CHƯƠNG Iv

VĂN PHẠM CHÍNH QUY VÀ CÁC TÍNH CHẤT

Nội dung chính : Trong chương này, ta sẽ đề cập đến lớp văn phạm chính quy (dạng văn phạm tuyến tính trái hoặc phải) - một phương tiện khác để xác định ngôn ngữ và ta lại thấy rằng lớp ngôn ngữ do chúng sinh ra vẫn là lớp ngôn ngữ chính quy. Điều này được thể hiện bởi mối tương quan giữa văn phạm chính quy và ôtômát hữu hạn. Tiếp sau đó, ta sẽ nghiên cứu một số tính chất của lớp ngôn ngữ chính quy, cũng như các giải thuật xác định tập chính quy.

Mục tiêu cần đạt: Cuối chương, sinh viên cần phải nắm vững :

- Định nghĩa một biểu thức chính quy ký hiệu cho tập ngôn ngữ.
- Mối liên quan giữa ôtômát hữu hạn và biểu thức chính quy.
- Các tính chất của tập chính quy.
- Xây dựng ôtômát từ biểu thức chính quy
- Viết văn phạm chính quy sinh ra cùng tập ngôn ngữ được cho bởi ôtômát.

Kiến thức cơ bản: Để tiếp thu tốt nội dung của chương này, sinh viên cần nắm vững các thành phần tổng quát của một văn phạm cấu trúc, các dạng luật sinh; hiểu biết về ngôn ngữ tự nhiên; cơ chế đoán nhận ngôn ngữ từ ôtômát hữu hạn và cách phát sinh một lớp ngôn ngữ thông qua biểu thức chính quy; …

Tài liệu tham khảo :

[1] V.J. Rayward-Smith – *A First course in Formal Language Theory (Second Editor)* – McGraw-Hill Book Company Europe – 1995 (***Chapter 3 : Regular Language I)***

[2] Hồ Văn Quân – *Giáo trình lý thuyết ôtômát và ngôn ngữ hình thức* – Nhà xuất bản Đại học quốc gia Tp. Hồ Chí Minh – 2002 (***Chương 4 : Văn phạm chính quy)***

[3] From Wikipedia, the free encyclopedia - *Regular Grammar:*

http://en.wikipedia.org/wiki/Regular_grammar

CuuDuongThanCong.com https://fb.com/tailieudientucntt

I. VĂN PHẠM CHÍNH QUY (rg : REGULAR GRAMMAR)

Như trong chương 3 ta đã biết, lớp ngôn ngữ được chấp nhận bởi ôtômát hữu hạn được gọi là ngôn ngữ chính quy và chúng có thể được ký hiệu một cách đơn giản bằng việc dùng một biểu thức chính quy. Chương này giới thiệu một cách khác để mô tả ngôn ngữ chính quy thông qua cơ chế sản sinh ngôn ngữ - đó là văn phạm chính quy.

Xét một định nghĩa cho văn phạm sinh ra các số nguyên không dấu (unsigned interger) bắt đầu bằng một chữ số, theo sau bởi một chuỗi các số (digit sequence) thường dùng trong các ngôn ngữ lập trình như sau:

<digit sequence> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
| 0 <digit sequence> | 1 <digit sequence>
| 2 <digit sequence> | 3 <digit sequence>
| 4 <digit sequence> | 5 <digit sequence>
| 6 <digit sequence> | 7 <digit sequence>
| 8 <digit sequence> | 9 <digit sequence>
<unsighed integer> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
| 1 <digit sequence> | 2 <digit sequence>
| 3 <digit sequence> | 4 <digit sequence>
| 5 <digit sequence> | 6 <digit sequence>
| 7 <digit sequence> | 8 <digit sequence>
| 9 <digit sequence>

Câu hỏi :

Bạn có nhận xét gì về dạng chuỗi trong vế phải của các luật sinh văn phạm ?

