

Ôtômat và ngôn ngữ hình thức

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP VINH
KHOA CÔNG NGHỆ
NGÀNH CÔNG NGHỆ THÔNG TIN

NGUYỄN NGỌC THUẦN

ÔTÔMÁT VÀ NGÔN NGỮ HÌNH THỨC

Vinh, 2016

LỜI NÓI ĐẦU

Ôtômát và ngôn ngữ hình thức là lĩnh vực thuộc chuyên ngành Khoa học máy tính, chủ yếu dùng trong việc mô tả, hình thức hóa các dãy tính toán và điều khiển tự động, được phát sinh trong nhiều ngành khoa học khác nhau từ các hệ thống tính toán, ngôn ngữ học đến sinh học,

Ngày nay, các ngôn ngữ lập trình, các hệ thống máy tính, các quá trình xử lý thông tin và các quá trình tính toán nói chung đều được hình thức hoá thành các mô hình toán học mà lý thuyết ôtômát và ngôn ngữ hình thức là công cụ. Ôtômát và ngôn ngữ hình thức được áp dụng rộng rãi trong nhiều lĩnh vực khoa học và ứng dụng như: mô hình hoá, mô phỏng các hệ thống tính toán, các kỹ thuật dịch, thông dịch, trí tuệ nhân tạo, công nghệ tri thức, ...

Nhằm đáp ứng nhu cầu giảng dạy, học tập và nghiên cứu cho các sinh viên ngành Công nghệ thông tin của trường Đại học Công nghiệp Vinh – là trường Đại học hoạt động theo hướng thực hành, chúng tôi biên soạn giáo trình “**Ôtômát và ngôn ngữ hình thức**” theo hướng kết hợp ba mảng chính: lý thuyết ôtômát, ngôn ngữ hình thức và lý thuyết tính toán gồm những khái niệm, kiến thức cơ bản nhất với nhiều ví dụ minh họa, bỏ qua các chứng minh lý thuyết không thực sự cần thiết, nhằm tập trung tối đa thời gian cho việc giải các bài tập, ví dụ thực tiễn.

Giáo trình giới thiệu một cách hệ thống những khái niệm cơ bản và các tính chất chung của ôtômát và ngôn ngữ hình thức.

Chương mở đầu trình bày các khái niệm cơ bản, các tính chất quan trọng của các cấu trúc đại số, logic mệnh đề,... làm cơ sở cho các chương sau. Chương 2, giới thiệu về Văn phạm và các ngôn ngữ hình thức. Lý thuyết Ôtômát, những khái niệm cơ sở và các hoạt động của Ôtômát được đề cập ở Chương 3. Những vấn đề liên quan đến tập chính qui, ngôn ngữ chính qui và ôtômát hữu hạn được trình bày chi tiết ở Chương 4. Chương 5, nghiên cứu các khái niệm cơ sở, mối quan hệ giữa các lớp ngôn ngữ và các tính chất rất quan trọng của ngôn ngữ phi ngữ cảnh, Ôtômát đẩy xuống.

Sau mỗi Chương có các bài tập hệ thống hoá lại kiến thức và thông qua đó, học viên nắm bắt được các khái niệm cơ bản, các kiến thức chủ yếu của từng Chương và cuối cùng là để nâng cao sự hiểu biết về ôtômát và ngôn ngữ hình thức, đồng thời phát triển khả năng ứng dụng chúng trong nghiên cứu và triển khai ứng dụng Công nghệ thông tin.

Ôtômat và ngôn ngữ hình thức

Mặc dù đã có nhiều cố gắng, nhưng tài liệu này chắc không tránh khỏi những thiếu sót. Chúng tôi rất mong nhận được các ý kiến, góp ý của các đồng nghiệp và bạn đọc để hoàn thiện hơn cuốn giáo trình Ôtômat và ngôn ngữ hình thức. Các ý kiến đóng góp xin được gửi về: Bộ môn CNTT, Khoa Công Nghệ, trường ĐHCN Vinh. Số 26, Đường Nguyễn Thái Học, Phường Đội Cung, Tp. Vinh, Tỉnh Nghệ An.

Chủ biên

TS. Nguyễn Ngọc Thuận

MỤC LỤC

Nội dung	Trang
Chương 1: Bồ túc – Các khái niệm cơ bản.	1 – 10
1.1 Tập hợp	1
1.2 Quan hệ và Đồ thị.	3
1.3. Đại cương về ngôn ngữ.	7
1.4. Bài tập:	10
Chương 2: Văn phạm và các ngôn ngữ hình thức	11 – 37
2.1 Văn phạm và các ngôn ngữ	11
Định nghĩa Văn phạm: Bộ 4 thành phần $G = (\Sigma, V, P, S)$.	12
2.2. Các dẫn xuất và ngôn ngữ sinh bởi văn phạm	13
+ Dẫn xuất	14
+ Ngôn ngữ sinh.	15
+ Văn phạm tương đương	15
+ Bài toán thuận và Bài toán ngược.	16 - 21
2.3. Phân loại ngôn ngữ (Văn phạm) của Chomsky	22 - 27
2.4. Tính đệ qui và tập đệ qui.	28 - 30
2.5. Các phép toán trên các ngôn ngữ.	31 – 35
2.6. Các ví dụ và bài tập.	36 – 37
Chương 3: Lý thuyết Ôtômát	38 - 57
3.1 Ôtômát hữu hạn	38
3.1.1 Hàm chuyển trạng thái và tính chất	40
3.1.2 Các phương pháp biểu diễn ôtômát	41
3.1.3 Ngôn ngữ đoán nhận được của ôtômát	42
3.2. Ôtômát hữu hạn không đơn định	44
3.3. Sự tương đương của ôtômát đơn định và không đơn định.	47 - 50
3.4. Cực tiểu hoá ôtômát hữu hạn.	51 – 55
3.5. Các ví dụ và Bài tập.	56 – 57

Ôtômát và ngôn ngữ hình thức

Chương 4: Tập chính qui và văn phạm chính qui	58 - 75
4.1 Các biểu thức chính qui.	58
4.2 Sự tương đương của các biểu thức chính qui	59
Định lý (Định lý Arden)	60
4.3 Ôtômát hữu hạn và biểu thức chính qui	62
4.3.1 Phương pháp đại số ứng dụng định lý Arden	63 - 66
4.3.2 Thiết lập ôtômát hữu hạn tương đương với biểu thức chính qui	67
4.4 Bỏ đề Bơm đối với các tập chính qui.	67
4.5 Ứng dụng Bỏ đề Bơm	68
4.6 Các tập chính qui và văn phạm chính qui	69
4.6.1 Xây dựng VP chính qui tương đương với ÔTĐĐ cho trước	69 - 70
4.6.2 Xây dựng ÔTĐĐ hữu hạn tương đương với VP chính qui G	70 - 71
4.5. Các ví dụ và Bài tập.	71 - 75
Chương 5: Ngôn ngữ phi ngữ cảnh và Ôtômát đẩy xuống	76 - 95
5.1 Ngôn ngữ phi ngữ cảnh và cây dẫn xuất	76 - 81
5.2 Sự nhập nhằng trong văn phạm phi ngữ cảnh	82
5.3 Dạng chuẩn Chomsky	82
5.4 Bỏ đề Bơm cho ngôn ngữ phi ngữ cảnh	83 - 85
5.5 Ôtômát đẩy xuống (PDA) và ứng dụng.	86
5.5.1. Cấu tạo và hoạt động của PDA	87
5.5.2. Đoán nhận xâu vào của PDA	88
5.5.3. Ngôn ngữ đoán nhận của PDA	89 - 90
5.5.4. Mối quan hệ giữa PDA và Văn phạm phi ngữ cảnh	91 - 93
5.6. Các ví dụ và Bài tập.	94 - 95

CHƯƠNG 1. BỔ TÚC – CÁC KHÁI NIỆM CƠ BẢN

Chương mở đầu giới thiệu khái quát và tóm lược lại các khái niệm cơ bản, các tính chất và các ký hiệu được sử dụng trong tất cả các chương sau:

- ✓ Tập hợp, quan hệ và đồ thị.
- ✓ Các phép toán trên tập hợp.
- ✓ Đại cương về ngôn ngữ và các tính chất.

1.1 Tập hợp

Tập hợp và các tập con

Tập hợp là sự kết tập những đối tượng có cùng một số thuộc tính giống nhau.

Ví dụ: Tập tất cả các sinh viên của Khoa CNTT. Mỗi *sinh viên* được gọi là một *phần tử* của tập hợp.

Ký hiệu tập hợp: Các chữ in hoa $A, B, C, \dots$

Ký hiệu phần tử của tập: Các chữ thường $a, b, c, \dots$

Phần tử a thuộc tập A , ký hiệu là $a \in A$. Ngược lại, khi a không phải là phần tử của A , ($a \notin A$).

Các mô tả của Tập hợp:

- (i) *Đếm các phần tử.* Ta có thể liệt kê các phần tử của tập hợp theo thứ tự bất kỳ đặt trong cặp dấu ngoặc $\{, \}$. Ví dụ: Tập các số tự nhiên chia hết cho 7 và nhỏ hơn 50 có thể viết là $\{7, 14, 21, 28, 35, 42, 49\}$.
- (ii) *Mô tả tập hợp theo tính chất của các phần tử.* $T = \{x \mid P(x)\}$ - Đây là tập tất cả x thỏa mãn tân từ $P(x)$.

Ví dụ: Tập các số tự nhiên chia hết cho 7 và nhỏ hơn 50 có thể viết:
 $\{n \mid n \text{ là số nguyên dương chia hết cho 7 và nhỏ hơn 50}\}$.

Ở đây, $P(x)$ là tân từ: $P(x) = n \text{ là số nguyên dương chia hết cho 7 và nhỏ hơn 50}$.

- (iii) *Định nghĩa đệ quy.* Các phần tử của tập hợp được xác định thông qua các qui tắc tính toán từ các phần tử đã biết.

Ví dụ: Tập các số giai thừa của n được định nghĩa:

$$n! = \{f_n \mid f_0 = 1; f_n = n * f_{n-1}, n = 1, 2, \dots\}.$$

Tập con và các phép toán trên tập hợp

Tập A được gọi là *tập con* của B (viết $A \subseteq B$), nếu mọi phần tử của A đều là phần tử của B .

Hai tập A và B là *bằng nhau* (viết $A = B$), nếu chúng có cùng tập các phần tử. Thông thường, để chứng minh $A = B$, chúng ta cần chứng minh $A \subseteq B$ và $B \subseteq A$.

Một tập đặc biệt không chứa phần tử nào được gọi là *tập trống (rỗng)*, ký hiệu là $\emptyset$.

Trên tập các tập hợp xác định một số phép toán: phép hợp $\cup$, phép giao $\cap$, phép trừ - và tích *Đề Các* được định nghĩa như sau:

$A \cup B = \{x \mid x \in A \text{ hoặc } x \in B\}$, gọi là *hợp* của hai tập hợp,

$A \cap B = \{x \mid x \in A \text{ và } x \in B\}$, gọi là *giao* của hai tập hợp,

$A - B = \{x \mid x \in A \text{ và } x \notin B\}$, gọi là *hiệu* của hai tập hợp.

$C = U - A$, với U tập vũ trụ tất cả các phần tử đang xét, được gọi là *phần bù của A*.

Tập tất cả các tập con của A được ký hiệu là $2^A = \{B \mid B \subseteq A\}$, hoặc $\wp(A)$.

$A \times B = \{(a, b) \mid a \in A \text{ và } b \in B\}$, gọi là *tích Đề Các của A và B* hay còn được gọi là *tích tự nhiên* của hai tập hợp.

Ví dụ: Cho $A = \{1, 2\}$, $B = \{2, 3\}$. Ta có:

$$A \cup B = \{1, 2, 3\}; A \cap B = \{2\}; A \times B = \{(1, 2), (1, 3), (2, 2), (2, 3)\};$$

$$A - B = \{1\}; 2^A = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\};$$

Lực lượng của tập hợp:

+) Nếu các tập A, B là hữu hạn ($|A| = m, |B| = n$), thì $|A \times B| = m \times n$; $|2^A| = 2^{|A|} = 2^m$

+) S_1, S_2 có cùng bản số (lực lượng) $\Leftrightarrow \exists f$ (1-1, lên) $f: S_1 \rightarrow S_2$

+) Nếu S_1, S_2 hữu hạn và S_1 là tập con thực sự của S_2 thì $|S_1| < |S_2|$.

+) Nếu S_1, S_2 là các tập vô hạn, thì khẳng định trên không đúng. ($S_2 = \mathbb{Z}$, $S_1 = \mathbb{N}$ nguyên chẵn; $f(x) = 2x$ là ánh xạ 1-1 từ $\mathbb{Z}$ lên S_1).

