathfrak{I}(w_i) = \mathfrak{I}(y_{2i}) \oplus \mathfrak{I}(y_{2i+1}) \mid y_{2i}, y_{2i+1} \in \boldsymbol{X_{k+2}}, i=1,2,\cdots,2^{k+1}\text{-}1\} \cup \{\mathfrak{I}(y_1), \mathfrak{I}(y_{2^{k+2}})\}$, điều này thỏa mãn (theo mệnh đề 2.1-*iv*), để ý rằng $\mathfrak{I}(\mathbf{0}) = \mathfrak{I}(y_1)$, $\mathfrak{I}(\mathbf{1}) = \mathfrak{I}(y_{2^{k+2}}) \in \mathfrak{S}_{(k)}$. Trong $\boldsymbol{AX^2}$, sinh tập $\boldsymbol{X_{k+1}} = \{Lw_i, Vw_i : \forall w_i \in \boldsymbol{X_k}\}$, $\boldsymbol{X_{k+3}} = \{Ly_i, Vy_i : \forall y_i \in \boldsymbol{X_{k+2}}\}$ (bằng cách dùng hai gia tử $\{L,V\}$ tác động lên mỗi hạng từ trong $\boldsymbol{X_k}$, $\boldsymbol{X_{k+2}}$), tương ứng ta có $\boldsymbol{I_{k+3}}$. Biết rằng $\boldsymbol{X_{(k+1)}} = \boldsymbol{X_{(k)}} \cup \boldsymbol{X_{k+1}}$, $\boldsymbol{X_{(k+2)}} = \boldsymbol{X_{(k+1)}} \cup \boldsymbol{X_{k+2}}$, $\boldsymbol{X_{(k+3)}} = \boldsymbol{X_{(k+2)}} \cup \boldsymbol{X_{k+3}}$. Theo bổ đề 2.1, từ $\boldsymbol{X_{(k)}}$ và $\boldsymbol{X_{k+1}}$ ta có thứ tự tập $\boldsymbol{X_{(k+1)}} = \{\cdots < w_{i-1} < x_i < w_i < x_{i+1} < w_{i+1} < \cdots\}$, trong đó w_{i-1}, w_i, $w_{i+1} \in \boldsymbol{X_{(k)}}$, x_i, $x_{i+1} \in \boldsymbol{X_{k+1}}$, i=1,2,$\cdots$, 2^{k+1}-1. Tương tự, với x_{j-1}, x_j, $x_{j+1} \in \boldsymbol{X_{(k+1)}}$, y_j, $y_{j+1} \in \boldsymbol{X_{k+2}}$, j=1,2,$\cdots$,2^{k+2}-1, ta có thứ tự tập $\boldsymbol{X_{(k+2)}} = \{\cdots < x_{j-1} < y_j < x_j < y_{j+1} < x_{j+1} < \cdots\}$, và z_{2j-1}, z_{2j}, z_{2j+1}, $z_{2(j+1)} \in \boldsymbol{X_{k+3}}$, suy ra thứ tự tập $\boldsymbol{X_{(k+3)}} = \{\cdots < x_{j-1} < z_{2j-1} < y_j < z_{2j} < x_j < z_{2j+1} < y_{j+1} < z_{2(j+1)} < x_{j+1} < \cdots\}$. Xây dựng tập $\mathfrak{S}_{(k+1)} = \{$

$\mathfrak{I}(x_j)=\mathfrak{I}(z_{2j}) \oplus \mathfrak{I}(z_{2j+1}) \mid z_{2j}, z_{2j+1} \in \boldsymbol{X}_{k+3}, x_j \in \boldsymbol{X}_{(k+1)}, j=1,2,\cdots,2^{k+2}-1\} \cup \{\mathfrak{I}(z_1), \mathfrak{I}(z_{2^{k+3}})\}$, để ý rằng $\mathfrak{I}(\mathbf{0}) = \mathfrak{I}(z_1)$, $\mathfrak{I}(\mathbf{1}) = \mathfrak{I}(z_{2^{k+3}}) \in \mathfrak{S}_{(k+1)}$. Vậy trường hợp k+1 được khẳng định.

(*ii*) Chứng minh không tồn tại $m \geq 1$, $m \neq 2$, để $\mathfrak{S}_{(k)}$ được tạo từ $\boldsymbol{I}_{k+m}$. Theo mệnh đề 2.1, $|\boldsymbol{I}_{k+m}| = 2^{k+m}$, $|\mathfrak{S}_{(k)}| = |\boldsymbol{X}_{(k)}| = 1+2^{k+1}$, mặt khác ta phải có $2.(|\mathfrak{S}_{(k)}|-2)+2 = |\boldsymbol{I}_{k+m}|$, hay $2.(1+2^{k+1}-2)+2 = 2^{k+m} \Leftrightarrow 2^{k+2} = 2^{k+m} \Leftrightarrow m = 2$.