Trong ví dụ trên, ta thấy mỗi vế phải hoặc là một ký hiệu kết thúc hoặc có dạng của một ký hiệu kết thúc theo sau là một biến. Trong hầu hết mọi ngôn ngữ lập trình, tất cả các ký hiệu cơ bản (số nguyên, tên biến, toán hạng, từ khóa, các ký hiệu hết câu,…) đều có thể định nghĩa bởi những quy luật ngắn dạng này. Vì phần lớn thời gian tiêu tốn trong một trình biên dịch là dùng để nhận dạng các ký hiệu cơ bản, cho nên việc khảo sát lớp văn phạm với các luật sinh dạng như trên là rất cần thiết.

1.1. Văn phạm tuyến tính

CuuDuongThanCong.com https://fb.com/tailieudientucntt

Một văn phạm G(V, T, P, S) được gọi là **tuyến tính trái** (left - linear) nếu tất cả các luật sinh của nó có dạng :

$$A \rightarrow Bw$$
$$A \rightarrow w$$

trong đó A, B là các biến $\in$ V; w là một chuỗi các ký hiệu kết thúc $\in T^*$ (có thể rỗng).

Một văn phạm G(V, T, P, S) được gọi là **tuyến tính phải** (right - linear) nếu tất cả các luật sinh của nó có dạng :

$$A \rightarrow wB$$
$$A \rightarrow w$$

Một văn phạm được gọi là văn phạm chính quy nếu nó thuộc dạng văn phạm tuyến tính trái hoặc tuyến tính phải.

Thí dụ 4.1 : Văn phạm sinh ra các số nguyên không dấu như đã nêu ở trên là văn phạm chính quy vì các luật sinh của nó có dạng tuyến tính phải.

Thí dụ 4.2 : Các văn phạm sau đây là văn phạm chính quy :

Văn phạm G_1 ({S}, {a, b}, P_1, S) với các luật sinh được cho như sau :

$$S \rightarrow abS \mid a$$

là văn phạm tuyến tính phải.

Văn phạm G_2 ({S, A, B}, {a, b}, P_2, S) với các luật sinh được cho như sau :

$$S \rightarrow Aab$$
$$A \rightarrow Aab \mid B$$
$$B \rightarrow a$$

là văn phạm tuyến tính trái.

Thí dụ 4.3 : Ngôn ngữ được ký hiệu bởi biểu thức chính quy **0(10)*** được sinh bởi văn phạm tuyến tính phải có các luật sinh sau :

$$S \rightarrow 0A \qquad (1)$$
$$A \rightarrow 10A \mid \varepsilon$$

Và bởi văn phạm tuyến tính trái :

$$S \rightarrow S10 \mid 0 \qquad (2)$$

1.2. Sự tương đương giữa văn phạm chính quy và ôtômát hữu hạn

Văn phạm chính quy mô tả tập hợp chính quy trong ngữ cảnh một ngôn ngữ là chính quy khi và chỉ khi nó được sinh ra từ văn phạm tuyến tính trái hoặc văn phạm tuyến tính phải. Kết quả này được xác định bởi hai định lý sau :

ĐỊNH LÝ 4.1 : Nếu L được sinh ra từ một văn phạm chính quy thì L là tập hợp chính quy.

CuuDuongThanCong.com https://fb.com/tailieudientucntt

Chứng minh

Trước hết, ta giả sử L = L(G) với một văn phạm tuyến tính phải G(V, T, P, S). Ta xây dựng một NFA có chứa ε-dịch chuyển M (Q, T, δ, [S], [ε]) mô phỏng các dẫn xuất trong G.

Q bao gồm các trạng thái có dạng [α] với α là S hoặc chuỗi hậu tố của vế phải một luật sinh nào đó trong P.

Ta định nghĩa δ như sau :

1) Nếu A là một biến, thì $\delta([A], \varepsilon) = \{[\alpha] \mid A \to \alpha$ là một luật sinh$\}$
2) Nếu a thuộc T và α thuộc $T^* \cup T^*V$, thì $\delta([a\alpha], a) = \{[\alpha]\}$

Sau đó, ta có thể dễ dàng chứng minh quy nạp theo độ dài của dẫn xuất rằng δ([S], w) chứa [α] khi và chỉ khi có chuỗi dẫn xuất $S \Rightarrow^* xA \Rightarrow xy\alpha$ *với* $A \to y\alpha$ *là một luật sinh trong P và xy = w, hay nếu* $\alpha = S$ *thì* $w = \varepsilon$*. Khi [ε] là trạng thái kết thúc duy nhất, M chấp nhận w khi và chỉ khi* $S \Rightarrow^* xA \Rightarrow w$*. Nhưng vì mọi chuỗi dẫn xuất cho một chuỗi ký hiệu kết thúc qua ít nhất 1 bước, nên ta thấy rằng M chấp nhận w khi và chỉ khi G sinh ra w. Vì vậy, mọi văn phạm tuyến tính phải đều sinh ra một tập hợp chính quy.*