Định nghĩa: Giả sử S là một tập hợp bất kỳ. Một họ các tập con $\{A_1, A_2, \dots, A_n\}$ của S được gọi là một *phân hoạch* của S nếu:

(i) $A_i \cap A_j = \emptyset, i \neq j$, các tập con khác nhau là rời nhau,

(ii) $S = A_1 \cup A_2 \cup \dots \cup A_n$, hợp của các tập con đó chính bằng S

Ví dụ: $S = \{1, 2, 3, \dots, 10\}$ có thể phân hoạch thành hai tập $A_1 = \{1, 3, 5, 7, 9\}$ - tập các số lẻ và tập các số chẵn $A_2 = \{2, 4, 6, 8, 10\}$.

1.2 Quan hệ và Đồ thị.

Quan hệ là khái niệm cơ sở quan trọng trong khoa học máy tính cũng như trong cuộc sống. Khái niệm quan hệ xuất hiện khi xem xét mối quan hệ giữa các cặp đối tượng và so sánh một đối tượng này với đối tượng khác, ví dụ, quan hệ “anh em” giữa hai người. Chúng ta có thể biểu diễn mối quan hệ giữa a và b bởi cặp được sắp xếp (a, b) thể hiện a là anh của b , chẳng hạn. Trong khoa học máy tính, khái niệm quan hệ xuất hiện khi cần nghiên cứu các cấu trúc của dữ liệu.

Định nghĩa (hai ngôi): Quan hệ hai ngôi R trên tập S là tập con các cặp phần tử của S , $R \subseteq S \times S$. Khi x và y có quan hệ R với nhau ta viết $x R y$ hoặc $(x, y) \in R$.

Ví dụ: Trên tập Z , định nghĩa quan hệ $R: x R y$ nếu $x = y + 5$.

Định nghĩa các tính chất: Quan hệ R trên tập S là:

(i) *Phản xạ.* Nếu xRx với mọi $x \in S$.

(ii) *Đối xứng.* Với mọi $x, y \in S$, nếu xRy thì yRx

(iii) *Bắc cầu.* Với mọi $x, y, z \in S$, nếu xRy và yRz thì xRz .

(iv) (Phản đối xứng) Với mọi $x, y \in S$, nếu xRy và yRx thì $x = y$.

Định nghĩa: +). Quan hệ R trên tập S được gọi là *quan hệ tương đương* nếu nó có tính phản xạ, đối xứng và bắc cầu.

+). Quan hệ R trên tập S được gọi là *quan hệ thứ tự bộ phận* $\Leftrightarrow R$ thỏa mãn các tính chất (i), (iii), (iv)

+). Quan hệ *thứ tự bộ phận* R trên S được gọi là *thứ tự toàn phần* trên $S \Leftrightarrow \forall x, y \in S$, hoặc $(x, y) \in R$ hoặc $(y, x) \in R$.

Ví dụ:

a/ Trên tập S , định nghĩa $xRy \Leftrightarrow x = y$. Dễ kiểm tra cả ba tính chất trên quan hệ = đều thỏa mãn, vậy = là quan hệ tương đương.

b/ Trong tập các sinh viên của Khoa CNTT ta có thể thiết lập quan hệ R : Sinh viên a quan hệ R với sinh viên b nếu cả hai đều học cùng một lớp. Hiển nhiên quan hệ này cũng là quan hệ tương đương. (R = Quan hệ cùng lớp).

c/ Trên tập S , định nghĩa $xRy \Leftrightarrow x > y$. Dễ kiểm tra tính đối xứng không được thỏa mãn, do đó quan hệ $>$ không phải là quan hệ tương đương.

d/ $(R, \leq)$, $(2^A, \subseteq)$ là quan hệ thứ tự bộ phận. $(R, <)$?

e/ Tập $X = \{2, 3, 4, 5, 6, 10, 12, 15, 30, 60\}$ và quan hệ chia hết “ $|$ ” là thứ tự bộ phận trên X . ($x|y \Leftrightarrow \exists k, y = kx$ - đọc là: x chia hết y hoặc y chia hết cho x)

Tập A với quan hệ thứ tự bộ phận R gọi là được sắp theo thứ tự R

Định nghĩa: Giả sử R là quan hệ tương đương trên tập S và $a \in S$. Lớp các phần tử tương đương với a , ký hiệu là $C_a = \{b \mid b \in S \text{ và } a R b\}$. Lớp C_a có thể ký hiệu là $[a]_R$.

Định lý: Tập tất cả các lớp tương đương của quan hệ R trên tập S tạo thành một phân hoạch của S .

Chứng minh: Chúng ta cần chứng minh rằng

$$(i) \quad S = \bigcup_{a \in S} C_a$$

$$(ii) \quad C_a \cap C_b = \emptyset \text{ nếu } C_a \text{ và } C_b \text{ khác nhau, } C_a \neq C_b.$$

Giả sử $s \in S$ và R là quan hệ tương đương, nên $s \in C_s$ (sRs - tính phản xạ). Vì $C_s \subseteq \bigcup_{a \in S} C_a$

nên $S \subseteq \bigcup_{a \in S} C_a$. Mặt khác, theo định nghĩa của C_a , ta luôn có $C_a \subseteq S$ với mọi a của S .

Vậy $\bigcup_{a \in S} C_a \subseteq S$. Từ đó suy ra (i).

Trước khi chứng minh (ii), chúng ta chú ý rằng

$$C_a = C_b \text{ nếu } aRb$$

Bởi vì aRb thì bRa do R đối xứng. Giả sử $d \in C_a$, theo định nghĩa của C_a , aRd . Hơn nữa bRa và aRd , nên căn cứ vào tính chất bắc cầu của quan hệ tương

đương, ta có bRd . Suy ra $d \in C_b$. Từ đó chúng ta có $C_a \subseteq C_b$. Tương tự chúng ta có thể chứng minh ngược lại $C_a \subseteq C_b$. Kết hợp cả hai chúng ta có (1.1).

Chúng ta chứng minh (ii) bằng phản chứng. Giả sử $C_a \cap C_b \neq \emptyset$, nghĩa là tồn tại phần tử $d \in C_a \cap C_b$. Khi $d \in C_a$ thì aRd . Tương tự $d \in C_b$ thì bRd . Vì R đối xứng nên dRb . Kết hợp aRd, dRb suy ra aRb . Sử dụng (1.1) ta có $C_a = C_b$, vậy mâu thuẫn với giả thiết.

Kết luận: Các lớp tương đương là trùng nhau hoặc rời nhau. ■

Ví dụ:

a/ Trên tập số tự nhiên N , nRm khi m và n đều chia hết cho 2 sẽ có hai lớp tương đương là tập các số chẵn và tập các số lẻ.

b/ Các lớp tương đương của quan hệ tương đương trong ví dụ trên (b/) chính là các lớp học đã được xếp trong một Khoa.

Vấn đề tiếp theo được quan tâm nhiều khi nghiên cứu các hành vi của một hệ thống tính toán, đặc biệt các mối quan hệ giữa các phụ thuộc hàm đó là bao đóng của một quan hệ.

Cho trước quan hệ R không có tính phản xạ, không bắc cầu; phải bổ sung bao nhiêu cặp quan hệ với nhau để R có các tính chất đó. Ví dụ $R = \{(1, 2), (2, 3), (1, 1), (3, 3)\}$ trên tập $\{1, 2, 3\}$. R không phản xạ vì $(2, 2) \notin R$. R không có tính bắc cầu vì $(1, 2), (2, 3) \in R$, nhưng $(1, 3) \notin R$. Trong số các quan hệ mở rộng R và có tính phản xạ, bắc cầu, ví dụ $T = \{(1, 2), (2, 3), (1, 3), (1, 1), (2, 2), (3, 3)\}$, chúng ta quan tâm nhất tới quan hệ nhỏ nhất chứa R mà có các tính chất đó.

Định nghĩa: Giả sử R là một quan hệ trên tập S . Bao đóng bắc cầu của R , ký hiệu R^+ là quan hệ nhỏ nhất chứa R và có tính bắc cầu.

Định nghĩa: Giả sử R là một quan hệ trên tập S . Bao đóng phản xạ, bắc cầu của R , ký hiệu R^* là quan hệ nhỏ nhất chứa R , có tính phản xạ và bắc cầu.

Để xây dựng R^+ và R^* , chúng ta định nghĩa quan hệ hợp thành (ghép) của hai quan hệ R_1, R_2 .

Định nghĩa: Giả sử R_1, R_2 là hai quan hệ trên tập S . Hợp thành của R_1, R_2 , ký hiệu $R_1 \circ R_2$ được định nghĩa như sau.

$$(i) \quad R_1 \circ R_2 = \{(a, c) \mid \exists b \in S, aR_1b \text{ và } bR_2c\}$$

$$(ii) \quad R_1^2 = R_1 \circ R_1$$

$$(iii) \quad R_1^n = R_1^{n-1} \circ R_1, \text{ với } n \geq 2.$$

Dựa vào các tính chất của tập hợp và quan hệ, chúng ta có định lý khẳng định sự tồn tại của bao đóng bắc cầu của mọi quan hệ.

Định lý: Giả sử S là tập hữu hạn, R là quan hệ trên S . Bao đóng bắc cầu R^+ của R luôn tồn tại và $R^+ = R^1 \cup R^2 \cup R^3 \dots$. Bao đóng phản xạ, bắc cầu của R là $R^* = R^0 \cup R^+$. Trong đó $R^0 = \{(a, a) \mid a \in S\}$ là quan hệ đồng nhất.

Ví dụ: $R = \{(1, 2), (2, 3), (2, 4)\}$ là quan hệ trên $\{1, 2, 3, 4\}$. Tìm R^+

Chúng ta tính R^i . Lưu ý, nếu có (a, b) và $(b, c) \in R$ thì $(a, c) \in R^2$.

$$R = \{(1, 2), (2, 3), (2, 4)\}$$

$$\begin{aligned} R^2 &= R \circ R = \{(1, 2), (2, 3), (2, 4)\} \circ \{(1, 2), (2, 3), (2, 4)\} \\ &= \{(1, 3), (1, 4)\} \end{aligned}$$

$$\begin{aligned} R^3 &= R^2 \circ R = \{(1, 3), (1, 4)\} \circ \{(1, 2), (2, 3), (2, 4)\} \\ &= \emptyset \text{ (không có cặp nào trong } R^2 \text{ có thể ghép với các cặp trong } R). \end{aligned}$$

$$R^4 = R^5 = \dots = \emptyset.$$

Vậy, $R^+ = R \cup R^2 = \{(1, 2), (2, 3), (2, 4), (1, 4), (1, 3)\}$.

Ví dụ: $R = \{(a, b), (b, c), (c, a)\}$ là quan hệ trên $\{a, b, c\}$. Tìm R^* .

Trước tiên tìm R^+ .

$$\begin{aligned} R^2 &= \{(a, b), (b, c), (c, a)\} \circ \{(a, b), (b, c), (c, a)\} \\ &= \{(a, c), (b, a), (c, b)\} \end{aligned}$$

$$\begin{aligned} R^3 &= \{(a, c), (b, a), (c, b)\} \circ \{(a, b), (b, c), (c, a)\} \\ &= \{(a, a), (b, b), (c, c)\} \end{aligned}$$

$$\begin{aligned} R^4 &= \{(a, a), (b, b), (c, c)\} \circ \{(a, b), (b, c), (c, a)\} \\ &= \{(a, b), (b, c), (c, a)\} = R. \end{aligned}$$

Do vậy, $R^* = R^0 \cup R^+ = R^0 \cup R \cup R^2 \cup R^3$

$$= \{(a, b), (b, c), (c, a), (a, c), (b, a), (c, b), (a, a), (b, b), (c, c)\}$$

Đồ thị và cây (Graf and Tree):

+) Đồ thị là tập các cặp: $G = (V, E)$, trong đó V là tập hữu hạn các đỉnh và E là tập các cặp đỉnh (cạnh).

Đường đi trên đồ thị G là một dãy các đỉnh $v_1, v_2, \dots, v_k$, $k \geq 1$ và lúc đó (v_i, v_{i+1}) là một cạnh, với mỗi i ($1 \leq i < k$). Độ dài đường đi là $k-1$. Nếu $v_1 = v_k$ thì đường đi gọi là chu trình.

+) Đồ thị định hướng: Tập $G = (V, E)$, trong đó V là tập hữu hạn các đỉnh và E là tập các cặp đỉnh có thứ tự (cung) được gọi là đồ thị có hướng.

Ký hiệu một cung từ đỉnh v tới đỉnh w là $v \rightarrow w$.

Đường đi trong đồ thị định hướng: Là một dãy các đỉnh $v_1, v_2, \dots, v_k$, $k \geq 1$ sao cho với mỗi i ($1 \leq i < k$) có một cung từ v_i tới v_{i+1} . Nếu $v \rightarrow w$, thì v là đỉnh trước và w là đỉnh sau.