Từ (*i*) và (*ii*) $\Rightarrow$ *đpcm*. ■

Theo định nghĩa 2.2, với $k \geq 1$, $\forall x \in \boldsymbol{X}_{(k)}$, xác định $\mathfrak{I}_g(x) \in \mathfrak{S}_{(k)}$, ta có hai khoảng tính mờ tương tự trong $\mathfrak{S}_{(k)}$ kề bên trái là $Neig_{left}(\mathfrak{I}_g(x)) = \mathfrak{I}_{g1}(y)$, kề bên phải là $Neig_{right}(\mathfrak{I}_g(x)) = \mathfrak{I}_{g2}(z)$ với $y,z \in \boldsymbol{X}_{(k)}$, $y < x < z$ và không tồn tại hạng từ trong $\boldsymbol{X}_{(k)}$ nằm giữa chúng, tức là $lmp(\mathfrak{I}_g(x)) = rmp(\mathfrak{I}_{g1}(y)) \wedge rmp(\mathfrak{I}_g(x)) = lmp(\mathfrak{I}_{g2}(z))$. Trường hợp $x = \mathbf{0}$, ta xác định $Neig_{left}(\mathfrak{I}_g(x)) = \mathfrak{I}(\mathbf{0})$ (tức là y=$\mathbf{0}$), tương tự khi $x = \mathbf{1}$, $Neig_{right}(\mathfrak{I}_g(x)) = \mathfrak{I}(\mathbf{1})$ (tức là $z = \mathbf{1}$).



Hình 2.2. Hệ phân hoạch các khoảng tính mờ tương tự và láng giềng của chúng

Định nghĩa 2.3. Cho $\boldsymbol{AX}^2$, $k \geq 1$, $\forall x \in \boldsymbol{X}_{(k)}$, xác định một khoảng tính mờ tương tự $\mathfrak{I}_g(x) \in \mathfrak{S}_{(k)}$, định nghĩa độ tương hợp bậc $g = k+2-l(x)$ của giá trị định lượng v đối với hạng từ x là một ánh xạ $s_g : [0,1] \times \boldsymbol{X} \rightarrow [0,1]$, được xác định dựa trên khoảng cách từ v đến $\upsilon(x)$ và hai khoảng tính mờ tương tự láng giềng của $\mathfrak{I}_g(x)$ như sau:

$$s_g(v,x) = \max\left(\min\left(\frac{v-\upsilon(y)}{\upsilon(x)-\upsilon(y)}, \frac{\upsilon(z)-v}{\upsilon(z)-\upsilon(x)} \right), 0 \right), \tag{6}$$

trong đó y,z là hai hạng từ xác định hai khoảng tính mờ tương tự láng giềng trái và phải của $\mathfrak{I}_g(x)$.

Độ tương hợp này đóng vai trò như một hàm thuộc trong lí thuyết tập mờ, nó được sử dụng cho các tính toán lập luận trong các ứng dụng về sau.

Hệ quả 2.1. Cho $\boldsymbol{AX}^2$, $k \geq 1$, $\forall v \in [0,1]$, luôn tồn tại duy nhất một hạng từ $x \in \boldsymbol{X}_{(k)}$ xác định khoảng tính mờ tương tự bậc $g=k+l(x)$, $\mathfrak{I}_g(x) \in \mathfrak{S}_{(k)} \wedge v \in \mathfrak{I}_g(x)$.

Chứng minh. Dễ dàng suy ra từ tính phân hoạch của $\mathfrak{S}_{(k)}$.. ■

Hệ quả này cho thấy khả năng ứng dụng của hệ khoảng tính mờ tương tự $\mathfrak{S}_{(k)}$ của $\boldsymbol{AX}^2$ trong mọi quá trình thực.

Định lí 2.2. Cho $\boldsymbol{AX^2}$, bộ tham số mờ gia tử $0 < fm(\boldsymbol{c}^-), \mu(L) < 1$ (để ý rằng $fm(\boldsymbol{c}^+) = 1\text{-}fm(\boldsymbol{c}^-)$, $\mu(V) = 1\text{-}\mu(L)$), với $u,v \in [0,1]$, $u \neq v$, luôn tồn tại một hệ phân hoạch tương tự mức k, $\mathfrak{S}_{(k)}$ (tương ứng tập $\boldsymbol{X}_{(k)}$) sao cho $u \in \mathfrak{I}(x)$, $v \in \mathfrak{I}(y)$, $\mathfrak{I}(x), \mathfrak{I}(y) \in \mathfrak{S}_{(k)}$ (hay $x,y \in \boldsymbol{X}_{(k)}$) và $x \neq y$.

Chứng minh. Theo định lí về tính trù mật của $\upsilon(x)$, $\forall x \in \boldsymbol{X}$ trong [0,1] của ***HA*** [11, 12] và biết rằng $\boldsymbol{X} = \bigcup_{k=1}^{\infty} X_k$. Suy ra, $\forall \varepsilon$ đủ bé, $\exists k$, $\forall x,y \in \boldsymbol{X}_{(k)} \subset \boldsymbol{X}$, $x \neq y$, $|\upsilon(x) - \upsilon(y)| < \varepsilon$. Chọn $\varepsilon = |u - v|$, hơn nữa, $\upsilon(x) \in \mathfrak{I}(x)$, $\upsilon(y) \in \mathfrak{I}(y)$ (định nghĩa 2.1) và theo tính phân hoạch của $\mathfrak{S}_{(k)}$, do đó u,v không cùng nằm trong một khoảng tính mờ tương tự $\mathfrak{I}(x)$ hoặc $\mathfrak{I}(y)$. Vậy ta có hoặc $u \in \mathfrak{I}(x) \wedge v \notin \mathfrak{I}(x)$; hoặc $u \notin \mathfrak{I}(y) \wedge v \in \mathfrak{I}(y) \Rightarrow đpcm$. ■

Định lí này cho thấy khả năng xấp xỉ mọi điểm trong [0,1] của hệ khoảng tính mờ tương tự trong $\boldsymbol{AX^2}$.