Bây giờ, giả sử G(V, T, P, S) là một văn phạm tuyến tính trái. Đặt văn phạm G'(V, T, P', S) với P' chứa các luật sinh của P có vế phải đảo ngược, nghĩa là :

$P' = \{ A \to \alpha \mid A \to \alpha^R \in P \}$

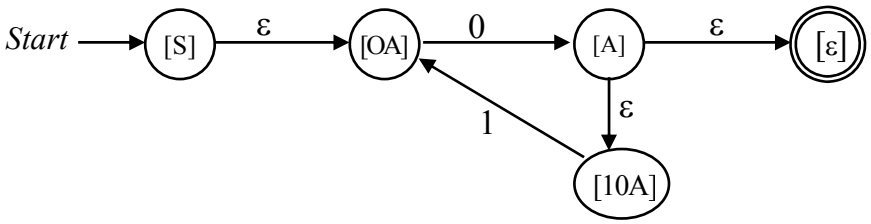
Nếu ta đảo ngược chuỗi vế phải các luật sinh trong một văn phạm tuyến tính trái, ta có văn phạm tuyến tính phải, và ngược lại. Do đó, hiển nhiên chúng ta có G' là một văn phạm tuyến tính phải, và cũng dễ dàng để chỉ ra rằng $L(G') = L(G)^R$. Theo chứng minh trên, ta có L(G') là một tập chính quy. Mà thông thường một tập chính quy cũng vẫn còn giữ nguyên tính chất khi áp dụng phép đảo ngược nên $L(G')^R = L(G)$ cũng là một tập chính quy.

Vậy, mọi văn phạm tuyến tính trái hay phải đều sinh ra một tập hợp chính quy.

Thí dụ 4.3 : NFA được xây dựng từ Định lý 4.1 cho văn phạm tuyến tính phải (1) ở thí dụ 4.2 có dạng như hình 4.1 sau :

Xét văn phạm tuyến tính trái (2) ở thí dụ 4.2, nếu đảo ngược các vế phải luật sinh, ta có:

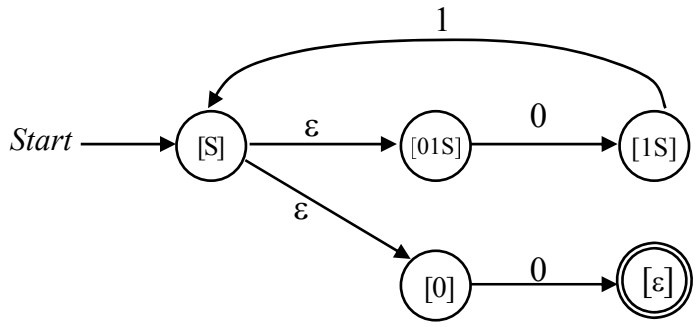
$S \to 01S \mid 0$



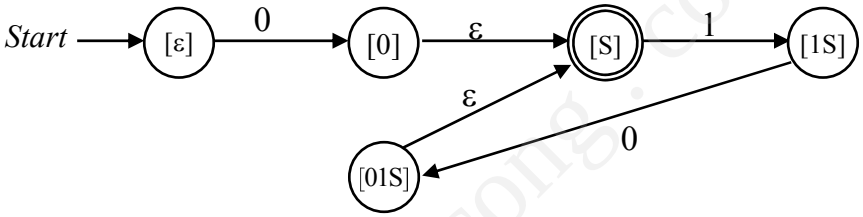
Hình 4.1 - NFA chấp nhận ngôn ngữ $0(10)^*$

CuuDuongThanCong.com https://fb.com/tailieudientucntt

Áp dụng các bước xây dựng NFA cho văn phạm này theo Định lý 4.1, ta có sơ đồ chuyển như Hình 4.2 (a). Nếu chúng ta đảo ngược các cạnh của NFA này và chuyển đổi vị trí các trạng thái bắt đầu và kết thúc, chúng ta sẽ có một NFA khác chấp nhận ngôn ngữ **0(10)***



Hình (a)



Hình (b)

Hình 4.2 - Xây dựng NFA cho 0(10)* từ văn phạm tuyến tính trái

<u>ĐỊNH LÝ 4.2</u> : Nếu L là một tập hợp chính quy, thì L được sinh từ một văn phạm tuyến tính trái hoặc một văn phạm tuyến tính phải nào đó.

