Tạp chí Tin học và Điều khiển học, T.28, S.2 (2012), 141-152

THUẬT TOÁN MỚI XÁC ĐỊNH ĐỘ TRỄ GIẢI MÃ CỦA NGÔN NGỮ CHÍNH QUY

ĐẶNG QUYẾT THẮNG[1], NGUYỄN ĐÌNH HÂN[2], PHAN TRUNG HUY[3]

[1] *Trường Đại học Sư phạm Kỹ thuật Nam Định*

[2] *Trường Đại học Sư phạm Kỹ thuật Hưng Yên*

[3] *Trường Đại học Bách khoa Hà Nội*

Tóm tắt. Bài báo đề xuất một giải thuật mới xác định độ trễ giải mã của ngôn ngữ chính quy được đoán nhận bởi ôtômat hữu hạn $\mathcal{A}$. Giải thuật có độ phức tạp thời gian là $\mathcal{O}(n^3)$, ở đó n là số cung và trạng thái của $\mathcal{A}$.

Abstract. In this paper, we propose a new algorithm determining deciphering delay of regular language, which recognizes by a finite automaton $\mathcal{A}$. The algorithm has time complexity $\mathcal{O}(n^3)$, where n is the number of states and edges of $\mathcal{A}$.

1. GIỚI THIỆU

Trong các phép giải mã thông thường, khi xâu cần giải mã được đọc từ trái qua phải, thời điểm phát hiện thấy một từ mã trong xâu và thời điểm tất cả các từ mã trong xâu được xác định một cách chắc chắn là khác nhau. Khoảng thời gian trễ này được hình thức hóa bằng khái niệm độ trễ giải mã, khái niệm này xuất hiện rất sớm trong lý thuyết mã, như trong các công trình của Gilbert and Moore (1959) (xem [9]), của Levenshtein (1964) (xem [10]). Với khái niệm độ trễ giải mã thì lớp mã prefix là lớp mã có độ trễ giải mã bằng 0, từ đó độ trễ giải mã được sử dụng trong lý thuyết mã như là một tiêu chuẩn quan trọng để phân loại mã và là một tham số phản ánh độ khó trong quá trình giải mã. Đối với các ứng dụng, việc xác định chính xác độ trễ giải mã của một ngôn ngữ, cho phép các chương trình mật mã tăng hiệu quả thời gian và loại bỏ được thao tác quay lui trong quá trình giải mã. Do vai trò quan trọng của độ trễ giải mã, nhiều tác giả đã quan tâm nghiên cứu, một loạt các công trình như của Markov (1962) (xem [12]), Schützenberger (1966) (xem [11]), Choffrut (1979) (xem [13]), L. Staiger (1986) (xem [5]), J. Devolder (1994) (xem [3]), Stavros Konstantinidis (2002) (xem [8]), P.T. Huy-V.T. Nam (2002) (xem [14]), D. L. Van – I. Litovsky (2003) (xem [17]), N.D. Han - D.Q. Thang - H.N. Vinh (2010) (xem [16]), ... tập trung nghiên cứu tính chất của độ trễ giải mã.

Cho đến nay, việc tính toán chính xác độ trễ giải mã được nhiều tác giả quan tâm, phương pháp tính toán chủ yếu dựa trên cách tiếp cận tổ hợp và ý tưởng từ thủ tục kiểm tra tính chất mã của một ngôn ngữ (thủ tục Sardinas-Patterson) (xem [14, 16]), hay xác định một máy biến đổi có là hàm hay không (xem [8]). Giải thuật tốt nhất được biết cho tới nay là giải thuật của Stavros Konstantinidis (xem [8]), giải thuật này có độ phức tạp thời gian trong trường hợp

xấu nhất là $\mathcal{O}(n^4 log\, n)$. Để cho gọn, khi đề cập đến độ phức tạp thời gian thì luôn được hiểu là độ phức tạp thời gian trong trường hợp xấu nhất.

Bài báo đề xuất giải thuật mới xác định độ trễ giải mã của ngôn ngữ chính quy được đoán nhận bởi ôtômat hữu hạn $\mathcal{A}$, sử dụng kỹ thuật sao chép đồ thị, ghép và tích ôtômat, từ kỹ thuật này cho phép xây dựng giải thuật có độ phức tạp thời gian $\mathcal{O}(n^3)$.

Mục 2 sẽ trình bày một số khái niệm ngôn ngữ, mã, độ trễ giải mã, ôtômat và đồ thị để phục vụ cho các mục tiếp theo. Mục 3 trình bày các giải thuật mở rộng ôtômat. Mục 4 đề xuất giải thuật mới xác định độ trễ giải mã và cuối cùng là phần kết luận của bài báo.

2. MỘT SỐ KHÁI NIỆM

Trước hết, ta nhắc lại một số khái niệm và ký hiệu được sử dụng trong bài báo (chi tiết xem trong [2, 4]). Cho bảng chữ cái hữu hạn Σ, ký hiệu Σ^* là tập tất cả các từ hữu hạn trên Σ. Từ rỗng ký hiệu là ε và $\Sigma^+ = \Sigma^* - \{\varepsilon\}$. Tập con của Σ^* gọi là ngôn ngữ. Cho $L \subseteq \Sigma^*$ là ngôn ngữ trên Σ. Ta ký hiệu:

$L^* = L^0 \cup L^1 \cup \cdots$, ở đó $L^0 = \{\varepsilon\}$, $L^1 = L$, $L^2 = L^1L, \ldots, L^n = L^{n-1}L$,

$L^+ = L^1 \cup L^2 \cup \ldots$, ta có $L^* = L^+ \cup \{\varepsilon\}$.

Định nghĩa 2.1. Cho bảng chữ cái Σ, tập $L \subseteq \Sigma^*$ được gọi là mã nếu với mọi $m, n \geq 1$ và với mọi $x_1, \ldots, x_n, y_1, \ldots, y_m \in L$, nếu có $x_1 \cdots x_n = y_1 \cdots y_m$ thì suy ra $m = n$ và $x_i = y_i$ với $i = 1, \ldots, n$.

Nói cách khác, tập L là mã nếu mọi xâu trong L^+ chỉ có một phân tích duy nhất thành các xâu trong L. Do $\varepsilon.\varepsilon = \varepsilon$ nên mọi tập mã đều không chứa xâu rỗng.

Định nghĩa 2.2. Cho $L \subseteq \Sigma^+$, L được gọi là có độ trễ giải mã hữu hạn nếu tồn tại một số nguyên $d \geq 0$ sao cho:

$$\forall x, x' \in L, \forall y \in L^d, \forall u \in \Sigma^*, xyu \in x'L^* \Rightarrow x = x'. \tag{2.1}$$

Dễ thấy rằng nếu hệ thức (2.1) thỏa mãn với d hữu hạn nào đó thì nó cũng đúng với mọi $d' \geq d$. Nếu L có độ trễ giải mã hữu hạn thì số nguyên nhỏ nhất thoả hệ thức (2.1) gọi là *độ trễ giải mã* của L, số nguyên bất kỳ thỏa hệ thức (2.1) gọi là *độ trễ giải mã yếu* của L. Ngược lại, nếu L không có độ trễ giải mã hữu hạn thì L gọi là có *độ trễ giải mã vô hạn*.