Hệ quả 2.2. Cho $\boldsymbol{AX^2}$, bộ tham số mờ gia tử $0 < fm(\boldsymbol{c}^-), \mu(L) < 1$ (để ý rằng $fm(\boldsymbol{c}^+) = 1\text{-}fm(\boldsymbol{c}^-)$, $\mu(V) = 1\text{-}\mu(L)$), một tập con $E \subset [0,1]^1$, luôn tồn tại một hệ phân hoạch tương tự mức k, $\mathfrak{S}_{(k)}$ (tương ứng tập $\boldsymbol{X}_{(k)}$) sao cho $\forall u,v \in E$, $u \neq v$, $!\exists x \neq y$, $u \in \mathfrak{I}(x)$, $v \in \mathfrak{I}(y)$, $\mathfrak{I}(x), \mathfrak{I}(y) \in \mathfrak{S}_{(k)}$.

Chứng minh. Dễ dàng suy ra từ định lí 2.2 bằng cách đặt $k = max\{ k_i \mid k_i$ xác định bằng định lí 2.2 cho mọi cặp $u \neq v \in E\}$. ■

3. XÂY DỰNG HỆ LUẬT MỜ CHO BÀI TOÁN PHÂN LỚP

3.1. Thiết kế thuật toán sinh tập luật khởi đầu

Với tập dữ liệu mẫu $\boldsymbol{P}=\{p_i=(d_{i1},\ldots d_{iN};\ C_i) \mid i=1,\ldots,M\}$, mỗi thuộc tính được xem xét với miền ngữ nghĩa là các hạng từ của đại số gia tử $\boldsymbol{AX^2}$, kí hiệu $\boldsymbol{X}_j$. Trước hết, tính toán các hạng từ này với mức k_j cho trước và hệ khoảng tính mờ tương tự của chúng dựa trên các tham số mờ gia tử, kí hiệu $\boldsymbol{X}_{(kj)}=\{x_1, x_2,\ldots\}$ và $\mathfrak{S}_{(kj)}=\{\mathfrak{I}_{g1}(x_1), \mathfrak{I}_{g2}(x_2),\ldots\}$.

Theo định lí 2.1, mỗi mẫu dữ liệu $p_i=(d_{i1}, \ldots, d_{iN}; C_i)$ xác định duy nhất tập các hạng từ, kí hiệu $A_i=\{(x_{i1}, x_{i2},\ldots,x_{iN}) \mid x_{ij} \in \boldsymbol{X}_{(kj)}, d_{ij} \in \mathfrak{I}_{gij}(x_{ij}), j=1,2,\ldots,N\}$. Dựa trên cơ sở này, thuật toán sinh tập luật khởi đầu từ tập dữ liệu mẫu dựa trên đại số gia tử $\boldsymbol{AX^2}$ được đề xuất như sau:

Thuật toán 3.1. Khởi sinh tập luật từ tập mẫu (kí hiệu ***RFRG*** - Robust Fuzzy Rule Generation Algorithm)

Vào:

+ Tập mẫu $\boldsymbol{P}=\{p_i=(d_{i1},\ldots d_{iN};\ C_i) \mid i=1,\ldots,M\}$ có N thuộc tính, M là số mẫu,

+ Các tham số mờ gia tử trong $\boldsymbol{AX^2}$ và mức k_j cho tất cả các thuộc tính (có thể áp dụng giống nhau cho mọi thuộc tính hoặc các thuộc tính có các tham số khác nhau).

Ra:

+ Tập các luật mờ, kí hiệu S_0.

Các bước:

Step 1) Tính toán các hạng từ $\boldsymbol{X}_{(kj)}$ và hệ khoảng tính mờ tương tự $\mathfrak{S}_{(kj)}$ cho tất cả các thuộc tính dựa trên các tham số mờ gia tử và mức k_j,

Step 2) Lặp với mỗi dữ liệu $p_i \in \boldsymbol{P}$, thực hiện:

Step 2.a) Tính tập các điều kiện A_i của luật sinh bởi mẫu p_i (dựa trên định lí 2.1),

$$A_i = \{(x_{i1}, x_{i2},\ldots,x_{iN}) \mid x_{ij} \in \boldsymbol{X}_{(kj)}, d_{ij} \in \Im_{gij}(x_{ij}), j=1,2,\ldots,N\}$$

Step 2.b) Sinh luật mới gồm vế trái là A_i và kết luận là lớp C_i tương ứng với mẫu p_i,

$$r_i = (A_i \Rightarrow C_i)$$

Step 2.c) Thêm r_i vào tập luật: $S_0 = S_0 \cup \{r_i\}$.

Step 3) Kết quả S_0.

Đánh giá độ phức tạp của ***RFRG*** đối với tập mẫu $\boldsymbol{P}$: $\mathbf{O}_{\boldsymbol{RFRG}}(M.N)$.

Thuật toán ***RFRG*** này cho ta một hệ luật thô với độ dài đúng bằng số thuộc tính, số luật tối đa được tạo đúng bằng số mẫu. Tuy nhiên, theo sự phân bố của dữ liệu mẫu cũng như cách chọn mức phân hoạch hệ khoảng tính mờ tương tự, mức phân hoạch càng cao ($\mathfrak{S}_{(k)}$ với k càng lớn) thuật toán có cơ hội sinh số luật càng nhiều. Trong phần tiếp chúng tôi đề xuất một chiến lược tìm kiếm hệ luật tối ưu trong tập luật S_0 sinh bởi thuật toán này.