Chứng minh

Đặt L = L(M) với DFA M(Q, Σ, δ, q_0, F).

Trước hết, ta giả sử rằng trạng thái q_0 không phải là trạng thái kết thúc. Kế tiếp, ta đặt L = L(G) với văn phạm tuyến tính phải G(V, Σ, P, q_0), trong đó P chứa các luật sinh dạng $p \rightarrow aq$ nếu $\delta(p, a) = q$ và luật sinh dạng $p \rightarrow a$ nếu $\delta(p, a)$ là một trạng thái kết thúc. Rõ ràng $\delta(p, w) = q$ khi và chỉ khi có chuỗi dẫn xuất $p \Rightarrow^* wq$. Nếu wa được chấp nhận bởi M, ta đặt $\delta(q_0, w) = p$, suy ra dẫn xuất $q_0 \Rightarrow^* wq$. Tương tự, nếu $\delta(p, a)$ là trạng thái kết thúc, vì $p \rightarrow a$ là một luật sinh, nên $q_0 \Rightarrow^* wa$. Ngược lại, đặt $q_0 \Rightarrow^* x$. Ta có x = wa và $q_0 \Rightarrow^* wq \Rightarrow wa$ với mọi p. Và vì $\delta(q_0, w) = p$ và $\delta(p, a)$ là trạng thái kết thúc nên do đó $x \in L(M)$. Hay nói cách khác : L(M) = L(G) = L.

Bây giờ, xét $q_0 \in F$, vì thế chuỗi rỗng ε thuộc L. Lưu ý rằng văn phạm G vừa định nghĩa ở trên chỉ sinh ra ngôn ngữ L – {ε}. Chúng ta có thể sửa đổi G bằng cách thêm vào một ký hiệu bắt đầu S mới với luật sinh $S \rightarrow q_0 \mid \varepsilon$. Văn phạm thu được vẫn có dạng tuyến tính phải và phát sinh ngôn ngữ L.

CuuDuongThanCong.com https://fb.com/tailieudientucntt

Để phát sinh một văn phạm tuyến tính trái cho L, ta bắt đầu với một NFA cho L^R và sau đó đảo ngược chuỗi vế phải cho tất cả mọi luật sinh của văn phạm tuyến tính phải vừa thu được.

Thí dụ 4.4 : Trong Hình 4.3 ta thấy sơ đồ chuyển DFA cho **0(10)***

Văn phạm tuyến tính phải sinh từ DFA này có dạng :

$A \rightarrow 0B \mid 1D \mid 0$

$B \rightarrow 0D \mid 1C$

$C \rightarrow 0B \mid 1D \mid 0$

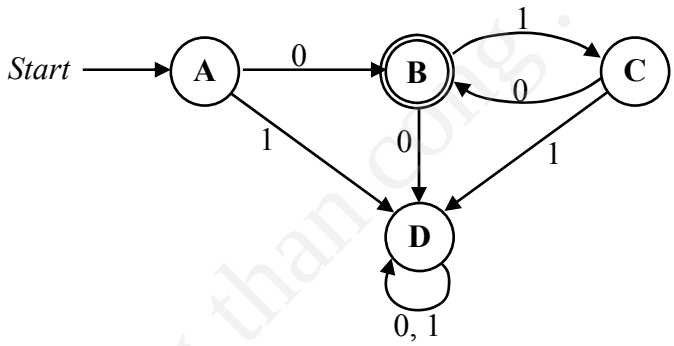
$D \rightarrow 0D \mid 1D$

Trong văn phạm trên, biến D không có ích nên ta có thể loại bỏ D và tất cả các luật sinh liên quan tới D, rút gọn văn phạm thành :

$A \rightarrow 0B \mid 0$

$B \rightarrow 1C$

$C \rightarrow 0B \mid 0$



Hình 4.3 - DFA cho 0(10)*

Ii. MỘT SỐ TÍNH CHẤT CỦA TẬP HỢP CHÍNH QUY

Một câu hỏi khá quan trọng được đặt ra là: Cho ngôn ngữ L với một số tính chất đặc tả nào đó, liệu L có phải là tập chính quy không ? Phần này cung cấp một số lý thuyết giúp trả lời câu hỏi này.