Ôtômat hữu hạn là một bộ 5 $\mathcal{A} = (Q, \Sigma, E, I, F)$, ở đó: Q là tập hữu hạn các trạng thái, Σ là bảng chữ cái hữu hạn, $E \subseteq Q \times \Sigma \times Q$ là tập hữu hạn các cung (không chứa cung rỗng), $I \subseteq Q$ là tập các trạng thái đầu, $F \subseteq Q$ là tập các trạng thái kết thúc.

Cho cung $e = (q_1, a, q_2) \in E$, ta nói rằng e rời q_1 đến q_2, q_1 là đầu của e ký hiệu $p[e]$, q_2 là cuối của e ký hiệu $n[e]$, a là nhãn của e ký hiệu $l[e]$. Cho $q \in Q$, ta ký hiệu $E[q]$ là tập các cung rời q. Trong bài báo này ta còn xét dạng ôtômat hữu hạn mở rộng có chuyển rỗng, gọi tắt là ε-ôtômat, khi mà nhãn của một cung e nào đó có thể là từ rỗng ε.

Cho $\pi = e_1 \cdots e_k \in E^*$, ở đó $e_1 = (p_0, a_1, p_1), e_2 = (p_1, a_2, p_2), \ldots, e_k = (p_{k-1}, a_k, p_k)$ được gọi là đường đi từ p_0 đến p_k. Một đường đi thành công trong ôtômat $\mathcal{A}$ là một đường đi từ trạng thái đầu đến trạng thái kết thúc. Từ $w = a_1a_2 \cdots a_k$ gọi là nhãn của đường đi π, tập hợp nhãn của các đường đi thành công trong ôtômat $\mathcal{A}$ gọi là ngôn ngữ đoán nhận bởi $\mathcal{A}$, ký hiệu là $\mathcal{L}(\mathcal{A})$.

Đồ thị có hướng G là một cặp $G = (V, E)$, V là tập các đỉnh, E là tập các cặp có thứ tự gồm hai phần tử của V gọi là các cung. Nếu $e = (u, v)$ là cung của đồ thị có hướng G thì ta nói đỉnh v kề u, ta kí hiệu $Next(u) = \{v \in V \mid (u, v) \in E\}$. Nếu V, E là hữu hạn thì ta gọi G là đồ thị hữu hạn có hướng.

Đường đi từ đỉnh u đến đỉnh v trên đồ thị có hướng G là dãy đỉnh $x_0, \ldots, x_n$, trong đó $u = x_0, v = x_n, (x_i, x_{i+1}) \in E, i = 0, \ldots, n-1$. Đỉnh u gọi là đến được đỉnh v nếu có một đường đi từ đỉnh u đến đỉnh v.

3. MỘT SỐ GIẢI THUẬT MỞ RỘNG ÔTÔMAT

Trong mục này, ta nhắc lại về ôtômat thu gọn và xem xét một số kỹ thuật mở rộng ôtômat từ ôtômat hữu hạn cho trước. Các ôtômat này được dùng cho việc thiết kế giải thuật tính độ trễ giải mã ở phần sau. Trước tiên, về ôtômat thu gọn: Cho ôtômat hữu hạn $\mathcal{A} = (Q, \Sigma, E, I, F)$, một trạng thái $q \in Q$ gọi là đạt được nếu tồn tại đường đi từ một trạng thái đầu đến q, một trạng thái $q \in Q$ gọi là đối đạt được nếu tồn tại đường đi từ q đến một trạng thái kết thúc. Ôtômat hữu hạn $\mathcal{A}$ gọi là thu gọn nếu tất cả các trạng thái của $\mathcal{A}$ là đạt được và cũng là đối đạt được. Giải thuật kinh điển xây dựng ôtômat thu gọn được thực hiện bởi hàm ký hiệu là Trim($\mathcal{A}$), có độ phức tạp thời gian là $\mathcal{O}(|Q| + |E|)$ và $\mathcal{L}(\mathcal{A}) = \mathcal{L}(Trim(\mathcal{A}))$. Tiếp theo, ta xem xét một số kỹ thuật mở rộng ôtômat dưới đây.

3.1. Ôtômat lưỡng cực

Ôtômat hữu hạn $\mathcal{A}$ được gọi là ôtômat lưỡng cực nếu $\mathcal{A}$ có một trạng thái đầu và một trạng thái kết thúc khác nhau, không có cung đến trạng thái đầu và không có cung rời trạng thái kết thúc.

Cho ôtômat hữu hạn $\mathcal{A} = (Q, \Sigma, E, I, F)$ đoán nhận ngôn ngữ $L = \mathcal{L}(\mathcal{A}) \subseteq \Sigma^+$. Ta xây dựng ôtômat lưỡng cực $\mathcal{A}' = (Q', \Sigma, E', I', F')$ đoán nhận L từ ôtômat hữu hạn $\mathcal{A}$ như sau

i) $Q' = Q \cup \{s, f\}, s, f \notin Q$ và $s \neq f, I' = \{s\}, F' = \{f\}$.

ii) $E' = E_1 \cup \{(s, a, q) \mid (p, a, q) \in E_1, p \in I\}$ với $E_1 = E \cup \{(p, a, f) \mid (p, a, q) \in E, q \in F\}$.

Để đơn giản, ta ký hiệu $\mathcal{A}' = (Q', \Sigma, E', s, f)$ và gọi s là cực vào (trạng thái đầu), f là cực ra (trạng thái kết thúc). Giải thuật xây dựng ôtômat lưỡng cực $\mathcal{A}'$ từ ôtômat hữu hạn $\mathcal{A}$ thực hiện bởi hàm ký hiệu là $D(\mathcal{A})$ có độ phức tạp thời gian $\mathcal{O}(|Q| + |E|)$.

Cho ôtômat lưỡng cực $\mathcal{A}$ đoán nhận ngôn ngữ $L = \mathcal{L}(\mathcal{A}) \subseteq \Sigma^+$. Ta có thể xây dựng ε-ôtômat *mở rộng* $\mathcal{A}'$ đoán nhận L^+ (tức là $L^+ = \mathcal{L}(\mathcal{A}')$), bằng cách bổ sung một cung rỗng đi từ cực ra tới cực vào của $\mathcal{A}$. Giải thuật xây dựng ε-ôtômat mở rộng thực hiện bởi hàm ký hiệu là $Ex(\mathcal{A})$.