+) Cây: Một cây (Cây định hướng có thứ tự) là một đồ thị định hướng với các tính chất sau:

(1). Có 1 nút – gọi là nút gốc (không có nút trước nó và có một đường đi từ đó tới mỗi một nút khác).

(2). Mỗi nút khác gốc, có đúng một nút trước.

(3). Các nút sau của một nút đều được sắp thứ tự từ trái qua phải.

(4). Nút không có con gọi là lá của cây.

1.3. Đại cương về ngôn ngữ.

Các khái niệm và định nghĩa:

+ Bảng chữ: Là một tập hợp các ký hiệu (chữ, ký hiệu).

$$\text{Ví dụ: } \Sigma = \{0, 1\}; \quad \Sigma = \{ (,) \}; \dots \Sigma = \{a, b, c, \dots, z\}; \dots$$

Ôtômat và ngôn ngữ hình thức

+ **Xâu, chuỗi con:** Xâu (String, từ) kí tự là dãy hữu hạn các kí hiệu trên bảng chữ nào đó; w_1 là chuỗi con của xâu w nếu w_1 được tạo thành bởi một dãy các kí hiệu kề nhau trong w ; $\#_a w$ = Số kí tự a có trong xâu w .

+ **Độ dài xâu:** Là số kí tự có trong xâu w (ký hiệu là $|w|$). Xâu rỗng ε là xâu không có kí tự nào và $|\varepsilon| = 0$.

+ **Phép ghép của 2 xâu α và β :** $\alpha\beta$ Tạo thành bằng cách viết các kí hiệu của α trước, sau đó là các kí hiệu của β . Độ dài: $|\alpha\beta| = |\alpha| + |\beta|$

Ví dụ: Cho $w = 110010$ là một xâu trên $\Sigma = \{0,1\}$, thì $w_1 = 100$ là chuỗi con, $w_2 = 011$ không phải là chuỗi con của w ; $|w| = 6$; $\#_1 w = 3$.

+ **Đảo ngược xâu w là $w^R = 010011$** (Xâu đảo vị hoặc xâu soi gương).

Các phép toán trên tập xâu

Cho A, B là 2 tập các xâu

+ **Các phép toán thông dụng:**

$$A \cup B = \{ \alpha \mid \alpha \in A \text{ hoặc } \alpha \in B \}$$

$$AB = \{ \alpha\beta \mid \alpha \in A \text{ và } \beta \in B \};$$

$$A \times B = \{ (\alpha, \beta) \mid \alpha \in A \text{ và } \beta \in B \}. \text{ (Tích Đề các).}$$

Lũy thừa của một tập:

$$A^n = \begin{cases} AA^{n-1} & n \geq 1 \\ \{ \varepsilon \} & n=0 \end{cases}$$

+ **Bao đóng, bao đóng dương:** $A^+ = A^1 \cup A^2 \cup A^3 \cup \dots = \bigcup A^i$

$A^* = A^0 \cup A^+$ - là tập tất cả các xâu trên A (lấy kí hiệu của A).

Ví dụ: Cho $A = \{0,1\}$ tập của 2 phần tử 1 và 0.

Ôtômat và ngôn ngữ hình thức

$$A^1 = \{0,1\}; A^2 = AA = \{0,1\}\{0,1\} = \{00, 01, 10, 11\};$$

$$A^3 = AA^2 = \{0,1\}\{00, 01, 10, 11\} = \{000, 001, 010, 011, 100, 101, 110, 111\};$$

$$A^+ = \{0,1, 00,01, 10, 11, 000, \dots\}.$$

Ngôn ngữ: Là một tập hợp các xâu có một cấu trúc được quy định nào đó. Cấu trúc của câu được quy định ra sao thì phải xét đến vấn đề biểu diễn của ngôn ngữ đó.

Ví dụ: Câu lệnh NNLT: + if.....then.....else....
+ program.....end.

Các từ: if, then, else, các từ khóa $\in$ Bảng chữ Σ (từ điển).

Cho Σ là một bảng chữ và $L \subset \Sigma^*$ là một ngôn ngữ trên Σ .

Vấn đề đặt ra: Làm thế nào để biểu diễn L (Chỉ rõ các xâu của L)?

Nếu L hữu hạn thì chỉ cần liệt kê các phần tử. Ngược lại, khi L là tập vô hạn thì chúng ta phải tìm một cách biểu diễn hữu hạn cho 1 ngôn ngữ vô hạn.

Ví dụ: $L_1 = \{a^i \mid i \text{ là số nguyên tố}\};$

$$L_2 = \{a^i b^j \mid i, j \geq 0\};$$

$$L_3 = \{w \in \{a, b\}^* \mid \#_a w = \#_b w\};$$

Tổng quát: Để biểu diễn ngôn ngữ (dưới dạng hình thức), người ta thường sử dụng các công cụ: **Văn phạm hoặc Ôtômat**.

Văn phạm: Là 1 công cụ cho phép sản sinh ra mọi xâu của ngôn ngữ.

Ôtômat: Là 1 công cụ cho phép đoán nhận một xâu w bất kỳ cho trước nào đó có thuộc ngôn ngữ cho trước hay không?

Ví dụ: Cho $\Sigma = \{a, b\}$, L là một ngôn ngữ trên Σ được định nghĩa như sau:

- (1). $\varepsilon \in L$
- (2). If $X \in L$ then $aXb \in L$
- (3). Không còn xâu nào khác $\in L$.

Theo định nghĩa, dễ dàng nhận thấy $L = \{ a^i b^i \mid i = 0, 1, 2, \dots \}$;

1.4. Bài tập:

- 1.1 Chứng minh rằng, nếu X là tập hữu hạn thì $|2^X| = 2^{|X|}$
- 1.2 Những quan hệ R sau có phải là quan hệ tương đương hay không?
 - (a) Trên tập tất cả các đường thẳng: $l_1 R l_2$ nếu l_1 song song với l_2 ,
 - (b) Trên tập các số tự nhiên N : $m R n$ nếu $m - n$ chia hết cho 3,
 - (c) Trên tập các số tự nhiên N : $m R n$ nếu m chia hết cho n ,
 - (d) Trên $S = \{1, 2, \dots, 10\}$: $a R b$ nếu $a + b = 10$.
- 1.3 Cho $f: \{a, b\}^* \rightarrow \{a, b\}^*$ xác định bởi $f(x) = ax$, với $x \in \{a, b\}^*$. Hỏi $f()$ có những tính chất gì?
- 1.4 Chứng minh rằng, nếu $w \in \{a, b\}^*$ thỏa mãn $abw = wab$, thì $|w|$ là số chẵn.

Giải bài 1.4: Ta có thể chứng minh qui nạp theo $|w|$.

- (1). Nếu $w = \varepsilon$, hiển nhiên $ab\varepsilon = \varepsilon ab$. Rõ ràng $|w| = 0$ là số chẵn, bước cơ sở qui nạp được khẳng định. (ε là từ rỗng)
 - (2). Giả sử điều cần chứng minh đúng với $|w| < n$.
 - (3). Ta xét những dãy w có độ dài n và $abw = wab$. Ta sẽ chứng minh $|w|$ chẵn.
- Do $abw = wab$ và $w \in \{a, b\}^*$, $w \neq \varepsilon$, nên $w = abw_1$ với $w_1 \in \{a, b\}^*$ và $|w_1| < n$. Khi đó, $ababw_1 = abw_1ab$ suy ra $abw_1 = w_1ab$. Theo giả thiết qui nạp $|w_1|$ là số chẵn. Từ đó suy ra $|w| = |w_1| + 2$ cũng là số chẵn.

CHƯƠNG 2: VĂN PHẠM VÀ CÁC NGÔN NGỮ HÌNH THỨC

Nhiệm vụ chính của lý thuyết ngôn ngữ hình thức là nghiên cứu các cách đặc tả hữu hạn của các ngôn ngữ vô hạn.

Chương hai, trình bày những khái niệm, các tính chất cơ bản của văn phạm và ngôn ngữ hình thức. Nội dung bao gồm:

- Các khái niệm cơ bản của văn phạm và ngôn ngữ hình thức.
- Phân loại văn phạm và ngôn ngữ hình thức của Chomsky.
- Kỹ thuật tìm ngôn ngữ của Văn phạm G (Bài toán thuận), Bài toán ngược.
- Các phép toán trên các ngôn ngữ và các tính chất của chúng.
- Tính đệ quy của văn phạm cảm ngữ cảnh.

2.1 Văn phạm và các ngôn ngữ

Trong ngôn ngữ tự nhiên, ví dụ tiếng Việt, có hai mẫu câu chính. Những câu mà chúng ta tập trung khảo sát có dạng:

<danh từ> <động từ> <trạng từ> hoặc <danh từ> <động từ>.

Ví dụ: “Ca chạy nhanh” hoặc “Ca chạy”. Trong câu “Ca chạy nhanh” có các từ “Ca”, “chạy” và “nhanh” được viết theo thứ tự xác định. Nếu chúng ta thay các danh từ, động từ và trạng từ bằng các danh từ, động từ và trạng từ khác tương ứng thì vẫn nhận được những câu chính xác về mặt văn phạm. Khi thay “Ca” bằng “Hoa”, thay “chạy” bằng “đi” và thay “nhanh” bằng “chậm” chúng ta vẫn có những câu đúng văn phạm. Tương tự đối với những câu ở dạng thứ hai.

Lưu ý: *<danh từ> <động từ> <trạng từ>* không phải là câu mà chỉ là mô tả một mẫu câu. Nếu chúng ta thay các thành phần *<danh từ>*, *<động từ>*, *<trạng từ>* bằng các từ tương thích thì sẽ nhận được các câu đúng văn phạm.

Chúng ta gọi: *<danh từ>*, *<động từ>*, *<trạng từ>* là *các biến* (variable) hay các ký hiệu chưa kết thúc.

Các từ “Ca”, “chạy”, “nhanh” tương ứng được sử dụng để tạo ra các câu cụ thể là các từ kết thúc hoặc ký hiệu cuối.

Như vậy, *mỗi câu của một ngôn ngữ là một xâu được tạo ra bằng các ký hiệu kết thúc*. Để tiện cho việc mô tả, chúng ta sử dụng S để ký hiệu cho một câu. Khi đó, hai mẫu câu trên có thể được sinh ra từ các quy tắc như sau:

$$\begin{aligned} S &\rightarrow \langle \text{danh từ} \rangle \langle \text{động từ} \rangle \langle \text{trạng từ} \rangle \\ S &\rightarrow \langle \text{danh từ} \rangle \langle \text{động từ} \rangle \\ \langle \text{danh từ} \rangle &\rightarrow \text{Ca} \\ \langle \text{danh từ} \rangle &\rightarrow \text{Hoa} \\ \langle \text{động từ} \rangle &\rightarrow \text{chạy} \\ \langle \text{động từ} \rangle &\rightarrow \text{đi} \\ \langle \text{trạng từ} \rangle &\rightarrow \text{nhanh} \\ \langle \text{trạng từ} \rangle &\rightarrow \text{chậm, ...} \end{aligned}$$

Trong đó, $\rightarrow$ chỉ ra rằng từ bên phải (vế phải) thay thế cho từ bên trái (vế trái).

Ký hiệu P là tập tất cả các qui tắc như trên, được gọi là các qui tắc dẫn xuất (gọi tắt là luật sinh); V là tập các thành phần của câu như: $\langle \text{danh từ} \rangle$, $\langle \text{động từ} \rangle$, $\langle \text{trạng từ} \rangle$ và bảng từ vựng là Σ bao gồm như: “Ca”, “Hoa”, “chạy”, “đi”, “nhanh”, “chậm”, ... Từ đó chúng ta có thể định nghĩa hình thức văn phạm là:

Một bộ 4 thành phần $G = (\Sigma, V, P, S)$, trong đó

$\Sigma = \{\text{Ca, Hoa, chạy, đi, nhanh, chậm}\}$ - các ký hiệu kết thúc

$V = \{(\langle \text{danh từ} \rangle, \langle \text{động từ} \rangle, \langle \text{trạng từ} \rangle)\}$ - các ký hiệu chưa kết thúc,

P là tập các luật sinh như trên và S là ký hiệu (biến) đặc biệt khởi đầu để tạo sinh một câu. Các câu được tạo ra như sau:

- (i) Bắt đầu từ S ,
- (ii) Lần lượt thay thế các ký hiệu chưa kết thúc (biến) bằng những ký hiệu kết thúc theo luật sinh.
- (iii) Kết thúc, khi xâu nhận được chỉ toàn là những ký hiệu kết thúc.

Tổng quát hóa quá trình nêu trên, chúng ta định nghĩa văn phạm hình thức như sau.