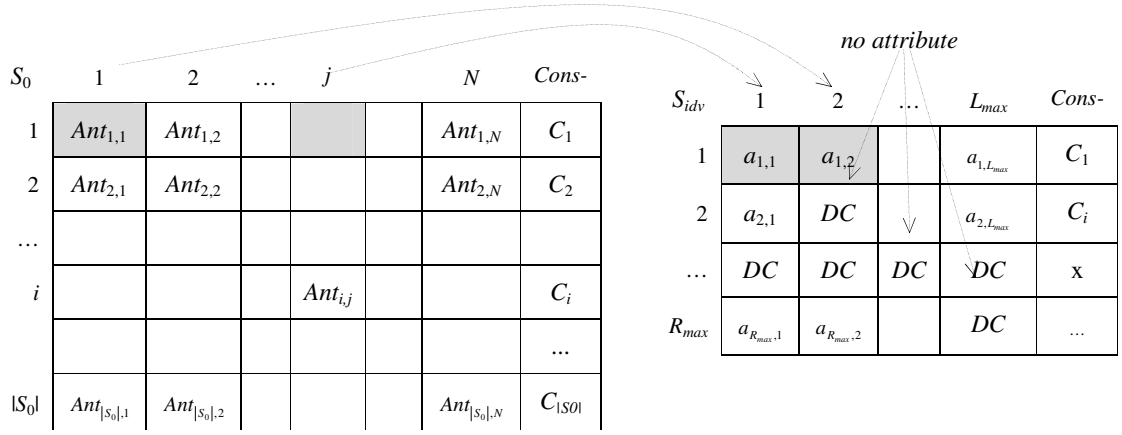
3.2. Thiết kế phương pháp tối ưu hệ luật

Bài toán đặt ra là cho tập mẫu $\boldsymbol{P} = \{p_i=(d_{i,1},\ldots,d_{i,n}; C_i) \mid i=1,\ldots,M\}$, $(d_{i,1},\ldots,d_{i,n}) \in \boldsymbol{D}$, $C_i \in \{C_1,\ldots,C_m\}$, n là số thuộc tính, M là số mẫu, m là số lớp. Với bộ tham số mờ gia tử của $\boldsymbol{AX^2}$ và mức phân hoạch hệ các khoảng tính mờ tương tự ($fm_j(c^-)$, $\mu_j(L)$, k_j), để ý rằng $fm_j(c^+)$=1- $fm_j(c^-)$, $\mu_j(V)$=1-$\mu_j(L)$, cho các thuộc tính X_j, j=1,…,N. Sử dụng thuật toán ***RFRG*** để sinh tập luật S_0, tập này có thể biểu diễn dưới dạng một bảng FAM (Fuzzy Association Memory), chúng ta thiết kế thuật toán tìm kiếm tối ưu hệ luật S dựa trên S_0 này (hình 3.1) với mục tiêu:

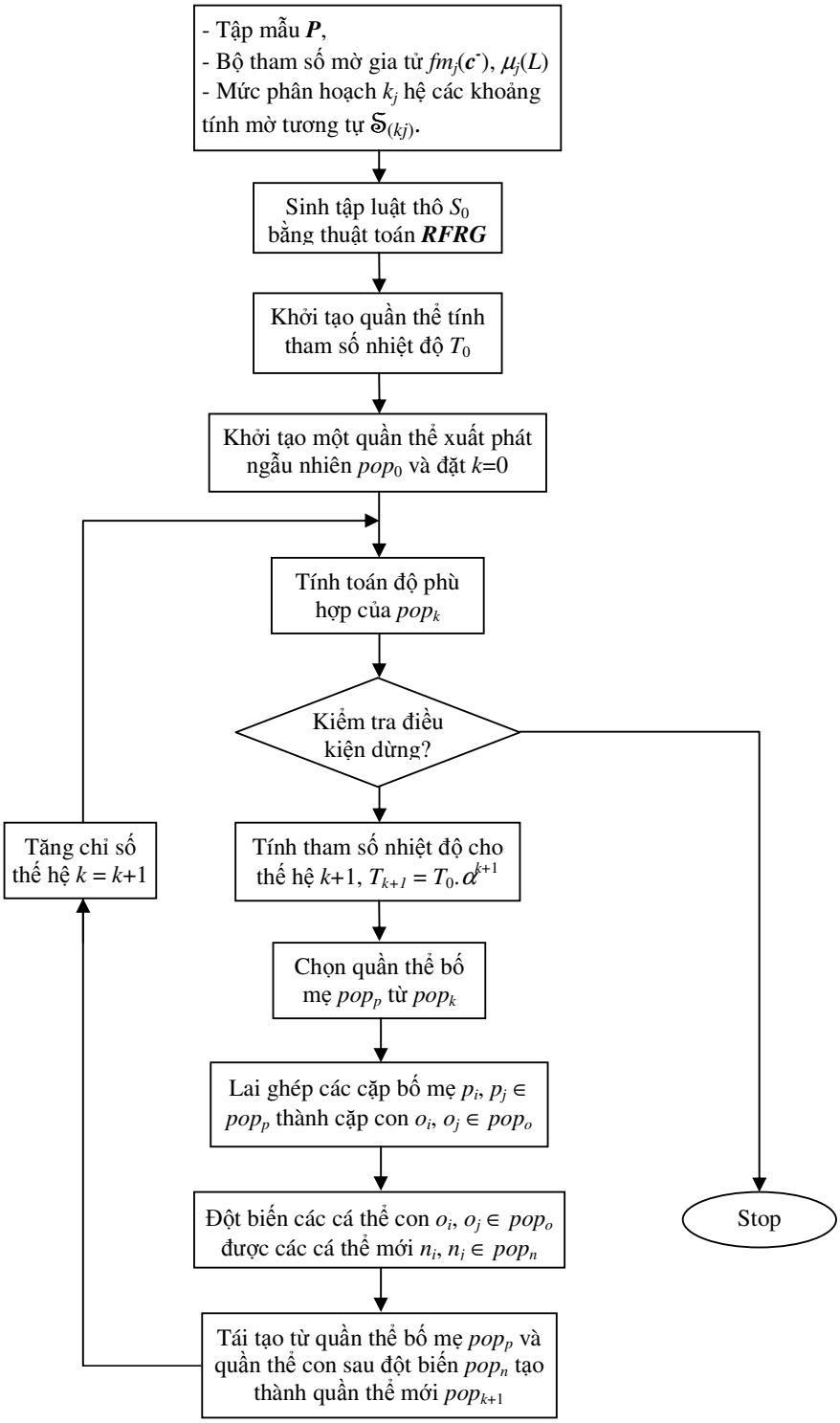
$$\textit{maximize } f_p(S), \textit{ minimize } f_n(S) \text{ và } \textit{minimize } f_a(S), \tag{7}$$