Cho ε-ôtômat mở rộng $\mathcal{A}_1 = (Q_1, \Sigma, E_1, s_1, f_1)$ đoán nhận L^+. Xét ε-ôtômat $\mathcal{A}_2 = (Q_2, \Sigma, E_2, s_2, f_2)$, ở đó $Q_2 = \{s_2\}, s_2 = f_2$ ($\mathcal{A}_2$ có duy nhất một trạng thái, vừa là trạng thái đầu và cũng là trạng thái kết thúc), E_2 có $|\Sigma| + 1$ cung đi từ trạng thái duy nhất trong $\mathcal{A}_2$ đến chính nó với nhãn là ký tự thuộc $\Sigma \cup \{\varepsilon\}$, ta có $\Sigma^* = \mathcal{L}(\mathcal{A}_2)$. Ta xây dựng ε-ôtômat *ghép* $\mathcal{A} = (Q, \Sigma, E, s, f)$, bằng cách bổ sung một cung rỗng đi từ trạng thái kết thúc của $\mathcal{A}_1$ đến trạng thái duy nhất của $\mathcal{A}_2$, ta lấy $Q = Q_1 \cup Q_2, E = E_1 \cup E_2 \cup (f_1, \varepsilon, f_2), s = s_1, f = f_2$, trạng thái f_1 gọi là *trạng thái ghép* trong ε-ôtômat ghép $\mathcal{A}$. ε-ôtômat ghép $\mathcal{A}$ đoán nhận ngôn ngữ $L^+\Sigma^*$, giải thuật xây dựng ε-ôtômat ghép thực hiện bởi hàm ký hiệu là $Graft(\mathcal{A}_1)$.

Nhận xét 3.1. Cho ôtômat hữu hạn $\mathcal{A} = (Q, \Sigma, E, I, F)$ với $c = |\Sigma|$ là hằng số. Khi đó, số trạng thái của $D(\mathcal{A}), Ex(D(\mathcal{A}))$ và $Graft(Ex(D(\mathcal{A})))$ không quá $|Q| + 3$ do đó có cỡ $\mathcal{O}(|Q|)$. Số cung của $D(\mathcal{A})$ không quá $4|E|$, của $Ex(D(\mathcal{A}))$ không quá $4|E| + 1$ và của $Graft(Ex(D(\mathcal{A})))$ không quá $4|E| + c + 3$, do đó cũng có cỡ $\mathcal{O}(|E|)$.

3.2. Ôtômat tích

Phép lấy tích ôtômat (xem [6, 7]) được sử dụng trong nhiều ứng dụng để tạo ra ôtômat phức hợp từ những ôtômat đơn giản. Cho hai ε-ôtômat mở rộng, hay ε-ôtômat ghép như đã xét ở trên, $\mathcal{A}_1 = (Q_1, \Sigma, E_1, s_1, f_1)$ và $\mathcal{A}_2 = (Q_2, \Sigma, E_2, s_2, f_2)$. Khi đó, tích của $\mathcal{A}_1$ và $\mathcal{A}_2$ là ε-ôtômat ký hiệu $Prod(\mathcal{A}_1, \mathcal{A}_2) = (Q, \Sigma, E, (s_1, s_2), (f_1, f_2))$, trong đó $Q \subseteq Q_1 \times Q_2$, E được xác định theo quy tắc sau:

i) $\forall (q_1, a, p_1) \in E_1, \forall (q_2, a, p_2) \in E_2, a \in \Sigma$ thì $((q_1, q_2), a, (p_1, p_2)) \in E$;

ii) $\forall (q_1, \varepsilon, p_1) \in E_1, \forall (q_2, \varepsilon, p_2) \in E_2$ thì $((q_1, q_2), \varepsilon, (p_1, p_2)) \in E$;

iii) $\forall (q_1, \varepsilon, p_1) \in E_1, \forall (q_2, a, p_2) \in E_2, a \in \Sigma$ thì $((q_1, q_2), \varepsilon, (p_1, q_2)) \in E$;

iv) $\forall (q_1, a, p_1) \in E_1, \forall (q_2, \varepsilon, p_2) \in E_2, a \in \Sigma$ thì $((q_1, q_2), \varepsilon, (q_1, p_2)) \in E$;

v) E chỉ chứa các cung đã xét ở bốn trường hợp trên.

Giải thuật có thể bắt đầu từ trạng thái (s_1, s_2), rồi thực hiện theo quy tắc ở trên xác định các trạng thái và cung của ôtômat tích. Dưới đây là giải thuật.

Function Prod($\mathcal{A}_1$, $\mathcal{A}_2$)

Input: $\mathcal{A}_1$, $\mathcal{A}_2$ là ε-ôtômat mở rộng, hoặc ε-ôtômat ghép.
Output: ε-ôtômat $\mathcal{A} = (Q, \Sigma, E, s, f)$ là tích của $\mathcal{A}_1$ và $\mathcal{A}_2$.
{S là hàng đợi, CQpush, CQPop là phép bổ sung và loại bỏ một phần tử trên S.}

1. $Q = \{(s_1, s_2)\}$; CQpush$(S, (s_1, s_2))$;
   $s = (s_1, s_2); f = (f_1, f_2); E = \emptyset$;
2. while $S \neq \emptyset$ do
3. $(q_1, q_2) \longleftarrow$ CQPop(S);
4. for each (e_1, e_2) in $E[q_1] \times E[q_2]$ do
5. $t = true; label = \varepsilon$;
6. case
   $l[e_1] = l[e_2] \wedge l[e_1] \neq \varepsilon : (p_1, p_2) = (n[e_1], n[e_2]); label = l[e_1]$;
   $l[e_1] = l[e_2] = \varepsilon : (p_1, p_2) = (n[e_1], n[e_2])$;
   $l[e_1] = \varepsilon \wedge l[e_2] \neq \varepsilon : (p_1, p_2) = (n[e_1], q_2)$;
   $l[e_1] \neq \varepsilon \wedge l[e_2] = \varepsilon : (p_1, p_2) = (q_1, n[e_2])$;
   else $t = false$;
7. end case;
8. if $t = true$ then {*Nếu* $(p_1, p_2) \notin Q$ *thì* $Q = Q \cup (p_1, p_2)$}
   if BelongQ$(p_1, p_2) = 0$ then
   BelongQ$(p_1, p_2) = 1$;
   CQPush$(S, (p_1, p_2))$;

$$E = E \cup \{((q_1, q_2), label, (p_1, p_2))\};$$

9. return $\mathcal{A}$.

Nhận xét 3.2. *i)* Tương tự như phân tích của Mohri trong [6, 7], giải thuật có độ phức tạp thời gian là $\mathcal{O}((|Q_1| + |E_1|)(|Q_2| + |E_2|))$.

ii) Cho $\mathcal{A}_2 = (Q_2, \Sigma, E_2, s_2, f_2)$ là ε-ôtômat mở rộng đoán nhận L^+. ε-ôtômat ghép $\mathcal{A}_1 = Graft(\mathcal{A}_2) = (Q_1, \Sigma, E_1, s_1, f_1)$ có trạng thái ghép f_G (ở đó ta đổi tên trạng thái ghép từ f_2 thành f_G, trạng thái đầu từ s_2 thành s_1). Trên ε-ôtômat tích $Prod(\mathcal{A}_1, \mathcal{A}_2)$, nhãn của đường đi giữa hai trạng thái kế tiếp (f_G, q_i) và (f_G, q_j) (hoặc (p_i, f_2) và (p_j, f_2), hoặc (s_1, q_i) và (f_G, q_j), hoặc (p_i, s_2) và (p_j, f_2)) là từ thuộc L.

4. XÁC ĐỊNH ĐỘ TRỄ GIẢI MÃ CỦA NGÔN NGỮ CHÍNH QUY

Cho ngôn ngữ chính quy L được đoán nhận bởi ôtômat hữu hạn $\mathcal{A}$, để xác định độ trễ giải mã của L, trước hết ta giải quyết bài toán bổ trợ trên đồ thị như dưới đây.