Định nghĩa. Văn phạm là một bộ 4 thành phần $G = (\Sigma, V, P, S)$, trong đó

- (i) V là tập hữu hạn khác rỗng các phần tử được gọi là các biến, hay các ký hiệu không kết thúc,
- (ii) Σ là tập hữu hạn khác rỗng các phần tử được gọi là các ký hiệu kết thúc, hay bảng chữ cái,
- (iii) $V \cap \Sigma = \emptyset$, tập các biến và tập các ký hiệu kết thúc là rời nhau,
- (iv) S là ký hiệu đặc biệt của V được gọi là ký hiệu bắt đầu,
- (v) P là tập hữu hạn các luật sinh dạng $\alpha \rightarrow \beta$, trong đó α, β là các xâu trên $V \cup \Sigma$ và vế trái α có ít nhất một ký hiệu chưa kết thúc (các biến).

Ôtômat và ngôn ngữ hình thức

Lưu ý: Tập các luật sinh là hạt nhân của văn phạm và nó đặc tả ngôn ngữ sinh bởi văn phạm đó. Đối với các luật sinh chúng ta lưu ý:

- (i) Không cho phép thay thế ngược. Ví dụ, nếu $S \rightarrow AB$ là một qui tắc thì chúng ta có thể thay S bằng AB , nhưng ngược lại không thể thay AB bằng S được.
- (ii) Không cho phép dẫn xuất ngược. Ví dụ, nếu $S \rightarrow AB$ thì không đủ để có được qui tắc $AB \rightarrow S$.

Ví dụ 2.1 $G = (\Sigma, V, P, S)$ là văn phạm, trong đó

$V = \{ \langle \text{câu} \rangle, \langle \text{danh từ} \rangle \langle \text{động từ} \rangle, \langle \text{trạng từ} \rangle \},$

$\Sigma = \{ \text{Ca, Nam, Hoa, chạy, đi, ăn, nhanh, chậm} \},$

$S = \langle \text{câu} \rangle$

$P = \{ \langle \text{câu} \rangle \rightarrow \langle \text{danh từ} \rangle \langle \text{động từ} \rangle,$

$\langle \text{câu} \rangle \rightarrow \langle \text{danh từ} \rangle \langle \text{động từ} \rangle \langle \text{trạng từ} \rangle,$

$\langle \text{danh từ} \rangle \rightarrow \text{Ca}, \langle \text{danh từ} \rangle \rightarrow \text{Nam},$

$\langle \text{danh từ} \rangle \rightarrow \text{Hoa}, \langle \text{động từ} \rangle \rightarrow \text{chạy},$

$\langle \text{động từ} \rangle \rightarrow \text{đi}, \langle \text{động từ} \rangle \rightarrow \text{ăn},$

$\langle \text{trạng từ} \rangle \rightarrow \text{nhanh}, \langle \text{trạng từ} \rangle \rightarrow \text{chậm} \}$

Để tiện lợi và đơn giản, trong các phần sau chúng ta sử dụng các ký hiệu:

- a) Nếu A là một tập bất kỳ, A^* ký hiệu tập tất cả các xâu trên A , $A^+ = A^* - \Lambda$, với Λ là xâu rỗng,
- b) Chữ viết hoa $A, B, A_1, A_2, \dots$ ký hiệu cho các biến (ký hiệu chưa kết thúc),
- c) Chữ thường $a, b, c, \dots$ ký hiệu cho các từ kết thúc, các chữ cái,
- d) $x, y, z, w, \dots$ ký hiệu cho các xâu gồm toàn các ký hiệu kết thúc,
- e) $\alpha, \beta, \gamma, \dots$ ký hiệu cho các phần tử của $(V \cup \Sigma)^*$, các xâu có cả từ kết thúc lẫn chưa kết thúc (dạng xâu),
- f) $X^0 = \Lambda$, với mọi tập $X \subseteq V \cup \Sigma$.

2.2 Các dẫn xuất và ngôn ngữ sinh bởi văn phạm

Các luật sinh được sử dụng để sinh ra một xâu trên $V \cup \Sigma$ từ một xâu cho trước. Quá trình sử dụng liên tục các luật sinh được gọi là một *dẫn xuất* (suy dẫn).

Ôtômat và ngôn ngữ hình thức

Định nghĩa. Nếu $\alpha \rightarrow \beta$ là luật sinh trong văn phạm G và Υ, δ là hai xâu trên $V \cup \Sigma$, khi đó xâu $\Upsilon\alpha\delta$ suy dẫn trực tiếp ra $\Upsilon\beta\delta$ và được viết là $\Upsilon\alpha\delta \Rightarrow_G \Upsilon\beta\delta$.

Đối với luật sinh dạng $\alpha \rightarrow \beta$. Trong G ta có $\alpha \Rightarrow_G \beta$ và viết đơn giản $\alpha \Rightarrow \beta$.

Ví dụ 2.2 $G = (\{0, 1\}, \{S\}, \{S \rightarrow 0S1, S \rightarrow 01\}, S)$.

Do trong P có qui tắc $S \rightarrow 0S1$ nên $0^5S1^5 \Rightarrow_G 0^50S11^5 = 0^6S1^6$.

Các khái niệm: Suy dẫn $\Rightarrow$ và $\Rightarrow^*$

+) Xâu x suy dẫn trực tiếp xâu y trên $V \cup \Sigma$ (Viết $x \Rightarrow y$) $\Leftrightarrow \exists$ các xâu x_1, v, x_2, w sao cho: $x = x_1 v x_2, y = x_1 w x_2$ và $v \rightarrow w \in P$.

+) Xâu x suy dẫn xâu y ($x \Rightarrow^* y$) $\Leftrightarrow \exists x_0, x_1, \dots, x_k$ ($k \geq 0$) trên $V \cup \Sigma$ sao cho:

$x = x_0, \dots, x_k = y$ và $x_i \Rightarrow x_{i+1}$ ($0 \leq i \leq k-1$); k là độ dài suy dẫn.

+) Ngôn ngữ được sinh bởi Văn phạm G ($L(G)$) được định nghĩa:

$$L(G) = \{ w \mid w \in \Sigma^* \text{ và } S \Rightarrow^* w \}$$

Ví dụ: Cho $G = (\Sigma, V, P, S)$ với $\Sigma = \{a, b\}$; $V = \{S\}$;

$$P = \{S \rightarrow aSb, S \rightarrow ab\}, \text{ thì } L(G) = \{a^n b^n \mid n \in \mathbb{N}, n \geq 1\}.$$

Một cách mô tả khác về suy dẫn:

Dựa vào dẫn xuất $\Rightarrow_G$ chúng ta có thể thiết lập được quan hệ R trên $(V \cup \Sigma)^*$ như sau:

$$\alpha R \beta \text{ nếu } \alpha \Rightarrow_G \beta.$$

Định nghĩa. Giả sử α và β là các xâu trên $V \cup \Sigma$, chúng ta nói rằng α suy dẫn ra (dẫn xuất ra) β , nếu $\alpha R^* \beta$. Nói cách khác, α suy dẫn ra β nếu $\alpha \xRightarrow{*}_G \beta$, trong đó $\xRightarrow{*}_G$ là bao đóng phản xạ, bắc cầu của quan hệ $\Rightarrow_G$ trên tập $(V \cup \Sigma)^*$.

Chú ý: (i) Dựa vào tính chất của quan hệ $\xRightarrow{*}_G$, hiển nhiên $\alpha \xRightarrow{*}_G \alpha$ với mọi α

(iii) Khi $\alpha \xRightarrow{*}_G \beta$, $\alpha \neq \beta$ nghĩa là tồn tại một dãy $\alpha_1, \alpha_2, \dots, \alpha_n$, $n \geq 2$ sao cho $\alpha = \alpha_1 \xRightarrow{G} \alpha_2 \xRightarrow{G} \dots \xRightarrow{G} \alpha_n = \beta$. Khi $\alpha \xRightarrow{*}_G \beta$ thực hiện trong n bước dẫn xuất, ta viết $\alpha \xRightarrow{n}_G \beta$.

Định nghĩa.

Ngôn ngữ sinh bởi văn phạm G , ký hiệu $L(G)$, là tập tất cả các xâu gồm các ký hiệu kết thúc thu được từ ký hiệu đầu tiên S trong G sau một số lần dẫn xuất. Các phần tử của $L(G)$ gọi là các xâu (hay các từ) của ngôn ngữ đó. Hiển nhiên,

$$L(G) = \{w \in \Sigma^* \mid S \xRightarrow{*}_G w\}$$

Ta nói: Ngôn ngữ $L(G)$ được sinh bởi văn phạm G hoặc G sinh ra L .

Định nghĩa. Văn phạm G_1 tương đương với văn phạm G_2 nếu $L(G_1) = L(G_2)$.

Ví dụ 2.3 Cho văn phạm $G_1 = (\{a, b\}, \{S, A, B\}, \{S \rightarrow ASB, S \rightarrow AB, A \rightarrow a, B \rightarrow b\}, S)$ và $G_2 = (\{a, b\}, \{S'\}, \{S' \rightarrow aS'b, S' \rightarrow ab\}, S')$.

Dễ nhận thấy rằng $L(G_1) = L(G_2) = \{a^n b^n \mid n \geq 1\}$ và do vậy hai văn phạm này tương đương với nhau.

Lưu ý:

- Mọi dẫn xuất đều phải thực hiện trên cơ sở sử dụng các luật sinh; khi số các luật sinh sử dụng đúng một lần thì chúng ta viết $\alpha \xRightarrow{G} \beta$; khi số luật sinh lớn hơn một lần thì viết là $\alpha \xRightarrow{*}_G \beta$.
- Xâu không chứa các biến, là một xâu (câu) của ngôn ngữ sinh bởi văn phạm G cho trước.

Ký hiệu: (a) Khi G xác định thì chúng ta viết đơn giản $\alpha \Rightarrow \beta$ thay cho $\alpha \xRightarrow{*}_G \beta$.

(b) Nếu $A \rightarrow \alpha$, $A \in V$ là luật sinh trong P , khi đó chúng ta gọi là A -dẫn xuất.

- (c) Khi có $A \rightarrow \alpha_1, A \rightarrow \alpha_2, \dots, A \rightarrow \alpha_m$ là các A -dẫn xuất, chúng ta viết ngắn gọn $A \rightarrow \alpha_1 \mid \alpha_2 \mid \dots \mid \alpha_m$.

Sau đây chúng ta xét một số văn phạm và các ngôn ngữ sinh bởi chúng.

Bài toán thuận (Cho trước Văn phạm G , tìm ngôn ngữ sinh bởi G).

Ví dụ 2.4 Cho trước văn phạm $G = (\{0, 1\}, \{S\}, \{S \rightarrow 0S1, S \rightarrow \Lambda\}, S)$, tìm $L(G)$.

Vì $S \rightarrow \Lambda$ nên $S \xRightarrow{G} \Lambda$, nghĩa là $\Lambda \in L(G)$. Hơn nữa, với mọi $n \geq 1$,

$$S \xRightarrow{G} 0S1 \xRightarrow{G} 0^2S1^2 \xRightarrow{G} \dots \xRightarrow{G} 0^nS1^n \xRightarrow{G} 0^n1^n$$

Vì thế $0^n1^n \in L(G)$, với $n \geq 0$.

Để chỉ ra rằng $L(G) \subseteq \{0^n1^n \mid n \geq 0\}$, chúng ta bắt đầu xét một câu w bất kỳ thuộc $L(G)$. Vì mọi câu $w \in L(G)$ đều phải xuất phát từ S . Nên, nếu sử dụng $S \rightarrow \Lambda$ (ký hiệu rỗng) trước thì ta nhận được Λ . Trong trường hợp này $w = \Lambda = 0^01^0$. Ngược lại, khi sử dụng luật sinh $S \rightarrow 0S1$ thì sau đó chúng ta có thể hoặc sử dụng $S \rightarrow \Lambda$ hoặc tiếp tục sử dụng $S \rightarrow 0S1$. Quá trình dẫn xuất đó của w có dạng $S \xRightarrow{*} 0^nS1^n \xRightarrow{G} 0^n1^n$, nghĩa là $L(G) \subseteq \{0^n1^n \mid n \geq 0\}$

Vậy $L(G) = \{0^n1^n \mid n \geq 0\}$.

Ví dụ 2.5 Cho trước văn phạm $G = (\{a, b\}, \{S, C\}, P, S)$, trong đó $P = \{S \rightarrow aCa, C \rightarrow aCa \mid b\}$. Tìm $L(G)$.

Vì $S \rightarrow aCa \rightarrow aba$, vậy $aba \in L(G)$. Hơn nữa

$$S \rightarrow aCa$$

$$S \xRightarrow{G} a^nCa^n \text{ (áp dụng } n-1 \text{ lần luật sinh } C \rightarrow aCa)$$

$$\xRightarrow{G} a^nba^n \text{ (áp dụng luật sinh } C \rightarrow b).$$

Suy ra $a^nba^n \in L(G)$, $n \geq 1$; nghĩa là $\{a^nba^n \mid n \geq 1\} \subseteq L(G)$.