thỏa ràng buộc: $1 \le f_n(S) \le R_{max}$, $1 \le f_a(r) \le L_{max}$, $\forall r \in S$.

Trong đó S là một tập luật chọn từ S_0, $f_p(S)$ là hiệu quả phân lớp của hệ S đánh giá trên tập huấn luyện, $f_n(S)$ là số luật trong S, $f_a(S)$ là độ dài luật trung bình của tập S. R_{max} và L_{max} giới hạn số luật tối đa và độ dài luật tối đa được cho trước.



Hình 3.1. Minh họa chọn tập luật S từ tập S_0



Hình 3.2. Sơ đồ mô hình sinh và tìm kiếm hệ luật tối ưu bằng *GA+SA*

Một phương pháp thích nghi được nhiều tác giả sử dụng là giải thuật di truyền (GA) [8], trong bài này chúng tôi dùng *GA* kết hợp thuật toán mô phỏng tôi luyện thép (*SA*) [1, 3, 5, 7, 26] với hi vọng tăng tốc độ hội tụ và tránh sớm rơi vào hội tụ địa phương. Trước hết, mỗi cá thể mã hóa cho một lời giải tìm kiếm dưới dạng số thực gồm R_{max} nhiễm sắc thể như sau:

Mỗi nhiễm sắc thể (nst_i) chứa L_{max} gen ($a_{i,1}, a_{i,2}, \ldots, a_{i,L_{max}}$) biểu diễn chỉ số các thuộc tính được chọn cho luật thứ i (r_i), $0 \le \lfloor a_{i,j} \rfloor \le |S_0|.N$, i=1,2,...,R_{max}, j=1,2,...,L_{max}. Xác định điều kiện của thuộc tính được chọn cho S_{idv} theo $a_{i,j}$ là thuộc tính thứ $a_{i,j}\%N$ của luật thứ $\lfloor a_{i,j}/N \rfloor$ trong S_0, $a_{i,j} = 0$ có nghĩa thuộc tính không được chọn (*DC = Don't Care*) (phép % chia lấy số dư, kí hiệu $\lfloor \rfloor$ để lấy phần nguyên).

Trường hợp nếu $\exists i, \forall j : a_{i,j} = 0$, khi đó các thuộc tính làm điều kiện của luật thứ i không được chọn (*DC*), vậy luật này được loại bỏ trong hệ luật S_{idv} của cá thể tương ứng. Mặt khác, nếu $\exists i : a_{i,j1} = a_{i,j2}$, $j1 \ne j2$, tức là hai điều kiện thứ $j1$ và $j2$ của luật thứ i cùng chọn một thuộc tính trong S_0, thì loại bỏ một điều kiện thứ $j1$, khi đó xem như $a_{i,j1} = 0$.

Rõ ràng đây là một bài toán tối ưu đa mục tiêu, các tác giả trong [19] đã đưa ra một số phương pháp giải quyết như tối ưu *Pareto* bằng giải thuật di truyền. Ở đây, chúng tôi áp dụng giải thuật di truyền với một mục tiêu (hàm *fitness*) được kết nhập các mục tiêu của (7) theo trọng số:

$$fitness = w_p.f_p(S_{idv}) + w_n.(1\text{-}f_n(S_{idv})/R_{max}) + w_a.(1\text{-}f_a(S_{idv})/L_{max}),\ w_p + w_n + w_a = 1, \qquad (8)$$

trong đó S_{idv} là tập luật được chọn từ tập S_0 thông qua mã hóa của cá thể *idv*.

Các phép toán di truyền sử dụng ở đây như đã xét trong [3]. Mô hình sinh và tìm kiếm hệ luật mờ tối ưu phân lớp này được minh họa trong hình 3.2. Trong hình 3.2, điều kiện dừng được sử dụng là số thế hệ đạt tối đa cho trước hoặc đạt hiệu quả phân lớp tối đa.

4. THỬ NGHIỆM TRÊN BÀI TOÁN GLASS

Tập dữ liệu mẫu *glass* được công bố tại [28], đã có nhiều tác giả sử dụng thử nghiệm trong các kết quả nghiên cứu [18, 19, 21, 25]. Tập mẫu này có 214 mẫu dữ liệu, 9 thuộc tính và 6 lớp, phân bố các mẫu trên các lớp như sau (số mẫu/lớp): 70/0, 76/1, 17/2, 13/3, 9/4, 29/5.