4.1. Bài toán về chu trình hợp lệ trên đồ thị

Cho đồ thị hữu hạn có hướng (có thể có khuyên) $G = (V, E)$ với hai đỉnh đặc biệt là đỉnh khởi đầu s và đỉnh kết thúc $f (s \neq f)$, tập $U \subseteq V$ gọi là tập đỉnh khóa (đỉnh s và f không thuộc tập U), các đỉnh còn lại gọi là đỉnh không khóa.

Cho đường đi π gồm dãy đỉnh $v_1, \ldots, v_j, \ldots, v_i, \ldots, v_k$ (với $s = v_1$) trên đồ thị G. Nếu có $1 < i < k$ sao cho $v_i \in U, v_k = v_j, 1 \leq j \leq i$ thì π gọi là *chu trình hợp lệ* trên đồ thị G. Đỉnh $v_l, j \leq l \leq k$ gọi là *đỉnh trong* của chu trình hợp lệ. Đường đi π gọi là *đi qua* đỉnh v, nếu v là một trong các đỉnh $v_l, 2 \leq l \leq k$. π cũng được gọi là *chứa* đỉnh v, nếu v là một trong các đỉnh $v_l, 1 \leq l \leq k$.

Bài toán 1. Cho đồ thị hữu hạn có hướng $G = (V, E)$ như ở trên. Cho biết trên G có tồn tại chu trình hợp lệ hay không.

Để giải bài toán trên, trước hết ta xây dựng đồ thị sao chép $G' = (V', E')$ từ đồ thị $G = (V, E)$ nhờ kỹ thuật sao chép đồ thị như sau:

i) Với $v \in V$ sao chép thành hai đỉnh $(v, 1), (v, 2)$ của V'.

ii) Với $(u, v) \in E$:

+ Sao chép thành hai cung $((u, 1), (v, 1)), ((u, 2), (v, 2))$ của E';

+ Nếu $u \in U$ thì bổ sung cung $((u, 1), (v, 2))$ vào E';

Hàm xây dựng đồ thị sao chép G' ký hiệu là XCopy(G), dưới đây là giải thuật:

Function XCopy(G)

Input: Đồ thị hữu hạn có hướng G.
Output: Đồ thị sao chép G' từ đồ thị G.
{*Giải thuật dùng mảng: key*$[q] = 1$ *khi và chỉ khi đỉnh* $q \in U$}
1. $V' = \emptyset; E' = \emptyset;$
2. for each u in V do
   $V' = V' \cup \{(u,1),(u,2)\};$
3. for each u in V do
4. for each v in $Next(u)$ do
   $E' = E' \cup \{((u,1),(v,1)),((u,2),(v,2))\};$
   if $key[u] = 1$ then
   $E' = E' \cup \{((u,1),(v,2))\};$
5. return G'.

Nhận xét 4.1. *i)* Đồ thị sao chép G' có $|V'| = 2|V|, |E'| \leq 3|E|$.

ii) Tập các đỉnh dạng (v,k) cảm sinh trong G' đồ thị con $G_k, k = 1,2$. Mỗi đồ thị con G_k đều đẳng cấu với G. G' thực chất là sự kết nối có chọn lọc của hai đồ thị con G_1, G_2, chỉ có các cung đi từ G_1 đến G_2 mà không có chiều ngược lại.

Hiển nhiên ta có bổ đề sau:

Bổ đề 4.1. *Giải thuật XCopy có độ phức tạp thời gian* $\mathcal{O}(|V| + |E|)$.

Vai trò của đồ thị sao chép G' trong việc xác định chu trình hợp lệ trên G, được cho bởi định lý sau:

Định lý 4.1. *Cho đồ thị hữu hạn* $G = (V,E)$, *có đỉnh khởi đầu* $s \in V$, *đỉnh kết thúc* $f \in V (s \neq f)$, *với tập đỉnh khóa* U *(đỉnh* s *và* f *không thuộc tập* U*) và* $G' = XCopy(G)$. *Trên* G *tồn tại chu trình hợp lệ khi và chỉ khi trên* G' *tồn tại đường đi* π *từ đỉnh* $(s,1)$ *đến đỉnh* $(v,2)$ *và* $(v,1)$ *thuộc* π.

Chứng minh. ($\Rightarrow$) Trên G tồn tại chu trình hợp lệ: $u_1, \ldots, u_j, \ldots, u_i, \ldots, u_k$, (với $s = u_1$), ta có $1 < i < k$ sao cho $u_i \in U$ và $u_j = u_k, 1 \leq j \leq i$. Theo cách xây dựng đồ thị sao chép G', trên nó tồn tại đường đi π sau:

$$(u_1,1), \ldots, (u_j,1), \ldots, (u_i,1), (u_{i+1},2), \ldots, (u_k,2),$$

ở đó $(s,1) = (u_1,1), (u_k,2) = (v,2)$. Do $u_j = v, 1 \leq j \leq i$ suy ra $(v,1) \in \pi$.

($\Leftarrow$) Trên G' tồn tại đường đi π từ đỉnh $(s,1)$ đến đỉnh $(v,2)$ và $(v,1) \in \pi$. Theo cách xây dựng đồ thị sao chép G', khi đó đường đi π có dạng sau:

$$(u_1,1), \ldots, (u_j,1), \ldots, (u_i,1), (u_{i+1},2), \ldots, (u_k,2),$$

ở đó $(s,1) = (u_1,1), (u_k,2) = (v,2)$ và $u_i \in U$ với $1 < i < k$. Do $(v,1) \in \pi$, nên $v = u_j$ với $1 \leq j \leq i$. Tương ứng với đường đi π, trên G có đường đi sau đây: $u_1, \ldots, u_j, \ldots, u_i, u_{i+1}, \ldots, u_k$, ở đó $s = u_1, u_k = v, u_i \in U$ với $1 < i < k$, và $u_k = u_j$ với $1 \leq j \leq i$. Hay trên G có chu trình hợp lệ. ■

Định lý 4.1 là cơ sở cho ta xây dựng giải thuật xác định có chu trình hợp lệ trên G hay không. Ta thực hiện tô màu các đỉnh trên đồ thị sao chép G'. Trong kỹ thuật tô màu, mỗi

đỉnh có thuộc tính $mark$ để cho biết đỉnh đó được tô màu gì, ta sử dụng 4 màu như sau:

+ Đỉnh (u,i) được tô màu WHITE (hay $mark[(u,i)]$ = WHITE) để biểu diễn đỉnh (u,i) chưa được "thăm".

+ Tô màu một đỉnh đang xét: Nếu nó là đỉnh dạng $(u,1)$ thì được tô màu BLUE, ngược lại thì được tô màu GRAY. Đặc biệt khi một đỉnh đang được xét $(u,1)$ được tô màu BLUE thì đỉnh đồng bộ tương ứng $(u,2)$ cũng được tô màu BLUE. Trên con đường duyệt đồ thị G', khởi đầu từ đỉnh $(s,1)$, nếu đến được đỉnh $(u,2)$ tô màu BLUE thì theo Định lý 4.1 trên G có chu trình hợp lệ.