Mặt khác, mọi câu của $L(G)$ đều được dẫn ra và được bắt đầu bằng luật sinh $S \rightarrow aCa$. Nếu áp dụng $C \rightarrow b$ thì ta nhận được aba . Ngược lại áp dụng một số lần $C \rightarrow aCa$ thì

nhận được $a^n C a^n$, C là biến và $n \geq 1$. Sau đó áp dụng $C \rightarrow b$, chúng ta có $a^n b a^n$. Từ đó suy ra

$$L(G) \subseteq \{ a^n b a^n \mid n \geq 1 \}$$

Kết hợp hai bao hàm thức trên chúng ta có

$$L(G) = \{ a^n b a^n \mid n \geq 1 \}$$

Ví dụ 2.6 Cho trước G có các luật sinh $S \rightarrow aS \mid bS \mid a \mid b$. Tìm $L(G)$.

Chúng ta chỉ ra rằng $L(G) = \{a, b\}^+$.

Trước tiên chúng ta nhận xét G chỉ có hai ký hiệu kết thúc a, b , nên $L(G) \subseteq \{a, b\}^*$. Tất cả các luật sinh của G đều là S -đẫn xuất, vậy Λ thuộc $L(G)$ chỉ khi $S \rightarrow \Lambda$ là một luật sinh của G . Nhưng trong văn phạm trên không có luật sinh đó, nên

$$L(G) \subseteq \{a, b\}^* - \{\Lambda\} = \{a, b\}^+$$

Để chứng minh $\{a, b\}^+ \subseteq L(G)$, hãy xét dãy $a_1 a_2 \dots a_n$, trong đó a_i là a hoặc là b . Luật sinh đầu tiên có thể áp dụng để suy ra $a_1 a_2 \dots a_n$ có thể là $S \rightarrow aS$ hoặc $S \rightarrow bS$, tùy thuộc $a_1 = a$ hoặc $a_1 = b$. Tương tự xét với $a_2, a_3, \dots, a_{n-1}$. Cuối cùng đối với a_n chúng ta áp dụng $S \rightarrow a$ hoặc $S \rightarrow b$ tùy thuộc $a_n = a$ hoặc $a_n = b$. Vậy

$$L(G) = \{a, b\}^+.$$

Ví dụ 2.7 Cho $G = (\{0, 1, 2\}, \{S, A\}, P, S)$, trong đó P gồm:

1. $S \rightarrow 0SA2$,
2. $S \rightarrow 012$,
3. $2A \rightarrow A2$,
4. $1A \rightarrow 11$.

Chứng minh rằng $L(G) = \{0^n 1^n 2^n \mid n \geq 1\}$.

Khi áp dụng $S \rightarrow 012$ thì $S \Rightarrow 012$, do vậy $012 \in L(G)$.

Nếu áp dụng $S \rightarrow 0SA2$

$$S \xRightarrow{*} 0^{n-1} S (A2)^{n-1} \quad (\text{áp dụng } S \rightarrow 0SA2 \text{ } n-1 \text{ lần})$$

$$\stackrel{*}{\Rightarrow} 0^{n-1}012(A2)^{n-1} \quad (\text{áp dụng } S \rightarrow 012)$$

$$\stackrel{*}{\Rightarrow} 0^n 1 A^{n-1} 2^n \quad (\text{áp dụng } 2A \rightarrow A2)$$

$$\stackrel{*}{\Rightarrow} 0^n 1^n 2^n \quad (\text{áp dụng } 1A \rightarrow 11 \text{ } n-1 \text{ lần}).$$

Vậy, $0^n 1^n 2^n \in L(G)$, $n \geq 1$.

Để chứng minh $L(G) \subseteq \{0^n 1^n 2^n \mid n \geq 1\}$, chúng ta thực hiện như sau: Nếu đầu tiên sử dụng $S \rightarrow 012$ thì sẽ nhận được 012. Ngược lại sử dụng $S \rightarrow 0SA2$ một số lần thì sẽ có $0^{n-1}S(A2)^{n-1}$. Để loại được S thì bắt buộc phải sử dụng $S \rightarrow 012$ và sẽ nhận được $0^n 12(A2)^{n-1}$. Muốn loại tiếp A chúng ta sử dụng $2A \rightarrow A2$ hoặc $1A \rightarrow 11$. Luật sinh $2A \rightarrow A2$ chỉ đổi chỗ của A và 2. Sử dụng $1A \rightarrow 11$ là loại được biến A. Do vậy,

$$0^n 12(A2)^{n-1} = 0^n 12A2 A2 \dots A2, \text{ trong đó có } n-1 \text{ lần cặp } A2.$$

Áp dụng $n-1$ lần $2A \rightarrow A2$,

$$0^n 12(A2)^{n-1} \stackrel{*}{\Rightarrow} 0^n 1 A^{n-1} 2^n$$

Sau đó áp dụng $n-1$ lần $1A \rightarrow 11$ thì được

$$0^n 12(A2)^{n-1} \stackrel{*}{\Rightarrow} 0^n 1^n 2^n$$

Suy ra $L(G) \subseteq \{0^n 1^n 2^n \mid n \geq 1\}$ và cuối cùng $L(G) = \{0^n 1^n 2^n \mid n \geq 1\}$.

Bài toán ngược (Cho trước ngôn ngữ L, tìm Văn phạm G để $L(G)=L$).

Ví dụ 2.8 Xây dựng văn phạm G, sinh ra ngôn ngữ $L = \{wcw^T \mid w \in \{a, b\}^*\}$, trong đó w^T là xâu soi gương của xâu w, là xâu viết theo thứ tự ngược lại của w, thường được gọi là phép đảo vị. Ví dụ, $(abcde)^T = edcba$.

Chúng ta có thể chứng minh được $G = (\{a, b, c\}, \{S\}, P, S)$, với $P = \{S \rightarrow aSa \mid bSb \mid c\}$ sẽ sinh ra chính ngôn ngữ L.

Ví dụ 2.9 Tìm văn phạm G sinh ra ngôn ngữ $L = \{a^n b^n c^i \mid n \geq 1, i \geq 0\}$.

Để nhận thấy $L = L_1 \cup L_2$, với $L_1 = \{a^n b^n \mid n \geq 1\}$ (trường hợp $i = 0$) và

$$L_2 = \{a^n b^n c^i \mid n \geq 1, i \geq 1\}$$

Ngôn ngữ L_1 có thể được sinh bởi văn phạm có các luật sinh $S \rightarrow A, A \rightarrow ab, A \rightarrow aAb$ và L_2 là kết quả của phép ghép L_1 với $c^i, i \geq 1$. Vậy văn phạm $G = (\{a, b, c\}, \{S, A\}, P, S)$, trong đó

$$P = \{S \rightarrow A, A \rightarrow ab, A \rightarrow aAb, S \rightarrow Sc\}$$

Với $n \geq 1, i \geq 0$, chúng ta có

$$S \xRightarrow{*} Sc^i \xRightarrow{*} Ac^i \xRightarrow{*} a^{n-1}Ab^{n-1}c^i \Rightarrow a^{n-1}abb^{n-1}c^i = a^n b^n c^i$$

Suy ra $\{a^n b^n c^i \mid n \geq 1, i \geq 0\} \subseteq L(G)$.

Để chứng minh chiều ngược lại, lưu ý rằng mọi xâu của $L(G)$ đều phải bắt đầu bằng một trong các S-dẫn xuất: $S \rightarrow Sc$ hoặc $S \rightarrow A$.

Nếu bắt đầu bằng $S \rightarrow A$ và sau đó áp dụng một số lần $A \rightarrow aAb$ rồi cuối cùng áp dụng $A \rightarrow ab$ thì:

$$S \rightarrow A \xRightarrow{*} a^{n-1}Ab^{n-1} \xRightarrow{*} a^n b^n = a^n b^n c^0 \in L(G).$$

Nếu chúng ta bắt đầu bằng luật sinh $S \rightarrow Sc$ và lặp lại một số lần thì sẽ nhận được Sc^i . Sau đó áp dụng các luật sinh còn lại như trên thì chúng ta có $a^n b^n c^i \in L(G)$.

Kết luận: $L(G) = \{a^n b^n c^i \mid n \geq 1, i \geq 0\}$.

Ví dụ 2.10 Xây dựng văn phạm G để sinh ra $L = \{a^n b^n c^n \mid n \geq 1\}$.

Chúng ta có thể xây dựng văn phạm G để sinh ra L bằng phương pháp đệ qui. Tuy nhiên, nếu xây dựng văn phạm sinh trực tiếp ra L là tương đối khó. Vậy, chúng ta có thể thực hiện theo hai bước.

- (i) Xác định các qui tắc để dẫn xuất ra $a^n \alpha^n$,
- (ii) Chuyển α^n về $b^n c^n$.

Ôtômát và ngôn ngữ hình thức

Để có các xâu $a^n\alpha^n$ dạng (i), chúng ta có thể sử dụng những luật sinh đơn giản như $S \rightarrow aS\alpha \mid a\alpha$. Nếu ở bước (ii) chúng ta sử dụng ngay luật sinh $\alpha \rightarrow bc$ thì sẽ được $(bc)^n$. Nhưng đến đây không thể chuyển $(bc)^n$ về được b^nc^n vì vế trái không có biến nào cả. Do vậy ta có thể sử dụng $\alpha = BC$, với B và C là các biến. Sau đó dồn các biến B về bên trái còn biến C về bên phải bằng cách sử dụng qui tắc $CB \rightarrow BC$. Từ những lưu ý trên, chúng ta có thể định nghĩa $G = (\{a, b, c\}, \{S, B, C\}, P, S)$, trong đó

$$P = \{ S \rightarrow aSBC \mid aBC, CB \rightarrow BC, aB \rightarrow ab, bB \rightarrow bb, bC \rightarrow bc, cC \rightarrow cc \}$$

Tiếp theo chúng ta chứng minh rằng $L(G) = \{a^n b^n c^n \mid n \geq 1\}$.

Vì $S \Rightarrow aBC \Rightarrow abC \Rightarrow abc$, nên $abc \in L(G)$.

$$S \xRightarrow{*} a^{n-1}S(BC)^{n-1} \quad (\text{áp dụng } n-1 \text{ lần } S \rightarrow aSBC)$$

$$\Rightarrow a^{n-1}aBC(BC)^{n-1} \quad (\text{áp dụng qui tắc } S \rightarrow aBC)$$

$$\Rightarrow a^n B^n C^n \quad (\text{áp dụng } n-1 \text{ lần qui tắc } CB \rightarrow BC)$$

$$\Rightarrow a^{n-1}abB^{n-1}C^n \quad (\text{áp dụng 1 lần qui tắc } aB \rightarrow ab)$$

$$\Rightarrow a^n b^n C^n \quad (\text{áp dụng } n-1 \text{ lần qui tắc } bB \rightarrow bb)$$

$$\Rightarrow a^n b^{n-1}bc C^{n-1} \quad (\text{áp dụng 1 lần qui tắc } bC \rightarrow bc)$$

$$\Rightarrow a^n b^n c^n \quad (\text{áp dụng } n-1 \text{ lần qui tắc } cC \rightarrow cc)$$

Từ đó chúng ta có $L(G) \subseteq \{a^n b^n c^n \mid n \geq 1\}$.

Chiều ngược lại sẽ được khẳng định nếu mọi xâu $a^n b^n c^n$, $n \geq 1$ đều dẫn xuất ra được từ S-dẫn xuất ($S \xRightarrow{*} a^n b^n c^n$).

Thật vậy, nếu $n = 1$ và sử dụng các luật sinh $S \rightarrow aBC$, $aB \rightarrow ab$, $bB \rightarrow bb$, $bC \rightarrow bc$, $cC \rightarrow cc$ thì sẽ thu được abc .

Với $n > 1$ thì sử dụng $n-1$ lần $S \rightarrow aSBC$ và 1 lần $S \rightarrow aBC$ chúng ta sẽ có $a^n(BC)^n$. Tiếp theo chúng ta có thể sử dụng $CB \rightarrow BC$ hoặc $aB \rightarrow ab$ sau đó thêm $bB \rightarrow bb$, $bC \rightarrow bc$, $cC \rightarrow cc$. Sử dụng một trong những luật sinh trên chúng ta sẽ nhận được các xâu có các biến B, C nằm cả ở bên phải. Khi đó xảy ra các trường hợp sau:

+ Nếu B được dồn hết về bên trái của C thì sau đó chỉ có thể áp dụng $bB \rightarrow bb$ cho đến khi loại hết biến B và sau đó sử dụng $bC \rightarrow bc$, $cC \rightarrow cc$ để cuối cùng có được $a^n b^n c^n$.