2.1. Bổ đề bơm cho tập hợp chính quy

Một trong những nguyên lý hiệu quả là "Bổ đề bơm", đây là một công cụ mạnh giúp chứng minh các ngôn ngữ không là chính quy. Đồng thời, nó cũng thực sự hữu ích trong việc phát triển các giải thuật liên quan đến các ôtômát, chẳng hạn như một ngôn ngữ được chấp nhận bởi một FA cho trước là hữu hạn hay vô hạn ?

BỔ ĐỀ 4.1: (BỔ ĐỀ BƠM)

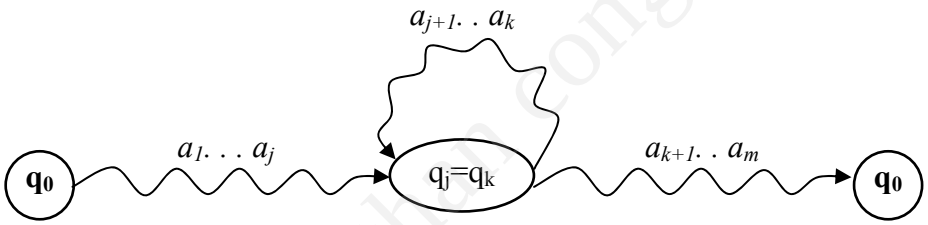
CuuDuongThanCong.com https://fb.com/tailieudientucntt

Nếu L là tập hợp chính quy thì có tồn tại hằng số n sao cho nếu z là một từ bất kỳ thuộc L và | z | ≤ n, ta có thể viết z = uvw với | uv | ≤ n, | v | ≥ 1 và ∀i ≥ 0, ta có $uv^iw \in L$.
Hơn nữa n không lớn hơn số trạng thái của FA nhỏ nhất chấp nhận L.

Chứng minh

Nếu một ngôn ngữ L là ngôn ngữ chính quy thì nó sẽ được chấp nhận bởi một DFA M (Q, Σ, δ, q_0, F) với n trạng thái.

Xét chuỗi nhập z có m ký hiệu được cho như trong bổ đề, vậy z = $a_1a_2 \ldots a_m$, m ≥ n, và với mỗi i = 1, 2, ..., m , ta đặt $\delta(q_0, a_1a_2...a_i) = q_i$. Do m ≥ n nên cần phải có ít nhất n+1 trạng thái trên đường đi của ôtômát chấp nhận chuỗi z. Trong n+1 trạng thái này phải có hai trạng thái trùng nhau vì ôtômát M chỉ có n trạng thái phân biệt, tức là có hai số nguyên j và k sao cho 0 ≤ j < k ≤ n thỏa mãn $q_j = q_k$. Đường đi nhãn $a_1a_2 \ldots a_m$ trong sơ đồ chuyển của M có dạng như sau:



Hình 4.4 - Đường đi trong sơ đồ chuyển của DFA M

Vì j < k nên chuỗi $a_{j+1}...a_k$ có độ dài ít nhất bằng 1 và vì k ≤ n nên độ dài đó không thể lớn hơn n.
Nếu q_m là một trạng thái trong F, nghĩa là chuỗi $a_1a_2...a_m$ thuộc L(M), thì chuỗi $a_1a_2...a_j\ a_{k+1}a_{k+2}...a_m$ cũng thuộc L(M) vì có một đường dẫn từ q_0 đến q_m ngang qua q_j nhưng không qua vòng lặp nhãn $a_{j+1}... a_k$. Một cách hình thức, ta có :

$$\begin{aligned}\delta(q_0, a_1a_2...a_j\ a_{k+1}a_{k+2}...a_m) &= \delta(\delta(q_0, a_1a_2...a_j), a_{k+1}a_{k+2}...a_m)\\ &= \delta(q_j, a_{k+1}a_{k+2}...a_m)\\ &= \delta(q_k, a_{k+1}a_{k+2}...a_m)\\ &= q_m\end{aligned}$$