+ Đỉnh được tô màu BLACK cho biết đỉnh bị loại.

Giải thuật có thể được mô tả như sau:

i) Ban đầu các đỉnh trên G' được tô màu WHITE.

ii) Sau đó ta gọi đến giải thuật *Visit* từ một đỉnh cho trước $(s,1)$, giải thuật *Visit* được cải tiến từ giải thuật DFS (Depth First Search) (xem [15]) để tô màu trên G'.

Function Visit$(graph\, G', vertex\, (u,i))$

1. if $i = 1$ then
   $mark[(u,1)] = mark[(u,2)]$ = BLUE
   else $mark[(u,i)]$ = GREY
2. for each (v,j) in $Next((u,i))$ do
   if $mark[(v,j)]$ = BLUE and $j = 2$ then return TRUE;
   if $mark[(v,j)]$= WHITE then
   if $Visit(G',(v,j))$ then return TRUE
3. $mark[(u,i)]$ = BLACK
   if $i = 1$ and $mark[(u,2)]$ = BLUE then $mark[(u,2)]$ = WHITE
4. return FALSE

Function ContainsCycle$(graph\, G', vertex\, (u,i))$

1. for each $(u,1)$ in V' do
   $mark[(u,1)] = mark[(u,2)]$ = WHITE
2. return $Visit(G',(u,i))$;

Nhận xét 4.2. *i*) Giải thuật Visit phát hiện chu trình hợp lệ (nếu có) trên G.

ii) Bước thực hiện giải thuật Visit chỉ là cải tiến giải thuật DFS, nên giải thuật ContainsCycle có độ phức tạp thời gian là $\mathcal{O}(|V'| + |E'|) = \mathcal{O}(2|V| + 3|E|) = \mathcal{O}(|V| + |E|)$.

4.2. Giải thuật xác định độ trễ giải mã

Bài toán 2. Cho ngôn ngữ chính quy $L \subseteq \Sigma^*$ được đoán nhận bởi ôtômat hữu hạn $\mathcal{A}$. Hãy xây dựng giải thuật xác định độ trễ giải mã của L.

Giả sử $\mathcal{A} = (Q, \Sigma, E, I, F)$ và $L = \mathcal{L}(\mathcal{A})$. Ta xét các trường hợp sau:

a) Nếu L không là mã thì kết thúc: Việc kiểm tra L có là mã hay không có thể áp dụng giải thuật của R. McCloskey (xem [1]). Ta ký hiệu hàm thực hiện kiểm tra là CodeAlgo($\mathcal{A}$), trả lại giá trị TRUE nếu $L = \mathcal{L}(\mathcal{A})$ là mã, ngược lại hàm trả lại giá trị FALSE, giải thuật có độ phức tạp thời gian $\mathcal{O}((|Q| + |E|)^2)$.

b) Nếu L là mã thì $L \subseteq \Sigma^+$: Ta xây dựng $\mathcal{A}_2$ =Ex(Trim($D(\mathcal{A})$)) = $(Q_2, \Sigma, E_2, s_2, f_2)$, $\mathcal{A}_1$ =Graft($\mathcal{A}_2$) = $(Q_1, \Sigma, E_1, s_1, f_1)$ có trạng thái ghép f_G (ở đó ta đổi tên trạng thái ghép từ f_2 thành f_G, trạng thái đầu từ s_2 thành s_1). Lấy tích $\mathcal{A}_1$ với $\mathcal{A}_2$ ta có $Prod(\mathcal{A}_1, \mathcal{A}_2)$.

Ta xem ε-ôtômat Prod($\mathcal{A}_1, \mathcal{A}_2$) như một đồ thị hữu hạn có hướng $G(Prod(\mathcal{A}_1, \mathcal{A}_2)$ xác định một đồ thị có hướng G), các trạng thái là các đỉnh của đồ thị, có đỉnh khởi đầu (s_1, s_2), đỉnh kết thúc (f_1, f_2), đỉnh (f_G, f_2) gọi là *đỉnh ghép tích* trên đồ thị G, tập đỉnh khóa $U = \{(f_G, q) \in Q_1 \times Q_2 \mid q \neq f_2, s_2\}$, một cung của $Prod(\mathcal{A}_1, \mathcal{A}_2)$ xác định một cung của đồ thị, cung đi đến đỉnh khóa được gán trọng số 1 các cung còn lại được gán trọng số 0.

Hình 4.1. Tính độ trễ giải mã của L

Phương pháp tính độ trễ giải mã có thể mô tả như sau: L có độ trễ giải mã $d \geq 0$, khi đó d là số nguyên nhỏ nhất sao cho:

$$\forall x, x' \in L, \forall y \in L^d, \forall u \in \Sigma^*, xyu \in x'L^* \Rightarrow x = x'$$

(hay d là số nguyên lớn nhất mà: $\exists x, x' \in L, x \neq x', \exists y = y_1 y_2 \cdots y_{d-1} \in L^{d-1}, y_i \in L, \exists u \in \Sigma^*$ sao cho $xyu \in x'L^*$ (ta cũng có $xyu = x'y'$ với $y' = y'_1 y'_2 \ldots y'_n \in L^*, y'_i \in L$)). Khi đó, ta thực hiện tích ε-ôtômat $\mathcal{A}_1$ đoán nhận $L^+\Sigma^*$ với ε-ôtômat $\mathcal{A}_2$ đoán nhận L^+. Trên đồ thị G xác định bởi ε-ôtômat tích của $\mathcal{A}_1$ với $\mathcal{A}_2$, ta xác định đường đi từ đỉnh khởi đầu đến đỉnh kết thúc có nhiều đỉnh khóa (f_G, q_i) nhất. Số đỉnh khóa tìm được chính là độ trễ giải mã d của L, như mô tả bởi Hình 4.1.

Trong trường hợp L là mã, ta thiết lập các kết quả dưới đây:

Bổ đề 4.2. *Cho đồ thị G xác định như ở trên. Nếu trên G có chu trình hợp lệ thì tồn tại chu trình hợp lệ thỏa mãn:*

i) *Không chứa đỉnh ghép tích và không đi qua đỉnh khởi đầu.*

ii) *Tồn tại đỉnh trong đến được đỉnh kết thúc.*

Chứng minh. Trên G có chu trình hợp lệ π gồm dãy đỉnh $v_1, \ldots, v_j, \ldots, v_i, \ldots, v_k$ với $1 < i < k, v_i \in U, v_k = v_j, 1 \leq j \leq i$, hay π có dạng:

$$(p_1, q_1), \ldots, (p_j, q_j), \ldots, (f_G, q_i), \ldots, (p_k, q_k),$$

với $v_1 = (p_1, q_1) = (s_1, s_2), v_j = (p_j, q_j), v_i = (f_G, q_i), v_k = (p_k, q_k)$.

i) Khi đó chu trình hợp lệ π không chứa đỉnh ghép tích (f_G, f_2) và không đi qua đỉnh khởi đầu (s_1, s_2), thật vậy:

+ Giả sử π đi qua đỉnh khởi đầu $v_l = v_1$ với $1 < l \leq j$, khi đó ta xét chu trình hợp lệ π bắt đầu từ đỉnh khởi đầu v_l.