Mô hình trên áp dụng thử nghiệm với bộ tham số mờ gia tử được chọn là $fm_j(\boldsymbol{c}^-) = 0{,}5$, $\mu_j(L) = 0{,}5$ (để ý $fm_j(\boldsymbol{c}^+) = 1\text{-}fm_j(\boldsymbol{c}^-)$, $\mu_j(V)=1\text{-}\mu_j(L)$), tuy nhiên chúng ta có thể tối ưu bộ tham số này [3]. Phân hoạch hệ các khoảng tính mờ tương tự cho miền ngôn ngữ của mỗi thuộc tính là $\mathfrak{S}_{(kj)}$ với $k_j = 2$, $j = 1,2,...,9$. Việc chọn bộ trọng số kết nhập các mục tiêu trong hàm *fitness* nhằm đảm bảo tính tối ưu của các mục tiêu là rất khó, chúng tôi chọn 4 bộ trọng số. Các tham số *GA* và của mô hình cho thử nghiệm như sau:

- Tham số tác động đến các toán tử gen mô phỏng *SA*: $\lambda = 0{,}7$, $\gamma_{max} = 9$.
- Kích thước quần thể cho việc tính tham số nhiệt của *SA* (T_0 và T_{end}): 333.
- Kích thước mỗi thế hệ trong tiến hóa: 500.
- Số thế hệ tiến hóa tối đa: 150.
- Bộ trọng số hàm *fitness*:

$W^1 = (w_p{=}0{,}5,\ w_a{=}10^{-4})$;
$W^2 = (w_p{=}0{,}9,\ w_a{=}10^{-4})$;
$W^3 = (w_p{=}0{,}99,\ w_a{=}10^{-4})$;
$W^4 = (w_p{=}0{,}999,\ w_a{=}10^{-4})$, để ý rằng $w_n = 1\text{-}(w_p{+}w_a)$.

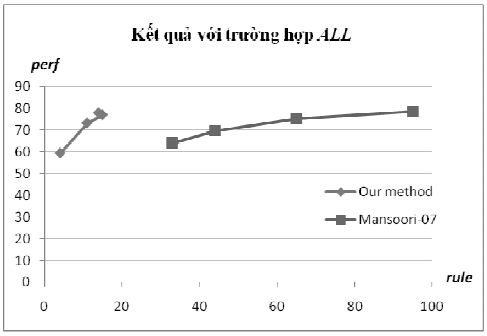
- Phương pháp lập luận (2): *single-winner-rule*
- Trọng số luật mờ xác định theo dạng 3 (5): *CF*3.
- Số luật giới hạn tối đa: R_{max} = 20.
- Số thuộc tính tối đa cho mỗi luật (độ dài luật tối đa): L_{max} = 3.

Bảng 4.1. Kết quả thử nghiệm trường hợp *ALL* và so sánh với [25]

Our method				Mansoori-07	
Weight of fitness	***PN***	***PL***	***PA/PT***	***PN***	***PA/PT***
W^1	4	2	59,35	-	-
W^2	11	2,55	73,36	-	-
W^3	15	2,6	77,1	-	-
W^4	14	2,5	78,04	-	-
-	-	-	-	33	64,02
-	-	-	-	44	69,63
-	-	-	-	65	75,23
-	-	-	-	95	78,5

(dấu "-" không có kết quả thử nghiệm, *PN* là số luật, *PL* là độ dài trung bình tập luật, *PA* là hiệu quả phân lớp trên tập training, *PT* là hiệu phân lớp trên tập testing)

Bằng các phương pháp thử nghiệm, thứ nhất tất cả tập mẫu dùng để sinh luật (training) và kiểm tra hiệu quả phân lớp (*ALL*), kết quả thể hiện trong bảng 4.1. So sánh với [25] cho thấy số luật nhỏ hơn nhiều nhưng hiệu quả phân lớp cao (hình 4.1). Với trường hợp W^4 cho kết quả 14 luật với hiệu quả phân lớp 78,04% trong khi của Mansoori đạt 78,5% với số luật lên đến 95, trường hợp W^3 cho 15 luật với hiệu quả 77,1% trong khi Mansoori đạt 75,23% với số luật 65.



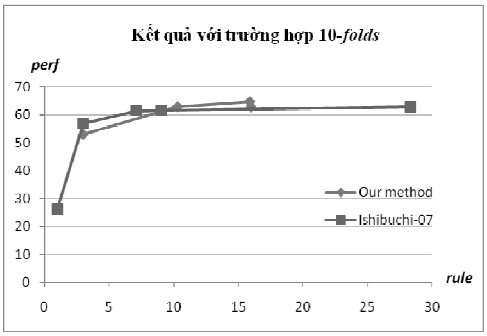
Hình 4.1. Đồ thị kết quả trường hợp *ALL* và so sánh với [25]

Một phương pháp thử nghiệm khác *k-cross-validation*, cụ thể *10-folds*, chia ngẫu nhiên tập mẫu thành 10 phần kích thước bằng nhau, sử dụng 1 phần để kiểm thử (testing) còn lại 9 phần để

sinh luật (training) và tìm kiếm hệ luật tối ưu. Chạy thử nghiệm 10 lần theo thứ thự mỗi phần kiểm thử được sử dụng trong 10 phần chia, tính kết quả gồm số luật, độ dài hệ luật, hiệu quả phân lớp trung bình trên các lần chạy thử nghiệm. Kết quả được thể hiện trong bảng 4.2 cho thấy với bộ trọng số W^3 đạt hiệu quả phân lớp cao nhất, trong khi W^1 cho số luật ít và hiệu quả phân lớp thấp. So sánh với Ishibuchi, mô hình đạt kết quả phân lớp tốt trên tập kiểm tra (hình 4.2), trường hợp W^2 có số luật trung bình 10,3 đạt 62,91% trên tập test, trong khi [19] chỉ đạt 61,64% tại 9,06 luật trung bình. Trường hợp W^3 mô hình đạt 64,67% với số luật trung bình 15,9, cao hơn [19] đạt 62,97% với số luật trung bình lên đến 28,32.