+ Giả sử biến C được chuyển thành c trước khi tất cả các biến B được chuyển thành b , nghĩa là $a^n(BC)^n \xRightarrow{*} a^n b^i c \alpha$, $i < n$ và α chứa cả biến B lẫn biến C .

Vì trong $a^n b^i c \alpha$ chỉ có α là chứa các biến chưa kết thúc và vì c đứng trước α nên trong các qui tắc của G chỉ có $cC \rightarrow cc$ là có thể được áp dụng. Nếu α không còn dẫn xuất tiếp được nữa (không có qui tắc nào có thể áp dụng) và bắt đầu bằng B thì không thể dẫn xuất được thành xâu không còn biến. Ngược lại, α bắt đầu bằng C thì lặp lại việc áp dụng luật sinh $cC \rightarrow cc$ cho đến khi gặp phải ít nhất biến B , dạng $a^n b^i c^k B \alpha'$, $i, k < n$. Trường hợp này cũng tương tự như trên, không thể sử dụng được luật sinh nào để loại bỏ được biến B . Do vậy, trường hợp thứ 2 này không thể dẫn xuất ra được câu của $L(G)$.

Kết hợp cả 2 trường hợp chúng ta có $\{a^n b^n c^n \mid n \geq 1\} \subseteq L(G)$ và cuối cùng là

$$\{a^n b^n c^n \mid n \geq 1\} = L(G)$$

Ví dụ 2.11 Cho trước G với các luật sinh $S \rightarrow aSa \mid bSb \mid aa \mid bb \mid \Lambda$. Chứng minh rằng:

- (i) Mọi xâu (câu) của $L(G)$ đều có độ dài chẵn,
- (ii) Số các xâu có độ dài $2 \cdot n$ là 2^n .

Mệnh đề đầu (i) dễ khẳng định vì khi áp dụng bất kỳ một luật sinh nào của G cũng đều thay một ký hiệu không kết thúc (S) bằng hai ký hiệu kết thúc (a hoặc b) (trừ $S \rightarrow \Lambda$) và có nhiều nhất một ký hiệu không kết thúc. Do vậy, qua mỗi lần áp dụng một luật sinh thì độ dài của xâu được dẫn ra tăng lên 2 trừ qui tắc $S \rightarrow \Lambda$.

Để chứng minh mệnh đề (ii), chúng ta xét xâu w có độ dài 2^n . Dựa vào các luật sinh của G , chúng ta thấy w có dạng $a_1 a_2 \dots a_n a_n \dots a_2 a_1$, a_i có dạng a hoặc b . Vậy số tất cả các xâu như thế sẽ là 2^n .

2.3 Phân loại ngôn ngữ (Văn phạm) của Chomsky

Cho trước văn phạm $G = (\Sigma, V, P, S)$. Muốn phân loại được các ngôn ngữ $L(G)$, chúng ta phải dựa vào các dạng khác nhau của luật sinh P . Chomsky đã chia các luật sinh của văn phạm G thành 04 loại.

Khái niệm về *ngữ cảnh*.

Xét luật sinh dạng $\phi A \psi \rightarrow \phi \alpha \psi$, trong đó A là biến; ϕ được gọi là *ngữ cảnh trái* còn ψ gọi là *ngữ cảnh phải* và $\phi \alpha \psi$ là xâu thay thế. Qua đó chúng ta hiểu là chỉ có thể thực hiện dẫn xuất theo ngữ cảnh và có được xâu thay thế trong ngữ cảnh trái và phải như đã xác định.

Ví dụ 2.12

- (i) Trong $abAbcd \rightarrow abBAbcd$, ab là ngữ cảnh trái và bcd là ngữ cảnh phải,
- (ii) $AB \rightarrow A$, A là ngữ cảnh trái, Λ là ngữ cảnh phải và $\alpha = \Lambda$.
- (iii) $B \rightarrow \Lambda$, cả ngữ cảnh trái, phải và α đều là Λ . Qui tắc này cho phép xoá biến B trong mọi ngữ cảnh.

Định nghĩa. (Phân loại Chomsky): Cho trước văn phạm $G = (\Sigma, V, P, S)$.

1. Nếu các luật sinh trong P không có ràng buộc nào cả thì được gọi là văn phạm loại 0, hay còn gọi là văn phạm ngữ cấu, văn phạm cấu trúc câu và ngôn ngữ sinh bởi nó được gọi là *ngôn ngữ loại 0 (ngữ cấu)*. Các luật sinh dạng trên được gọi là luật sinh loại 0.
2. Nếu các luật sinh của P có dạng $\phi A \psi \rightarrow \phi \alpha \psi$ với $\alpha \neq \Lambda$ thì văn phạm được gọi là loại 1 (cảm ngữ cảnh, phụ thuộc vào ngữ cảnh) và ngôn ngữ sinh bởi nó được gọi là *ngôn ngữ loại 1 (cảm ngữ cảnh)*. Các luật sinh dạng trên được gọi là luật sinh loại 1. Luật sinh $S \rightarrow \Lambda$ được phép xuất hiện trong văn phạm loại 1, nhưng khi đó S không được xuất hiện ở vế phải của bất kỳ luật sinh nào khác và khi đó G được gọi là văn phạm cảm ngữ cảnh suy rộng.
3. Nếu các luật sinh của P có dạng $A \rightarrow \alpha$ với $A \in V$ và $\alpha \in (V \cup \Sigma)^*$ thì văn phạm được gọi là loại 2 (*phi ngữ cảnh*, không phụ thuộc vào ngữ cảnh) và ngôn ngữ sinh bởi nó được gọi là *ngôn ngữ loại 2 (phi ngữ cảnh)*. Các luật sinh dạng trên được gọi là luật sinh loại 2.

4. Nếu các luật sinh của P có dạng $A \rightarrow a$ hoặc $A \rightarrow aB$ với $A, B \in V$ và $a \in \Sigma$ thì văn phạm được gọi là loại 3 (chính qui) và ngôn ngữ sinh bởi nó được gọi là *ngôn ngữ loại 3 (chính qui)*. Các luật sinh dạng trên được gọi là luật sinh loại 3. Luật sinh $S \rightarrow \Lambda$ được phép xuất hiện trong văn phạm loại 3, nhưng khi đó S không được xuất hiện ở vế phải của bất kỳ luật sinh nào khác. Văn phạm chính qui có luật sinh $S \rightarrow \Lambda$ được gọi là văn phạm chính qui suy rộng.

Lưu ý:

- ✓ Trong văn phạm cảm ngữ cảnh, chúng ta có thể sử dụng luật sinh $A \rightarrow \Lambda$ để xoá đi một biến A mà không có điều kiện nào. Luật sinh này không phải là loại 1 mà là loại 1 mở rộng.
- ✓ Tất cả các luật sinh loại 1: $\phi A \psi \rightarrow \phi \alpha \psi$ với $\alpha \neq \Lambda$ đều không làm giảm độ dài của xâu thay thế, nghĩa là $|\phi A \psi| \leq |\phi \alpha \psi|$. Tuy nhiên, nếu $\alpha \rightarrow \beta$ là luật sinh với $|\alpha| \leq |\beta|$ thì chưa hẳn nó là loại 1. Ví dụ, $AB \rightarrow BA$ không phải là luật sinh loại 1. Phần sau chúng ta sẽ chỉ ra rằng những luật sinh như thế có thể thay bằng luật sinh loại 1.
- ✓ Vế trái của luật sinh loại 2 (phi ngữ cảnh) luôn là một biến. Trong văn phạm loại 2 (phi ngữ cảnh) có thể có luật sinh $A \rightarrow \Lambda$, loại luật sinh rộng cho phép xoá đi các biến trong các dẫn xuất.
- ✓ Văn phạm chính qui là văn phạm phi ngữ cảnh; văn phạm phi ngữ cảnh và văn phạm cảm ngữ cảnh là văn phạm loại 0. Phần sau chúng ta khẳng định thêm văn phạm phi ngữ cảnh cũng là văn phạm cảm ngữ cảnh, chính xác hơn là có thể loại luật sinh xoá biến tự do $A \rightarrow \Lambda$ để mọi luật sinh loại 2 cũng là loại 1 mà vẫn đảm bảo tương đương giữa các văn phạm.

Mô tả phân loại ngắn gọn hơn:

Chomsky chia Văn phạm $G = (\Sigma, V, P, S)$ thành 4 loại dựa vào tập luật sinh P như sau:

Loại 0 (ngữ cấu): Khi $P: \alpha \rightarrow \beta$, không có ràng buộc gì.

Loại 1 (Cảm ngữ cảnh): Khi $P: \alpha \rightarrow \beta$, với $\alpha \in (V \cup \Sigma)^+$, $\beta \in (V \cup \Sigma)^*$ và $|\alpha| \leq |\beta|$.

Loại 2 (Phi ngữ cảnh): Khi $P: \alpha \rightarrow \beta$, với $\alpha \in V$, $\beta \in (V \cup \Sigma)^*$.

Loại 3 (Chính qui): Nếu P có dạng: $A \rightarrow aB$, $A \rightarrow a$ hoặc $A \rightarrow Ba$, $A \rightarrow a$

Với $A, B \in V$, $a \in \Sigma$.

Văn phạm loại nào sinh ra ngôn ngữ loại đó;

Hệ quả: $NNCQ \subseteq NNPNC \subseteq NNCNC \subseteq NNNC$

(3) (2) (1) (0)

Ví dụ 2.13 Tìm loại cao nhất có thể áp dụng cho các văn phạm có các luật sinh như sau:

- (i) $S \rightarrow Aa, A \rightarrow c \mid Ba, B \rightarrow abc$. Luật sinh $B \rightarrow abc$ thuộc loại 2 và $A \rightarrow c, S \rightarrow Aa, A \rightarrow Ba$ là loại 3, vậy văn phạm chứa các luật sinh này là loại 2.
- (ii) $S \rightarrow ASB \mid d, A \rightarrow aA$. Luật sinh $S \rightarrow ASB$ là loại 2, hai luật sinh còn lại là loại 3 nên văn phạm có những luật sinh trên là loại 2.
- (iii) $S \rightarrow aA \mid bB, A \rightarrow bB \mid aS, B \rightarrow bB \mid bS, S \rightarrow a, B \rightarrow a \mid b$. Để kiểm tra tất cả các luật sinh đều loại 3 và do vậy văn phạm chứa chúng là văn phạm chính qui.

Ví dụ 2.14 Cho văn phạm $G = (\{a, b\}, \{S, A, B\}, P, S)$ và P:

$S \rightarrow aB$	$A \rightarrow bAA$
$S \rightarrow bA$	$B \rightarrow b$
$A \rightarrow a$	$B \rightarrow bS$
$A \rightarrow aS$	$B \rightarrow aBB$

Văn phạm này là phi ngữ cảnh (loại 2). Ngôn ngữ sinh bởi văn phạm này gồm tất cả các xâu có số các chữ a bằng số các chữ b. Chúng ta có thể chứng minh điều khẳng định trên bằng phương pháp qui nạp theo độ dài của xâu. Ngoài ra bằng phương pháp qui nạp chúng ta có thể chứng minh các tính chất sau:

Với $w \in \Sigma^*$.

1. $S \xRightarrow{*} w$ nếu và chỉ nếu w có số các chữ a bằng số các chữ b,
2. $A \xRightarrow{*} w$ nếu và chỉ nếu w có số các chữ a lớn hơn số các chữ b là 1,
3. $B \xRightarrow{*} w$ nếu và chỉ nếu w có số các chữ b lớn hơn số các chữ a là 1.

Chúng ta chứng minh tính chất 1.

Bước cơ sở: $|w| = 1$ thì chúng ta chỉ có $A \rightarrow a$ hoặc $B \rightarrow b$. Do vậy từ S không thể dẫn xuất ra xâu nào có 1 ký hiệu, nghĩa là tính chất trên là đúng.

Giả thiết qui nạp: Giả sử tính chất trên đúng với w và $|w| < k$. Chúng ta cần chỉ ra rằng chúng cũng đúng với w với $|w| = k$.

Trước tiên, nếu $S \Rightarrow^* w$ thì dãy dẫn xuất này phải bắt đầu bằng luật sinh $S \rightarrow aB$ hoặc $S \rightarrow bA$. Trường hợp sử dụng $S \rightarrow aB$ thì $w = aw_1$, với $|w_1| < k$ và $B \Rightarrow^* w_1$.

Theo giả thiết qui nạp thì w_1 có số các chữ b lớn hơn số các chữ a là 1. Từ đó suy ra w có số các chữ a bằng số các chữ b . Tương tự đối với luật sinh thứ hai.

Chúng ta chứng minh điều kiện đủ (“chỉ nếu”): w có số các chữ a bằng số các chữ b và $|w| = k$ thì $S \Rightarrow^* w$. Biết rằng trong w chỉ có a hoặc b , do vậy ký hiệu đầu có thể là a hoặc b .