+ Giả sử π chứa đỉnh ghép tích $v_l = (f_G, f_2)$ với $1 \le l \le j$, trên cơ sở xây dựng các ε-ôtômat $\mathcal{A}_1$, $\mathcal{A}_2$ và phép tích ôtômat, đỉnh kế tiếp có thể đi đến từ đỉnh ghép tích (f_G, f_2) trên G, chỉ có thể là đỉnh (f_1, s_2) hoặc đỉnh khởi đầu (s_1, s_2). Đỉnh (f_1, s_2) không thể là đỉnh kế tiếp của đỉnh ghép tích (f_G, f_2), vì trên π có đỉnh khóa (f_G, q_i), mà từ đỉnh (f_1, s_2) không thể đến được đỉnh khóa (f_G, q_i). Vậy, kế tiếp đỉnh ghép tích (f_G, f_2) chỉ có thể là đỉnh (s_1, s_2), ta xét chu trình hợp lệ π bắt đầu từ đỉnh (s_1, s_2) này.

+ Giả sử π chứa đỉnh ghép tích $v_l = (f_G, f_2)$ với $j \le l \le k$, khi đó ta có đường đi trên chu trình là:

$$(f_G, f_2) \xrightarrow{x_1} (f_G, q_i) \xrightarrow{x_2} (p_k, q_k) \xrightarrow{x_3} (f_G, f_2),$$
$$(\text{hoặc } (f_G, f_2) \xrightarrow{x_1} (p_k, q_k) \xrightarrow{x_2} (f_G, q_i) \xrightarrow{x_3} (f_G, f_2))$$

vì (f_G, q_i) là đỉnh khóa nên $q_i \ne f_2, s_2$. Từ Nhận xét 3.2 thì $x_1, x_2x_3 \in L^+$ (hoặc $x_1x_2, x_3 \in L^+$), suy ra, xâu $x_1x_2x_3$ có hai phân tích khác nhau trong L, khi đó L không là mã, điều này trái với giả thiết L là mã.

+ Giả sử π đi qua đỉnh khởi đầu $v_l = (s_1, s_2)$ với $j \le l \le k$, khi đó đỉnh v_l có cung đi đến nó, trên cơ sở xây dựng các ε-ôtômat $\mathcal{A}_1$, $\mathcal{A}_2$ và phép tích ôtômat thì đỉnh kế trước v_l trong chu trình π là đỉnh (f_G, f_2), như phân tích ở trên, điều này là mâu thuẫn.

ii) Chu trình π có ít nhất một đỉnh trong đến được đỉnh kết thúc (f_1, f_2) trên G. Thật vậy, trên π có đỉnh trong $v_l = (f_G, q_i)$ với $j \le l \le k, q_i \ne f_2, s_2$, trên cơ sở xây dựng các ε-ôtômat $\mathcal{A}_1$, $\mathcal{A}_2$ và phép tích ôtômat, từ đỉnh (f_G, q_i) ta đến được đỉnh (f_1, q_i) bằng một cung với nhãn ε. Do $\mathcal{A}_2 = \text{Ex}(Trim(D(\mathcal{A})))$, nên từ trạng thái bất kỳ q_i trên $\mathcal{A}_2$ đều đến được trạng thái f_2, và từ f_1 trên $\mathcal{A}_1$ chỉ có các cung đến chính nó với nhãn là ký tự thuộc $\Sigma \cup \{\varepsilon\}$, vậy theo tích ôtômat thì trên G phải có đường đi từ (f_G, q_i) đến (f_1, f_2). ■

Định lý 4.2. *Cho ngôn ngữ chính quy $L \subseteq \Sigma^+$ là mã được đoán nhận bởi ôtômat hữu hạn $\mathcal{A}$. Cho $\mathcal{A}_1 = Graft(\mathcal{A}_2)$ có trạng thái ghép f_G với $\mathcal{A}_2 = Ex(Trim(D(\mathcal{A})))$ và $Prod(\mathcal{A}_1, \mathcal{A}_2)$ xác định một đồ thị G như ở trên.*

i) L có độ trễ giải mã vô hạn khi và chỉ khi trên G có chu trình hợp lệ.

ii) L có độ trễ giải mã $d \ge 0$ khi và chỉ khi trên G có đường đi từ đỉnh khởi đầu đến đỉnh kết thúc với trọng số lớn nhất là d.

Chứng minh. i) ($\Rightarrow$) L có độ trễ giải mã vô hạn, khi đó xét với một số nguyên không âm bất kỳ d, ta có $\exists x, x' \in L, x \ne x', \exists y = y_1y_2 \cdots y_d \in L^d, y_i \in L, \exists u = u_1u_2 \cdots u_l \in \Sigma^*, u_i \in \Sigma$ sao cho $xyu \in x'L^*$. Với từ xyu, trong $\mathcal{A}_1$ có đường đi π nhãn xyu:

$$s_1 \xrightarrow{x} f_G \xrightarrow{\varepsilon} s_1 \xrightarrow{y_1} f_G \xrightarrow{\varepsilon} s_1 \cdots s_1 \xrightarrow{y_d} f_G \xrightarrow{\varepsilon} f_1 \xrightarrow{u_1} f_1 \cdots f_1 \xrightarrow{u_l} f_1.$$

Tương tự vậy, với $xyu = x'y'_1y'_2 \cdots y'_n \in x'L^*, y'_i \in L$ trong $\mathcal{A}_2$ có đường đi θ nhãn xyu:

$$s_2 \xrightarrow{x'} f_2 \xrightarrow{\varepsilon} s_2 \xrightarrow{y'_1} f_2 \xrightarrow{\varepsilon} s_2 \cdots s_2 \xrightarrow{y'_n} f_2.$$

Khi đó, trên đồ thị G có đường đi ρ được tạo nên từ π và θ, khởi đầu từ (s_1, s_2) và kết thúc là (f_1, f_2) như sau:

$$(s_1, s_2), \ldots, (f_G, q_i), \ldots, (p_j, f_2), \ldots, (f_1, f_2),$$

do L là mã nên $q_i \neq f_2, s_2$ và $p_j \neq f_G, s_1$, do ta có thể chọn d tùy ý đủ lớn (chẳng hạn d lớn hơn số trạng thái của $\mathcal{A}_1$), số trạng thái của $\mathcal{A}_1$ là hữu hạn, nên trong ρ phải tồn tại hai đỉnh (f_G, q_i) trùng nhau, hay nói cách khác trong G có chu trình hợp lệ.

($\Leftarrow$) Trên G có chu trình hợp lệ π gồm dãy đỉnh $v_1, \ldots, v_j, \ldots, v_i, \ldots, v_k$ với $1 < i < k$, $v_i \in U, v_k = v_j, 1 \leq j \leq i$, hay π là đường đi có nhãn như sau:

$$(p_1, q_1) \xrightarrow{x_1} (p_j, q_j) \xrightarrow{x_2} (f_G, q_i) \xrightarrow{x_3} (p_k, q_k)$$

với $v_1 = (p_1, q_1) = (s_1, s_2), v_j = (p_j, q_j), v_i = (f_G, q_i) \in U, v_k = (p_k, q_k)$.