Bảng 4.2. Kết quả thử nghiệm 10-*folds* và so sánh với [19]

Our method					Ishibuchi-07		
Weight of fitness	***PN***	***PL***	***PA***	***PT***	***PN***	***PA***	***PT***
-	-	-	-	-	1	35,02	26,34
W^1	3	2,03	58,96	52,99	-	-	-
-	-	-	-	-	3,02	64,81	57,07
-	-	-	-	-	7,09	76,23	61,48
-	-	-	-	-	9,06	77,64	61,64
W^2	10,3	2,56	74,30	62,91	-	-	-
W^3	15,9	2,58	76,48	64,67	-	-	-
W^4	16	2,58	77,2	62,76	-	-	-
-	-	-	-	-	28,32	82,09	62,97
-	-	-	-	-	28,32	82,09	62,97



Hình 4.2. Đồ thị kết quả phân lớp trên tập testing (10-*folds*) và so sánh với [19]

5. KẾT LUẬN

Trong bài này chúng tôi đã giới thiệu những kết quả cơ bản về đại số gia tử $\boldsymbol{AX^2}$, một tiếp cận đại số gia tử cho bài toán phân lớp mờ trong khai phá dữ liệu và khả năng ứng dụng của nó.

Trên cơ sở đó, bài báo đề xuất một mô hình giải bài toán phân lớp dựa trên cơ sở hệ luật mờ và phương pháp lập trên hệ luật đó.

Tuy nhiên, đối với các mô hình phân lớp dựa trên hệ các luật mờ, các tác giả thường sử dụng các phương pháp tối ưu gồm: tối ưu về tham số mờ [3, 9, 22], hoặc/và tối ưu về hệ luật mờ [16, 17, 19] nhằm tăng hiệu quả phân lớp và giảm tính phức tạp của hệ luật mờ. Bài báo tiếp cận tối ưu hệ luật mờ dựa trên giải thuật di truyền (*GA*) được tích hợp thêm tham số nhiệt (phỏng theo giải thuật *SA*) làm tăng hiệu quả của *GA* [1, 5, 7]. Để tối ưu tham số mờ gia tử, chúng ta có thể tiếp cận dựa trên *GA* này [1, 3].

Áp dụng thử nghiệm mô hình trên tập dữ liệu *glass*, kết quả của hai trường hợp thử nghiệm là *ALL* và 10-*folds* cho thấy mô hình đạt hiệu quả tốt khi so sánh với một số mô hình khác [19, 25]. Đặc biệt mô hình này, trường hợp thử nghiệm 10-*folds*, cho hiệu quả phân lớp trên tập kiểm tra tốt hơn của Ishibuchi [19] (hình 4.1), với trường hợp thử nghiệm *ALL*, cho kết quả tốt hơn rất nhiều so với [25]. Kết quả này cho thấy mô hình được đề xuất dựa trên $\boldsymbol{AX^2}$ đạt hiệu quả cao trong bài toán phân lớp, nếu mô hình được tối ưu tham số mờ gia tử [3] sẽ cho kết quả cao hơn nữa.

TÀI LIỆU THAM KHẢO

1. Trần Ngọc Hà - Các hệ thống thông ming lai và ứng dụng trong xử lí dữ liệu, Luận án tiến sĩ, Đại học Bách khoa Hà Nội, 2002.
2. Nguyễn Cát Hồ - CSDL mờ với ngữ nghĩa đại số gia tử, Lectures on the Fuzzy Systems & Applications Autumn School, 9/2008, 2009, .
3. 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.
4. Johannes A. Roubos, Magne Setnes, Janos Abonyi - Learning fuzzy classification rules from labeled data, Information Sciences **150** (2003) 77–93.
5. Dan Adler - Genetic Algorithms and Simulated Annealing: A Marriage Proposal, Proc. of the International Conf. On Neural Networks **2** (1993) 1104-1109.
6. 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.
7. Sanjay Bisht - Hybrid Genetic-simulated Annealing Algorithm for Optimal Weapon Allocation in Multilayer Defence Scenario, Defence Science Journal **54** (3) (2004) 395-405.
8. Ulrich Bodenhofer - Genetic Algorithms: Theory and Applications, Lecture Notes, Third Edition-Winter 2003/2004, 2004.
9. Chia-Chong Chen - Design of PSO-based Fuzzy Classification Systems, Tamkang Journal of Science and Engineering **9** (1) (2006) 63-70.
10. Alberto Fernándeza, Salvador Garcíaa, María José del Jesusb, Francisco Herreraa - A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets, Fuzzy Sets and Systems **159** (2008) 2378-2398.
11. 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.