+ Nếu ký hiệu đầu là a thì chúng ta có $w = aw_1$ với $|w_1| = k - 1 < k$ và số ký hiệu b trong w_1 lớn hơn số ký hiệu a là 1. Theo giả thiết qui nạp thì $B \Rightarrow^* w_1$, từ đó suy ra $S \rightarrow aB \Rightarrow^* aw_1 = w$, nghĩa là $S \Rightarrow^* w$.

+ Tương tự đối với trường hợp w bắt đầu bằng b .

Tính chất 2. và 3. có thể chứng minh qui nạp tương tự như trên.

Ví dụ 2.15 Hãy lập văn phạm phi ngữ cảnh sinh các ngôn ngữ trên bảng chữ cái $\Sigma = \{a, b, c, d\}$ thỏa mãn:

$$L = \{x \in \Sigma^* \mid x \text{ có độ dài chẵn, bắt đầu bằng } abc, \text{ kết thúc bởi } cd \text{ và chứa đúng 3 vị trí của } c\}.$$

Giải:

Xây dựng văn phạm phi ngữ cảnh sinh ra L .

Xét $x \in L$, x có độ dài chẵn và phải bắt đầu bằng abc , kết thúc bởi cd . Vậy phần đầu và phần cuối đã chứa 2 chữ c và có tổng cộng 5 chữ cái. Vậy:

$$x \in L \Leftrightarrow x = abcwdc, w \in \Sigma^* \text{ với } w \text{ có độ dài là số lẻ và chứa đúng một chữ } c.$$

$$\Leftrightarrow x = abcw_1cw_2dc, \text{ với } w_1, w_2 \in (\Sigma - \{c\})^* = \{a, b, d\}^* \text{ sao cho } |w_1| + |w_2| \text{ là số chẵn.}$$

Dựa trên cấu trúc của từ x ta thiết lập được văn phạm G sinh ra L có dạng sau:

$G = (\{a, b, c, d\}, \{S, A, B, C, D\}, P, S)$, trong đó P có các luật sinh sau:

$P = \{S \rightarrow abcAcAcd \mid abcBcBcd,$

$A \rightarrow aC \mid bC \mid dC,$

$C \rightarrow aA \mid bA \mid dA \mid \Lambda,$

$B \rightarrow aD \mid bD \mid dD \mid \Lambda,$

$D \rightarrow aB \mid bB \mid dB\}$

Ví dụ 2.16 Hãy lập văn phạm phi ngữ cảnh sinh ra ngôn ngữ $L = \{a^{n+1}b^{n+2} \mid n > 0\}$.

Giải:

Khi xây dựng văn phạm sinh ra một ngôn ngữ cho trước, cần phải thực hiện theo hai bước: **Phân tích cấu trúc các từ của ngôn ngữ và sau đó tìm các qui tắc để sinh ra các từ đó.**

(i) *Phân tích:* mỗi từ $x = a^{n+1}b^{n+2}$ có thể viết thành hai phần:

- + Các từ con a, b^2 không phụ thuộc vào tham số n được gọi là *phần cố định*. Phần này có thể đặt ở các vị trí có thể trên cơ sở đảm bảo luật giao hoán. Với mỗi cách đặt sẽ tạo ra một văn phạm tương đương.
- + Phần phụ thuộc vào n : a^n, b^n . Khi n thay đổi thì số chữ a, b cũng phải thay đổi nhưng phải đảm bảo sinh ra bằng nhau. Ta gọi cụm này là *cụm đối xứng*. Hiển nhiên (a^n, b^n) lập thành cụm đối xứng. Để đảm bảo tính đối xứng thì mỗi cụm đối xứng phải do một biến riêng sinh ra chúng.

Ta có thể viết $w \in L$ bất kỳ dưới một trong bốn dạng khác nhau:

$$w = a^n ab^2 b^n = aa^n b^n b^2 = a^n ab^n b^2 = aa^n b^2 b^n$$

(ii) *Xây dựng văn phạm:* Từ bốn dạng khác nhau của xâu w có bốn văn phạm tương ứng như sau:

$$G_1 = (\{a, b\}, \{S, A\}, \{S \rightarrow aSb \mid aAb, A \rightarrow ab^2\}, S)$$

$$G_2 = (\{a, b\}, \{S, A\}, \{S \rightarrow aAb^2, A \rightarrow aAb \mid ab\}, S)$$

$$G_3 = (\{a, b\}, \{S, A\}, \{S \rightarrow Ab^2, A \rightarrow aAb \mid aab\}, S)$$

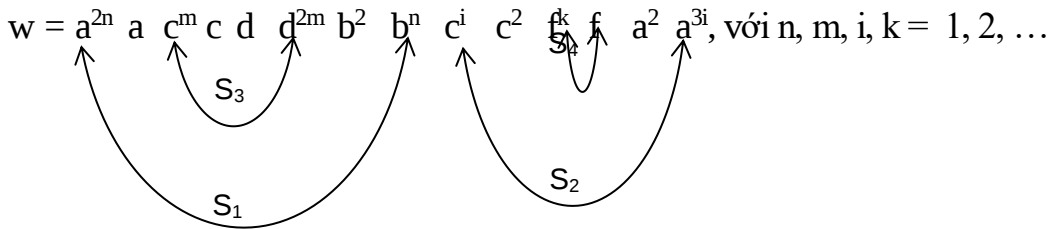
$$G_4 = (\{a, b\}, \{S, A\}, \{S \rightarrow aA, A \rightarrow aAb \mid ab^2b\}, S)$$

Hiển nhiên bốn văn phạm này tương đương với nhau: $L(G_1) = L(G_2) = L(G_3) = L(G_4) = L$.

Ví dụ 2.17 Xây dựng văn phạm phi ngữ cảnh sinh ra ngôn ngữ sau

$$L = \{a^{2n+1}c^{m+1}d^{2m+1}b^{n+2}c^{i+2}f^{k+1}a^{3i+2} \mid n, m, i, k = 0, 1, 2, \dots\}$$

Mỗi từ $w \in L$ đều phân tích được thành 4 cụm đối xứng như sau



Hai cụm đối xứng (a^{2n}, b^n) và (c^i, a^{3i}) có quan hệ tích ghép, nên chúng phải được sinh đồng thời. Ta chọn S_1 để sinh ra (a^{2n}, b^n) và S_2 để sinh ra (c^i, a^{3i}) cùng một lúc. Cụm (c^m, d^{2m}) được lồng trong (a^{2n}, b^n) , cụm (f^k) lồng trong (c^i, a^{3i}) . Những cụm đối xứng có quan hệ lồng nhau thì được sinh dẫn từ ngoài vào trong, nghĩa cụm bị chứa sinh sau cụm chứa nó. Ta chọn tiếp S_3 để sinh ra (c^m, d^{2m}) , S_4 để sinh ra (f^k) . Khi đó cụm (a^{2n}, b^n) phải được sinh ra trước sau mới đến cụm (c^m, d^{2m}) , tức là S_1 phải xuất hiện trước S_3 . Tương tự, cụm (c^i, a^{3i}) phải được sinh ra trước (f^k) , tức là S_2 phải xuất hiện trước S_4 .

Ngoài ra, các ký hiệu phụ S_1, S_2, S_3, S_4 đã đảm nhận việc sinh ra các xâu con của w , nên chúng không thể đóng vai trò là ký tự bắt đầu được. Bởi vậy ta đưa thêm ký tự bắt đầu S và văn phạm sinh ra L sẽ có dạng sau:

$G = (\{a, b, c, d, f\}, \{S, S_1, S_2, S_3, S_4\}, P, S)$, trong đó P gồm các luật sinh sau:

$$S \rightarrow S_1 S_2 \text{ (sinh ra 2 cụm đối xứng tích ghép } (a^{2n}, b^n) \text{ và } (c^i, a^{3i}))$$

$$S_1 \rightarrow a^2 S_1 b \text{ (lặp để sinh ra cụm đối xứng } (a^{2n}, b^n))$$

$$S_1 \rightarrow a S_2 b^2 \text{ (dẫn ra } S_2 \text{ chuẩn bị để sinh ra cụm đối xứng } (a^m, b^{2m}))$$

$$S_2 \rightarrow c S_2 a^2 \text{ (lặp để sinh ra cụm đối xứng } (c^m, b^{2m}))$$

$$S_2 \rightarrow cd \text{ (kết thúc việc sinh ra phần đầu của xâu } w)$$

$$S_3 \rightarrow c S_3 a^3 \text{ (lặp để sinh ra cụm đối xứng } (c^i, a^{3i}))$$

$$S_3 \rightarrow c^2 S_4 a^2 \text{ (đưa } S_4 \text{ vào chuẩn bị để sinh ra } (f^k))$$

$$S_4 \rightarrow f S_4 \mid f$$

Ví dụ 2.18 Cho bảng chữ cái $\Sigma = \{a_1, a_2, \dots, a_n\}$.

(i) *Ngôn ngữ* $L = \Sigma^+$ được sinh bởi văn phạm chính qui

$$G = (\Sigma, \{S\}, \{S \rightarrow a_i S \mid a_i, 1 \leq i \leq n\}, S).$$

(ii) *Ngôn ngữ* L gồm tất cả các từ có độ dài chẵn được sinh bởi văn phạm chính qui suy rộng như sau:

$$G = (\Sigma, \{S, A\}, \{S \rightarrow a_i A \mid \Lambda, A \rightarrow a_j S, 1 \leq i, j \leq n\}, S).$$

(iii) *Ngôn ngữ* L gồm tất cả các từ có độ dài lẻ được sinh bởi văn phạm chính qui suy rộng như sau:

$$G = (\Sigma, \{S, A\}, \{S \rightarrow a_i A, A \rightarrow a_j S \mid \Lambda, 1 \leq i, j \leq n\}, S).$$

2.4 Tính đệ qui và các tập đệ qui

Khái niệm tập đệ qui được sử dụng trong nhiều lĩnh vực khác nhau. Để định nghĩa tập đệ qui chúng ta cần nhắc lại khái niệm thủ tục và thuật toán.

Một thủ tục để giải một bài toán là một dãy các câu lệnh xử lý các dữ liệu vào (dãy vào) để cho ra các kết quả (output) mong muốn. Tập dữ liệu vào có thể là tập rỗng.

Một thuật toán là một thủ tục được thực hiện và kết thúc sau hữu hạn các bước đối với mọi dữ liệu vào.

Định nghĩa. Tập X là *đệ qui* nếu có một *thuật toán* kiểm tra được một phần tử có thuộc X hay không.

Định nghĩa. Tập X là *được đánh số đệ qui* nếu có một *thủ tục* kiểm tra được một phần tử có thuộc X hay không.

Hiển nhiên là *tập đệ qui sẽ được đánh số đệ qui* (còn gọi là đệ qui đếm được).

Định lý. Ngôn ngữ cảm ngữ cảnh là đệ qui.

Chứng minh: Cho $G = (\Sigma, V, P, S)$ là ngôn ngữ cảm ngữ cảnh và $w \in \Sigma^*$. Chúng ta cần xây dựng thuật toán kiểm tra xem $w \in L(G)$ hay không.

Nếu $w = \Lambda$, $w \in L(G) \Leftrightarrow S \rightarrow \Lambda$ có trong P . Việc kiểm tra xem $S \rightarrow \Lambda$ có trong P luôn thực hiện được sau hữu hạn lần vì lực lượng của P là hữu hạn.

Giả sử $|w| = n \geq 1$. Thuật toán sẽ được thiết lập dựa trên việc xây dựng dãy các tập con $\{W_i\}$ của $V \cup \Sigma$. W_i đơn giản là tập tất cả các xâu dẫn xuất ra ở bước thứ i và có độ dài nhỏ hơn hoặc bằng n . Dãy này được xây dựng đệ qui như sau:

$$(i) \quad W_0 = \{S\},$$

$$(ii) \quad W_{i+1} = W_i \cup \{\beta \in (V \cup \Sigma)^* \mid \exists \alpha \in W_i \text{ sao cho } \alpha \Rightarrow \beta \text{ và } |\beta| \leq n\},$$

Các tập W_i thỏa mãn các tính chất sau:

$$(iii) \quad W_i \subseteq W_{i+1}, \text{ với mọi } i \geq 0,$$

$$(iv) \quad \exists k \text{ sao cho } W_k = W_{k+1},$$

$$(v) \quad \text{Nếu } k \text{ là số nguyên nhỏ nhất để } W_k = W_{k+1} \text{ thì}$$

$$W_k = \{\alpha \in (V \cup \Sigma)^* \mid S \overset{*}{\Rightarrow} \alpha \text{ và } |\alpha| \leq n\}$$

Tính chất (iii) suy ra từ (ii). Để chứng minh (iv), chúng ta xét N là số tất cả các xâu trên $V \cup \Sigma$ có độ dài nhỏ hơn hoặc bằng n . Nếu $|V \cup \Sigma| = m$ thì $N = 1 + m + m^2 + \dots + m^n$ bởi vì m^i là số các xâu có độ dài i trên $V \cup \Sigma$. N được xác định và nó chỉ phụ thuộc vào n và m . Hiển nhiên là $|W_i| \leq N$, với mọi i . Từ đó suy ra (dựa vào nguyên lý chuồng bồ câu) $W_k = W_{k+1}$, với $k \leq N$.