Theo Bổ đề 4.2 ta có π không chứa đỉnh ghép tích (f_G, f_2), không đi qua đỉnh khởi đầu (s_1, s_2) và tồn tại đỉnh trong (f_G, q_i) đến được đỉnh kết thúc (f_1, f_2) trên G. Khi đó trên G có đường đi ρ có nhãn sau đây:

$$\begin{gathered}(s_1, s_2) \xrightarrow{x_1} (p_j, q_j) \xrightarrow{x_2} (f_G, q_i) \xrightarrow{x_3} (p_j, q_j) \xrightarrow{x_2} (f_G, q_i) \xrightarrow{x_3} \cdots \\ \cdots \xrightarrow{x_2} (f_G, q_i) \xrightarrow{x_3} (p_j, q_j) \xrightarrow{x_2} (f_G, q_i) \xrightarrow{\varepsilon} (f_1, q_i) \xrightarrow{u} (f_1, f_2),\end{gathered}$$

ta cũng có trên ρ không chứa đỉnh (f_G, f_2), không đi qua đỉnh (s_1, s_2). Không làm mất tính tổng quát, ta giả sử (f_G, q_i) là đỉnh khóa đầu tiên trên ρ. Theo Nhận xét 3.2 thì ta có $x_3 x_2 \in L^+$. Vậy, với d lớn tùy ý, $\exists x = x_1 x_2, x \neq x' \in L, \exists y = (x_3 x_2)^d \in L^+, \exists u \in \Sigma^*$ sao cho $xyu \in x'L^*$, hay nói cách khác L có độ trễ giải mã vô hạn.

ii) ($\Rightarrow$) Nếu L có độ trễ giải mã d

+ Ta xét với $d = 0$ thì

$$\forall x, x' \in L, \forall u \in \Sigma^*, xu \in x'L^* \Rightarrow x = x'.$$

Trên cơ sở xây dựng các ε-ôtômat $\mathcal{A}_1$, $\mathcal{A}_2$ và phép tích ôtômat. Nếu $d = 0$ thì trên G, mọi đường đi từ đỉnh khởi đầu đến đỉnh kết thúc không có đỉnh khóa, khi đó mọi cung trên các đường đi này đều có trọng số 0, vậy đường đi từ đỉnh khởi đầu đến đỉnh kết thúc có trọng số lớn nhất là $d = 0$.

+ Ta xét với $d > 0$

$$\forall x, x' \in L, \forall y \in L^d, \forall u \in \Sigma^*, xyu \in x'L^* \Rightarrow x = x',$$

hay d là lớn nhất thỏa mãn:

$$\begin{gathered}\exists x, x' \in L, x \neq x', \exists y = y_1 y_2 \cdots y_{d-1} \in L^{d-1}, y_i \in L, \exists u = u_1 u_2 \cdots u_l \in \Sigma^*, u_j \in \Sigma \\ \text{sao cho } xyu \in x'L^* \text{ (ta cũng có } xyu = x'y' \text{ với } y' = y'_1 y'_2 \cdots y'_n \in L^*, y'_i \in L).\end{gathered} \tag{4.2}$$

Với xâu xyu trong $\mathcal{A}_1$ tồn tại đường đi π như sau:

$$s_1 \xrightarrow{x} f_G \xrightarrow{\varepsilon} s_1 \xrightarrow{y_1} f_G \xrightarrow{\varepsilon} s_1 \cdots s_1 \xrightarrow{y_{d-1}} f_G \xrightarrow{\varepsilon} f_1 \xrightarrow{u_1} f_1 \cdots f_1 \xrightarrow{u_l} f_1.$$

Tương tự vậy, với xâu $x'y'$ trong $\mathcal{A}_2$ tồn tại đường đi θ:

$$s_2 \xrightarrow{x'} f_2 \xrightarrow{\varepsilon} s_2 \xrightarrow{y'_1} f_2 \xrightarrow{\varepsilon} s_2 \cdots s_2 \xrightarrow{y'_n} f_2.$$

Khi đó, theo tích ôtômat, trong đồ thị G có đường đi ρ được tạo lên từ π và θ, khởi đầu từ (s_1, s_2) và kết thúc là (f_1, f_2) như sau:

$$\begin{gathered}(s_1, s_2) \xrightarrow{x} (f_G, q_1) \xrightarrow{\varepsilon} (s_1, q_1) \xrightarrow{y_1} (f_G, q_2) \xrightarrow{\varepsilon} (s_1, q_2) \cdots \\ (s_1, q_{d-1}) \xrightarrow{y_{d-1}} (f_G, q_d) \xrightarrow{\varepsilon} (f_1, p_1) \xrightarrow{u_1} (f_1, p_2) \cdots (f_1, p_l) \xrightarrow{u_l} (f_1, f_2)\end{gathered}$$

do $x \neq x'$ và L là mã nên $q_i \neq f_2, s_2$, với mọi $i = 1, \ldots, d$. Vì d là lớn nhất thỏa mãn 4.2 nên đường đi ρ trên G từ đỉnh khởi đầu (s_1, s_2) đến đỉnh kết thúc (f_1, f_2) có trọng số lớn nhất là d.

($\Leftarrow$) Nếu trên G có đường đi ρ từ đỉnh khởi đầu đến đỉnh kết thúc có trọng số lớn nhất d.

+ Trường hợp $d = 0$. Khi đó, trên bất kỳ một đường đi từ đỉnh khởi đầu đến đỉnh kết thúc không có đỉnh khóa. Ta có

$$\forall x, x' \in L, \forall u \in \Sigma^*, xu \in x'L^* \Rightarrow x = x'.$$

Hay độ trễ giải mã của L là bằng 0.

+ Trường hợp $d > 0$. Khi đó đường đi ρ có dạng:

$$(s_1, s_2) \xrightarrow{x} (f_G, q_1) \xrightarrow{y_1} (f_G, q_2) \xrightarrow{y_2} (f_G, q_3) \cdots (f_G, q_{d-1}) \xrightarrow{y_{d-1}} (f_G, q_d) \xrightarrow{\varepsilon} (f_1, p_1) \xrightarrow{u} (f_1, f_2)$$

ở đó $q_i \neq f_2, s_2$, với $i = 1, \ldots, d$ và $x, y_1, \ldots, y_{d-1} \in L, u \in \Sigma^*$. Tương ứng với đường đi ρ trên G thì trên $\mathcal{A}_1$ tồn tại đường đi π với nhãn $xyu = xy_1 \cdots y_{d-1}u$:

$$s_1 \xrightarrow{x} f_G \xrightarrow{y_1} f_G \xrightarrow{y_2} f_G \cdots f_G \xrightarrow{y_{d-1}} f_G \xrightarrow{\varepsilon} f_1 \xrightarrow{u} f_1$$

và trên $\mathcal{A}_2$ cũng tồn tại đường đi θ với nhãn $x'y', x' \in L, y' \in L^*$ sao cho $xyu = x'y'$:

$$s_2 \xrightarrow{x'} f_2 \xrightarrow{y'} f_2.$$

Vậy d là lớn nhất mà $\exists x, x' \in L, x \neq x', \exists y \in L^{d-1}, \exists u \in \Sigma^*$ sao cho $xyu \in x'L^*$, hay d là nhỏ nhất sao cho: $\forall x, x' \in L, \forall y \in L^d, \forall u \in \Sigma^*, xyu \in x'L^* \Rightarrow x = x'$, suy ra L có độ trễ giải mã d. ■

Dưới đây là giải thuật xác định độ trễ giải mã của ngôn ngữ chính quy $L \subseteq \Sigma^*$.