Vòng lặp trong hình trên có thể được lặp lại nhiều lần - thực tế, số lần muốn lặp là tùy ý, do đó chuỗi $a_1...a_j\ (a_{j+1}...a_k)^i\ a_{k+1}...a_m \in L(M)$, ∀i ≥ 0. Điều ta muốn chứng tỏ ở đây là với một chuỗi có độ dài bất kỳ được chấp nhận bởi một FA, ta có thể tìm được một chuỗi con gần với chuỗi ban đầu mà có thể "bơm" - lặp một số lần tùy ý - sao cho chuỗi mới thu được cũng được chấp nhận bởi FA.
Đặt u = $a_1...a_j$, v = $a_{j+1}...a_k$ và w = $a_{k+1}...a_m$.
Ta có điều phải chứng minh.

Ứng dụng của bổ đề bơm

CuuDuongThanCong.com https://fb.com/tailieudientucntt

Bổ đề bơm rất có hiệu quả trong việc chứng tỏ một tập hợp không là tập hợp chính quy. Phương pháp chung để ứng dụng nó dùng phương pháp chứng minh "phản chứng" theo dạng sau :

1) Chọn ngôn ngữ mà bạn cần chứng tỏ đó không là ngôn ngữ chính quy.
2) Chọn hằng số n, hằng số được đề cập đến trong bổ đề bơm.
3) Chọn chuỗi z thuộc L. Chuỗi z phải phụ thuộc nghiêm ngặt vào hằng số n đã chọn ở bước 2.
4) Giả thiết phân chuỗi z thành các chuỗi con u, v, w theo ràng buộc $| uv | \leq n$ và $| v | \geq 1$
5) Mâu thuẫn sẽ phát sinh theo bổ đề bơm bằng cách chỉ ra với u, v và w xác định theo giả thiết, có tồn tại một số i mà ở đó $uv^iw \notin L$. Từ đó có thể kết luận rằng L không là ngôn ngữ chính quy. Chọn lựa giá trị cho i có thể phụ thuộc vào n, u, v và w.

Ta có thể phát biểu một cách hình thức nội dung của bổ đề bơm như sau :

$(\forall L)(\exists n)(\forall z)[$ z thuộc L và $| z | \geq n$ ta có

$(\exists u, v, w)(z = uvw, | uv | \leq n, | v | \geq 1$ và $(\forall i)(uv^iw$ thuộc L))]

Thí dụ 4.5 : Chứng minh tập hợp $L = \{ 0^{i2} \mid i$ là số nguyên, $i \geq 1\}$ (L chứa tất cả các chuỗi số 0 có độ dài là một số chính phương) là tập không chính quy.

Chứng minh

Giả sử L là tập chính quy và tồn tại một số n như trong bổ đề bơm.

Xét từ $z = 0^{n2}$. Theo bổ đề bơm, từ z có thể viết là z = uvw với $1 \leq | v | \leq n$ và $uv^iw \in L, \forall i \geq 0$. Trường hợp cụ thể, xét i = 2 : ta phải có $uv^2w \in L$.

Mặt khác : $n^2 < | uv^2w | \leq n^2 + n < (n+1)^2$.

Do n^2 và $(n+1)^2$ là 2 số chính phương liên tiếp nên $| uv^2w |$ không thể bằng một số chính phương, vậy $uv^2w \notin L$.

Điều này dẫn đến sự mâu thuẫn, vậy giả thiết ban đầu là sai. Suy ra L không là tập chính quy.

Câu hỏi :

Hãy tự liên hệ một số tập ngôn ngữ khác mà bạn nghĩ chúng không thuộc lớp ngôn ngữ chính quy vì không thể thỏa mãn các tính chất của Bổ đề bơm ?