12. 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.
13. 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.
14. Cheng-Jian Lin, Chi-Yung Lee, and Shang-Jin Hong - An Efficient Fuzzy Classifier Based on Hierarchical Fuzzy Entropy, International Journal of Information Technology **12** (6) (2006).
15. H.Ishibuchi, T.Nakashima - Effect of Rule Weights in Fuzzy Rule-Based Classification Systems, IEEE Trans. on Fuzzy Systems **9** (4) (2001) 506-515.
16. H. Ishibuchi, T. Nakashima, T. Murata - Three-Objective Genetics-Based Machine Learning for Linguistic Rule Extraction, Information Science **136** (1-4) (2001) 109-133.
17. 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.
18. Hisao Ishibuchi and Takashi Yamamoto - Rule weight specification in fuzzy rule-based classification systems, IEEE Trans. on Fuzzy Systems **13** (4) (2005) 428-435.
19. 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.
20. Chen Ji-lin, Hou Yuan-long, Xing Zong-yi, Jia Li-min, and Tong Zhong-zhi - A Multi-objective Genetic-based Method for Design Fuzzy Classification Systems, International Journal of Computer Science and Network Security **6** (8A) (2006).
21. S.M. Fakhrahmad and M. Zolghadri Jahromi - A New Rule-weight Learning Method based on Gradient Descent, Proceedings of World Congress on Engineering 2009, Vol.I, WCE-2009.
22. Enwang Zhou and Alireza Khotanzad - Fuzzy Classifier Design Using Genetic Algorithms, Pattern Recognition **40** (12) (2007) 3401-3414.
23. Jiri Kubalika, Leon Rothkrantz, Jiri Lazanskya - Genetic Programming Fuzzy Rule Extractor Using Class Preserving Representation, The 13th Belgian-Dutch Conference on Artificial Intelligence, University of Amsterdam, 2001, pp.167-174.
24. Plamen Angelov, Xiaowei Zhou, Dimitar Filev, Edwin Lughofer - Architectures for Evolving Fuzzy Rule-based Classifiers, Proc. Systems, Man and Cybernetics conference (SMC) 2007, Montreal, Canada, 2007, pp. 2050-2055.
25. Eghbal G. Mansoori, Mansoor J. Zolghadri, Seraj D. Katebi - A weighting function for improving fuzzy classification systems performance, Fuzzy Sets and Systems **158** (2007) 583-591.
26. Indrajit Saha and Anirban Mukhopadhyay - Genetic Algorithm and Simulated Annealing based Approaches to Categorical Data Clustering, Proceedings of the International MultiConference of Engineers and Computer Scientists, Hong Kong, Vol. I, 2008.
27. Yung-Chou Chena, Li-HuiWang, and Shyi-Ming Chen - Generating Weighted Fuzzy Rules from Training Data for Dealing with the Iris Data Classification Problem, International Journal of Applied Science and Engineering **4** (1) (2006) 41-52.

28. UC Irvine Machine Learning Repository, http://archive.ics.uci.edu/ml.

SUMMARY

HEDGE ALGEBRAS WITH LIMITED NUMBER OF HEDGES AND APPLIED TO FUZZY CLASSIFICATION PROBLEMS

In this paper we introduce the Hedge Algebras with a limited number of hedges, called AX^2. In the AX^2, we consider the g-grade similarity fuzziness interval of a linguistic term x, denote $\mathfrak{I}_g(x)$, which are constructed from two $(g+k)$-fuzziness intervals satisfy that $\upsilon(x)$ is inside the interval (Definition 2.1, $k = l(x)$ is the length of x). There is a system of k-similarity fuzziness intervals, denote $\mathfrak{S}_{(k)}$, corresponding to a set of linguistic terms that their length is less than or equal to k, denote $\boldsymbol{X}_{(k)}$ (Definition 2.2). Especially, we prove that the system is always exist and a partition of [0,1], it is constucted by a set of $(k+2)$-fuzziness intervals (Theorem 2.1), so the AX^2 with its partition of k-similarity fuzziness intervals can be used in any real domain (Theorem 2.2).

Fuzzy rule-based systems are widely used for classification problems. There are two main goals in the design of fuzzy rule-based systems: one is the accuracy maximization and the other is the complexity minimization. Various approaches have been proposed to deal with the problem [19, 22, 25, 23]. So in the section 3, we propose an extracting fuzzy rules algorithm (*RFRG*) for classification problems base on the partitions $\mathfrak{S}_{(k)}$ of domain of attributes. The generated rules, denote S_0, of this algorithm content all attributes, i.e. their antecedents have full attributes of the problem, we call them a robust fuzzy rules-set. These rules can improve accuracy upto 100% by choosing particularly k-similarity fuzziness intervals of attributes (Corollary 2.2). However, this may increase the complexity of the fuzzy rules-set. To overcome this problem, we design an algorithm to optimize the fuzzy rules-set by using genetic algorithms and annealing simulation [1, 5, 7, 8, 26]. The solutions of this optimal problems are encoded in real encoding, which represents rule's index and attribute's index in S_0 to be selected, then the *fitness* function is given as a weighting of three objectives: maximize *the performance of rules-set*, minimize *the number of rules* and minimize *the average rule-length*.

In the section 4, we apply our method to the *glass* problem that posted in UCI machine learning repository. The results, in all patterns for training case, are better than [25] in comparision, the best accuracy of our method is 78.04% with 14 fuzzy rules while [Mansoori-07] is 78.5% with 95 fuzzy rules. In the 10-*folds* experiment, the best accuracy on testing patterns of our method is 64.67% with 15.9 average fuzzy rules, comparing with [19] is 62.97% with 28.32 average fuzzy rules. The comparision shows that the accuracy of our method is better than [19] and [25].

Địa chỉ: *Nhận bài ngày 20 tháng 11 năm 2009*

Nguyễn Cát Hồ, Trần Thái Sơn,

Viện Công nghệ Thông tin, Viện Khoa học và Công nghệ Việt Nam.

Dương Thăng Long,

Khoa Công nghệ Tin học, Viện Đại học Mở Hà Nội.