Giải thuật DecipheringDelay($\mathcal{A}$)

Input: Ôtômat hữu hạn $\mathcal{A}$ và $L = \mathcal{L}(\mathcal{A}) \subseteq \Sigma^*$.
Output: Thông báo trong trường hợp L không là mã hoặc có độ trễ giải mã vô hạn, ngược lại giải thuật trả lại độ trễ giải mã của L.

1. if not CodeAlgo($\mathcal{A}$) then Return ("L không là mã");
2. $\mathcal{A}_2 = Ex(Trim(D(\mathcal{A}))) = (Q_2, \Sigma, E_2, s_2, f_2)$;
   $\mathcal{A}_1 = Graft(\mathcal{A}_2) = (Q_1, \Sigma, E_1, s_1, f_1)$;
3. $G = Prod(\mathcal{A}_1, \mathcal{A}_2)$; $\{cỡ\ |Q|^2\ trạng\ thái\ và\ |E|^2\ cung\}$
4. $G' = XCopy(G)$; $\{cỡ\ 2|Q|^2\ đỉnh\ và\ 3|E|^2\ cung\}$
5. if ContainsCycle($G', ((s_1, s_2), 1)$) then Return ("Độ trễ giải mã vô hạn")
6. Tìm trọng số dài nhất d của đường đi từ đỉnh (s_1, s_2) đến đỉnh (f_1, f_2) trên G
7. Return d.

Đánh giá độ phức tạp thời gian của giải thuật: Độ phức tạp thời gian của bước 1 là $\mathcal{O}((|Q| + |E|)^2)$, bước 2 đối với hàm $D(\mathcal{A})$ là $\mathcal{O}(|Q| + |E|)$, hàm Trim là $\mathcal{O}(|Q| + |E|)$, các hàm Ex và Graft là $\mathcal{O}(1)$, bước 3 là $\mathcal{O}((|Q| + |E|)^2)$, bước 4 và bước 5 là $\mathcal{O}(|Q|^2 + |E|^2)$.

Ở bước 6, G xác định một đồ thị có chu trình, các cung đi đến đỉnh khóa có trọng số là 1, các cung còn lại có trọng số là 0, do đồ thị xác định bởi G không có chu trình hợp lệ, cho nên một chu trình bất kỳ là không có đỉnh khóa, vậy các cung tham gia vào chu trình đều có

trọng số 0. Để thực hiện tìm trọng số dài nhất của đường đi từ đỉnh khởi đầu đến đỉnh kết thúc ta gán các cung có trọng số là 1 thành -1, rồi áp dụng giải thuật Bellman –Ford (xem [15]) tìm đường đi ngắn nhất trên đồ thị trọng số có thể âm và có chu trình không âm, khi đó $-d$ là giá trị nhỏ nhất tìm được thì d là trọng số dài nhất của đường đi từ đỉnh (s_1, s_2) đến đỉnh (f_1, f_2) trên G. Thực hiện toàn bộ bước 6 có độ phức tạp thời gian là $\mathcal{O}(|Q|^2|E|^2)$.

Tổng hợp lại giải thuật có độ phức tạp thời gian $\mathcal{O}(|Q|^2|E|^2) = \mathcal{O}(|E|^3) = \mathcal{O}((|E| + |Q|)^3) = \mathcal{O}(n^3)$, với $n = |Q| + |E|$, ở đây ta xem $\mathcal{O}(|E|) = \mathcal{O}(|Q|^2)$ (khi ôtômat dày cung nhất).

5. KẾT LUẬN

Nghiên cứu các mô hình ôtômat nâng cao và ứng dụng của nó là một trong các xu hướng nghiên cứu hiện đại được nhiều nhà khoa học - công nghệ quan tâm. Bài báo đề xuất giải thuật mới xác định độ trễ giải mã của ngôn ngữ chính quy, được đoán nhận bởi ôtômat hữu hạn. Đây là bài toán có ý nghĩa về lý thuyết cũng như ứng dụng thực tiễn.

TÀI LIỆU THAM KHẢO

[1] R. McCloskey, An $\mathcal{O}(n^2)$ Time algorithm for deciding whether a regular language is a code, *Journal of Computing and Information* **2** (1) (1996) 79–89.

[2] J. Berstel, D. Perrin, *Theory of Codes*, Academic Press Inc., NewYork, 1985.

[3] J. Devolder, M. Latteux, I. Litovsky, L. Staiger, Codes and infinite words, *Acta Cybernetica* **11** (4) (1994) 241-256.

[4] G. Lallement, *Semigroups and Combinational Applications*, John Wiley & Sons, Inc, 1979.

[5] L. Staiger, On infinitary finite length codes, *Informatique Theorique et Applications* **20** (4) (1986) 483–494.

[6] M. Mohri, Edit-Distance of Weighted Automata: General Definitions and Algorithms, *International Journal of Foundations of Computer Science* **14** (6) (2003) 957–982.

[7] M. Mohri, F. Pereira, M. Riley, *Speech recognition with weighted finite-state transducers*, Springer Handbook of Speech Processing, Springer, 2007.

[8] K. Stavros, Transducers and the properties of error–detection, error-correction, and finite-delay decodability. *Journal of Universal Computer Science* **8** (2) (2002) 278–291.

[9] E. N. Gilbert, E. F. Moore, Variable length binary encodings, *Bell System Technical Journal* **38** (1959) 933–967.

[10] V. I. Levenshtein, Some properties of coding and self-adjusting automata for decoding messages, *Problemy Kirbernet* **11** (1964) 63–121.

[11] M.-P. Schützenberger, On a question concerning certain free submonoids, *J. Combin. Theory* **1** (1966) 437–422.

[12] A. A. Markov, On alphabet coding *Soviet. Phys. Dokl.* **6** (1962) 553-554.

[13] C. Choffrut. Une caractérisation des codes à délai borné par leur fonction de décodage. *LITP*, 1979 (47–56).

[14] Phan Trung Huy, Vũ Thành Nam, Một số độ đo nhập nhằng của mã, *Tạp chí Tin học và Điều khiển học* **18** (3) (2002) 253–261.

[15] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms*, (3rd ed.), MIT Press and McGraw-Hill, 2009.

[16] Nguyễn Đình Hân, Đặng Quyết Thắng, Hồ Ngọc Vinh, Tính toán độ trễ giải mã của ngôn ngữ bằng otomat, *Kỷ yếu Hội thảo Quốc gia: Một số vấn đề chọn lọc của công nghệ thông tin và truyền thông*, 8/2010 (321–332).

[17] D.L. Van and I. Litovsky, On a family of code with bounded deciphering delay, *Lecture Notes of Computer Science* **2450** (2003) 369–380.

Ngày nhận bài 28 - 3 - 2012
Nhận lại sau sửa 20 - 8 - 2012