2.2. Tính chất đóng của tập hợp chính quy

Có nhiều phép toán trên ngôn ngữ chuyên sử dụng cho tập hợp chính quy, mà cho phép khi áp dụng chúng vào tập hợp chính quy thì vẫn giữ được các tính chất của tập

CuuDuongThanCong.com https://fb.com/tailieudientucntt

chính quy. Nếu một lớp ngôn ngữ nào đó "đóng" với một phép toán cụ thể, ta gọi đó là *tính chất đóng* của lớp ngôn ngữ này.

ĐỊNH LÝ 4.3 : Tập hợp chính quy đóng với các phép toán: hợp, nối kết và bao đóng Kleen.
Chứng minh

Hiển nhiên từ định nghĩa của biểu thức chính quy.

ĐỊNH LÝ 4.4 : Tập hợp chính quy đóng với phép lấy phần bù. Tức là, nếu L là tập chính quy và $L \subseteq \Sigma^*$ thì $\Sigma^* - L$ là tập chính quy.

Chứng minh

Gọi L là L(M) cho DFA M ($Q, \Sigma_1, \delta, q_0, F$) và $L \subseteq \Sigma^*$.

Trước hết, ta giả sử $\Sigma_1 = \Sigma$ vì nếu có ký hiệu thuộc Σ_1 mà không thuộc Σ thì ta có thể bỏ các phép chuyển trong M liên quan tới các ký hiệu đó. Do $L \subseteq \Sigma^*$ nên việc xóa như vậy không ảnh hưởng tới M. Nếu có ký hiệu thuộc Σ nhưng không thuộc Σ_1 thì các ký hiệu này không xuất hiện trong L. Ta thiết kế thêm một trạng thái "chết" d trong M sao cho $\delta(d, a) = d, \forall a \in \Sigma$ và $\delta(q, a) = d, \forall q \in Q$ và $a \in \Sigma - \Sigma_1$.

Bây giờ, để chấp nhận $\Sigma^* - L$, ta hoàn thiện các trạng thái kết thúc của M. Nghĩa là, đặt M' = ($Q, \Sigma, \delta, q_0, Q - F$). Ta có M' chấp nhận từ w nếu $\delta(q_0,w) \in Q - F$, suy ra $w \in \Sigma^* - L$.

ĐỊNH LÝ 4.5: Tập hợp chính quy đóng với phép giao

Chứng minh

Do ta có công thức biến đổi :

$$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$$

Nên theo các định lý trên, suy ra được tập $L_1 \cap L_2$ là tập chính quy.

Iii. các GIẢI THUẬT xác đỊnh TẬP hỢp CHÍNH QUY

Một vấn đề khác, cũng rất cần thiết là xác định các giải thuật giúp giải đáp nhiều câu hỏi liên quan đến tập hợp chính quy, chẳng hạn như : Một ngôn ngữ cho trước là rỗng, hữu hạn hay vô hạn ? Ngôn ngữ chính quy có tương đương với ngôn ngữ nào khác không ? ... Để xác định các giải thuật này, trước hết cần giả sử mỗi tập chính quy thì được biểu diễn bởi một ôtômát hữu hạn. Như đã biết, biểu thức chính quy dùng đặc tả cho tập hợp chính quy, do đó chỉ cần cung cấp thêm một cơ chế dịch từ dạng biểu thức này sang dạng ôtômát hữu hạn. Một số định lý sau có thể xem là nền tảng cho việc chuyển đổi này.

CuuDuongThanCong.com https://fb.com/tailieudientucntt

ĐỊNH LÝ 4.6: Tập hợp các chuỗi được chấp nhận bởi ôtômát M có n trạng thái là:
1) Không rỗng nếu và chỉ nếu ôtômát chấp nhận một chuỗi có độ dài < n.
2) Vô hạn nếu và chỉ nếu ôtômát chấp nhận một chuỗi có độ dài l với n ≤ l < 2n.

Chứng minh

1) Phần "nếu " là hiển nhiên.

Ta chứng minh "chỉ nếu": Giả sử M chấp nhận một tập không rỗng. Gọi w là chuỗi ngắn nhất được chấp nhận bởi M. Theo bổ đề bơm, ta có $|w| < n$ vì nếu w là chuỗi ngắn nhất và $|w| \geq n$ thì ta có thể viết w = uvy, và uy là chuỗi ngắn hơn trong L hay $|uy| < |w| \Rightarrow$ Mâu thuẫn.

2) Nếu $w \in L$ và $n \leq |w| < 2n$ thì theo bổ đề bơm ta có $w = w_1w_2w_3$ và $w_1w_2^{\,i}w_3 \in L$ với mọi $i \geq 0$, suy ra L(M) vô hạn .

Ngược lại, nếu L(M) vô hạn thì tồn tại $w \in L(M)$ sao cho $|w| \geq n$. Nếu $|w| < 2n$ thì xem như đã chứng minh xong. Nếu không có chuỗi nào có độ dài nằm giữa n và 2n-1 thì gọi w là chuỗi có độ dài ít nhất là 2n nhưng ngắn hơn mọi chuỗi trong L(M), nghĩa là $|w| \geq 2n$. Một lần nữa, cũng theo bổ đề bơm, ta có thể biểu diễn $w = w_1w_2w_3$, trong đó $1 \leq |w_2| \leq n$ và $w_1w_3 \in L(M)$. Ta có hoặc w không phải là chuỗi ngắn nhất có độ dài ≥ 2n, hoặc là $n \leq |w_1w_3| \leq 2n-1 \Rightarrow$ Mâu thuẫn. Vậy có tồn tại chuỗi có độ dài l sao cho $n \leq l < 2n$.

ĐỊNH LÝ 4.7 : Có giải thuật để xác định hai ôtômát tương đương (chấp nhận cùng một ngôn ngữ).

Chứng minh

Đặt M_1, M_2 là hai ôtômát chấp nhận L_1, L_2.

Theo các định lý 4.3, 4.4, 4.5, ta có $(L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2)$ được chấp nhận bởi ôtômát M_3 nào đó. Dễ thấy M_3 chấp nhận một chuỗi nếu và chỉ nếu $L_1 \neq L_2$. Theo định lý 4.6, ta thấy có giải thuật để xác định xem liệu $L_1 = L_2$ hay không.

CuuDuongThanCong.com https://fb.com/tailieudientucntt

Tổng kết chương IV: Qua chương này, chúng ta có thể thấy rõ hơn các tính chất của lớp ngôn ngữ chính quy và cách xác định chúng bằng một số giải thuật. Mối liên quan giữ hai cơ chế đoán nhận ngôn ngữ (ôtômát hữu hạn) và phát sinh ngôn ngữ (văn phạm) cũng đã được thiết lập và chứng minh rõ ràng. Đây là lớp ngôn ngữ nhỏ nhất theo sự phân cấp của Noam Chomsky. Trong những chương tiếp theo, chúng ta sẽ khảo sát những lớp ngôn ngữ rộng lớn hơn chứa cả ngôn ngữ chính quy trong nó.

BÀI TẬP CHƯƠNG IV

4.1. Xây dựng văn phạm tuyến tính trái và tuyến tính phải cho các ngôn ngữ sau :

a) $(0 + 1)^*00(0 + 1)^*$

b) $0^*(1(0 + 1))^*$

c) $(((01 + 10)^*11)^*00)^*$

4.2. Xây dựng văn phạm chính quy sinh ra các ngôn ngữ trên bộ chữ cái $\Sigma = \{0,1\}$ như sau :

a) Tập các chuỗi có chứa 3 con số 0 liên tiếp.

b) Tập các chuỗi kết thúc bằng 2 con số 0.

4.3. Xây dựng văn phạm chính quy sinh ra các ngôn ngữ sau :

a) $\{ w \mid w \in (0 + 1)^* \}$

b) $\{ a^m b^n \mid m, n > 0 \}$

4.4. Chứng tỏ rằng ngôn ngữ $L = \{0^n1^n \mid$ n là số nguyên dương$\}$ không chính qui.

4.5. Ngôn ngữ nào trong các ngôn ngữ sau không là ngôn ngữ chính qui? Chứng minh câu trả lời:

a) $L = \{0^{2n} \mid$ n là số nguyên dương $\}$

b) $L = \{0^n1^m0^{n+m} \mid$ m, n là số nguyên dương$\}$

c) $L = \{0^n \mid$ n là số nguyên tố $\}$

CuuDuongThanCong.com https://fb.com/tailieudientucntt