Từ (ii) và từ (iv) $W_k = W_{k+1}$ kéo theo $W_{k+1} = W_{k+2}$. Vậy chúng ta chứng minh được tính chất (v) vì

$$\{\alpha \in (V \cup \Sigma)^* \mid S \overset{*}{\Rightarrow} \alpha \text{ và } |\alpha| \leq n\} = W_1 \cup W_2 \cup \dots \cup W_k \cup W_{k+1} \dots$$

$$= W_1 \cup W_2 \cup \dots \cup W_k = W_k \text{ (từ (ii)).}$$

Tính chất (v) khẳng định $w \in L(G)$ ($S \overset{*}{\Rightarrow} w$) khi và chỉ khi $w \in W_k$. Chúng ta đã biết là $W_1, W_2, \dots, W_k$ được tạo ra sau hữu hạn bước.

Thuật toán kiểm tra $w \in L(G)$ được thực hiện như sau:

1. Xây dựng $W_1, W_2, \dots, W_k$ như trong (i) và (ii) và $W_k = W_{k+1}$.
2. Nếu $w \in W_k$ thì $w \in L(G)$, ngược lại $w \notin L(G)$. ■

Ví dụ 2.23 Cho văn phạm G có $S \rightarrow 0SA_12$, $S \rightarrow 012$, $2A_1 \rightarrow A_12$, $1A_1 \rightarrow 11$. Kiểm tra

a/ $00112 \in L(G)$?. Để kiểm tra $w = 00112 \in L(G)$ hay không, chúng ta hãy xây dựng $W_1, W_2, \dots, W_k$ biết rằng $|w| = 5$.

$$W_0 = \{S\},$$

$$W_1 = \{012, S, 0SA_12\},$$

$$W_2 = \{012, S, 0SA_12\},$$

$W_1 = W_2$ và như vậy $k = 2$ (mặc dù $0SA_12 \Rightarrow 0012A_12$, nhưng không thể bổ sung $0012A_12$ vào W_2 vì nó có độ dài lớn hơn 5). 00112 không nằm trong W_2 vậy nó không được sinh bởi văn phạm trên.

b/ $w = 001122 \in L(G)$?. Vì $|w| = 6$ nên chúng ta có

$$W_0 = \{S\},$$

$$W_1 = \{012, S, 0SA_12\},$$

$$W_2 = \{012, S, 0SA_12, 0012A_12\},$$

$$W_3 = \{012, S, 0SA_12, 0012A_12, 001A_122\},$$

$$W_4 = \{012, S, 0SA_12, 0012A_12, 001A_122, 001122\},$$

$$W_5 = \{012, S, 0SA_12, 0012A_12, 001A_122, 001122\},$$

Dễ kiểm tra được $001122 \in W_4$, suy ra $w \in L(G)$.

Định lý sau khá thú vị về mặt lý thuyết tập đệ quy và các ngôn ngữ hình thức.

Định lý. Tồn tại tập đệ quy nhưng không phải là ngôn ngữ cảm ngữ cảnh trên tập $\{0,1\}$.

Chứng minh:

Đặt $\Sigma = \{0, 1\}$. Các phân tử (xâu) của Σ^* được viết thành dãy w_i , $i = 1, 2, \dots$ và được đánh số: $\Lambda, 0, 1, 00, 01, 11, 000, \dots$ Ví dụ $w_{10} = 010$ là phân tử thứ 10 của dãy.

Biết rằng mọi văn phạm đều có tập các ký hiệu (chưa kết thúc và kết thúc), tập các qui tắc là hữu hạn, nên tất cả các văn phạm cảm ngữ cảnh xác định trên Σ có thể viết thành dãy $G_1, G_2, \dots$ sao cho G_i có chứa w_i .

Chúng ta định nghĩa $X = \{w_i \in \Sigma^* \mid w_i \notin L(G_i)\}$ và chứng minh rằng X là đệ qui. Nếu $w \in \Sigma^*$ thì chúng ta có thể tìm được i để $w = w_i$. Quá trình tìm i luôn kết thúc sau hữu hạn bước kiểm tra (số bước phụ thuộc vào $|w|$). Ví dụ, $w = 0100$ thì $w = w_{20}$, $i = 20$. Vì G_{20} là văn phạm cảm ngữ cảnh, do vậy chúng ta (có thuật toán) kiểm tra được xem $w \in L(G_{20})$ hay không (theo định lý 3.3), suy ra X là đệ qui.

Vấn đề còn lại là chứng minh X không phải là ngôn ngữ cảm ngữ cảnh. Chúng ta chứng minh bằng phản chứng. Giả thiết tồn tại n sao cho $X = L(G_n)$. Xét phần tử thứ n của Σ^* (tức w_n). Theo định nghĩa của X , $w_n \in X$ kéo theo $w_n \notin L(G_n)$. Điều này mâu thuẫn với $X = L(G_n)$. Ngược lại, nếu $w_n \notin X$ sẽ kéo theo $w_n \in L(G_n)$, lại mâu thuẫn với $X = L(G_n)$. Do đó $X \neq L(G_n)$ với mọi n và X không được sinh bởi văn phạm cảm ngữ cảnh nào cả. ■

2.5 Các phép toán trên các ngôn ngữ

Ngoài những tính chất của các lớp ngôn ngữ như đã khảo sát ở phần trước, nhiều tính chất khác như khả năng thực hiện các phép toán tập hợp trên các ngôn ngữ (xem như tập các xâu, hay các từ) cũng rất quan trọng và có nhiều ứng dụng trong việc nghiên cứu hành vi của các hệ thống, trong đó có lý thuyết chương trình dịch.

Giả sử A và B là hai tập các xâu bất kỳ. Ghép của hai tập A, B là $AB = \{uw \mid u \in A \text{ và } w \in B\}$ (uw là ghép hai xâu như ở chương 1 đã định nghĩa). Cho trước tập A các xâu,

$$A^0 = \Lambda \text{ (từ rỗng)}$$

$$A^1 = A,$$

$$A^{n+1} = A^n A = A A^n, \text{ với mọi } n \geq 1 \text{ và}$$

$$A^T = \{u^T \mid u \in A\} - \text{tập các từ đảo vị (soi gương) của } A.$$

Giả sử L là ngôn ngữ trên Σ , phần bù của L là ngôn ngữ, ký hiệu là $L^c = \Sigma^* - L$.

Định nghĩa. Ta nói ngôn ngữ $\mathcal{L}_i$ đóng với phép toán $\oplus$ nếu

$$\forall L_1, L_2 \in \mathcal{L}_i \text{ thì } L_1 \oplus L_2 \in \mathcal{L}_i, i = 0, 1, 2, 3.$$

Định lý. Các lớp ngôn ngữ $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$ đóng đối với phép hợp $\cup$.

Chứng minh. Giả sử $L_1, L_2 \in \mathcal{L}_i, i = 0, 1, 2, 3$. áp dụng định lý 3.1 chúng ta xây dựng được hai văn phạm $G_1 = (\Sigma_1, V^1, P_1, S_1), G_2 = (\Sigma_2, V^2, P_2, S_2)$ thuộc loại i để sinh ra L_1, L_2 tương ứng. Các qui tắc trong P_1, P_2 có dạng $\alpha \rightarrow \beta$, với α và β chỉ chứa các biến hoặc có dạng $A \rightarrow a$, với A là biến còn a là ký hiệu kết thúc.

Không mất tính tổng quát, chúng ta có thể giả thiết $V^1 \cap V^2 = \emptyset$ (bằng cách đặt tên các biến khác nếu chúng xuất hiện trong cả hai tập).

Chúng ta định nghĩa văn phạm mới

$$G = (V^1 \cup V^2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, P, S), \text{ trong đó}$$

S là ký hiệu mới không thuộc $V^1 \cup V^2$ và

$$P = P_1 \cup P_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\} \text{ với } i = 0, 1, 2$$

Riêng đối với trường hợp các ngôn ngữ chính qui ($i = 3$) thì

$$P = P_1 \cup P_2 \cup \{S \rightarrow a \mid bA, S \rightarrow c \mid dB\} \text{ nếu có } S_1 \rightarrow a \mid bA \text{ và } S_2 \rightarrow c \mid dA$$

$$P = P_1 \cup P_2 \cup \{S \rightarrow aS, S \rightarrow bS\} \text{ nếu có } S_1 \rightarrow aS_1 \text{ và } S_2 \rightarrow bS_2$$

Chứng minh $L(G) = L_1 \cup L_2$ có thể tiến hành như sau: Nếu $w \in L_1 \cup L_2$ thì $S_1 \xRightarrow{*}_{G_1} w$ hoặc $S_2 \xRightarrow{*}_{G_2} w$. Vì có $S \rightarrow S_1$ và $S \rightarrow S_2$, nên $S \xRightarrow{G} S_1 \xRightarrow{*}_{G_1} w$ hoặc $S \xRightarrow{G} S_2 \xRightarrow{*}_{G_2} w$, suy ra $w \in L(G)$, nghĩa là $L_1 \cup L_2 \subseteq L(G)$.

Để chứng minh chiều ngược lại $L(G) \subseteq L_1 \cup L_2$, hãy xét câu $w \in L(G), S \xRightarrow{*}_G w$. Nếu sử dụng qui tắc $S \rightarrow S_1$ thì các bước dẫn xuất sau đó chỉ sử dụng các qui tắc trong G_1 vì các biến có thể xuất hiện ở vế trái để dẫn xuất được của hai văn phạm là rời nhau. Do vậy

$$S \xRightarrow{G} S_1 \xRightarrow{*}_{G_1} w, \text{ suy ra } w \in L_1.$$

Tương tự đối với trường hợp khi sử dụng $S \rightarrow S_2$ sẽ dẫn ra

$$S \xRightarrow{G} S_2 \xRightarrow{*}_{G_2} w \text{ và } w \in L_2.$$

Cuối cùng chúng ta có $L(G) = L_1 \cup L_2$.

Hơn nữa, dễ nhận thấy, nếu G_1 và G_2 là loại 0 hay loại 2 thì G cũng là văn phạm thuộc cùng loại vì tất cả các qui tắc trong P cùng loại với P_1 và P_2 . Nếu $\Lambda \notin L_1 \cup L_2$ và G_1, G_2 cùng loại 1 hoặc 3 thì G cũng sẽ cùng loại đó. Khi $\Lambda \in L_1 \cup L_2$ và G_1, G_2 cùng loại 1 hoặc 3, giả sử $\Lambda \in L_1$ (tương tự đối với $\Lambda \in L_2$) thì chúng ta định nghĩa

$$G = (V^1 \cup V^2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, P, S), \text{ trong đó}$$

S là ký hiệu mới không thuộc $V^1 \cup V^2$ và

$$P = (P_1 - \{S_1 \rightarrow \Lambda\}) \cup (P_2 - \{S_2 \rightarrow \Lambda\}) \cup \{S \rightarrow S_1, S \rightarrow S_2, S \rightarrow \Lambda\}.$$

Hiển nhiên là $\Lambda \in L(G)$ và G cũng sẽ cùng loại với G_1 và G_2 . ■

Định lý. Các lớp ngôn ngữ $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$ đóng đối với phép ghép o

Chứng minh: Giả sử L_1 và L_2 là hai ngôn ngữ loại $i, i = 0, 1, 2, 3$. Giống như trong định lý 3.5 có thể xây dựng hai văn phạm $G_1 = (\Sigma_1, V^1, P_1, S_1), G_2 = (\Sigma_2, V^2, P_2, S_2)$ thuộc loại i để sinh ra L_1, L_2 tương ứng. Chúng ta sẽ chứng minh rằng $L_1 L_2$ cũng là ngôn ngữ loại i .

Văn phạm mới được định nghĩa như sau:

$$G_C = (\Sigma_1 \cup \Sigma_2, V^1 \cup V^2 \cup \{S\}, P_C, S), \text{ trong đó}$$

S là ký hiệu mới không thuộc $V^1 \cup V^2$ và

$$P_C = P_1 \cup P_2 \cup \{S \rightarrow S_1 S_2\} \text{ với } i = 0, 1, 2$$

Riêng đối với trường hợp chính qui ($i = 3$) thì

$$P_C = P_1 \cup P_2 \cup \{S \rightarrow aS_1 \mid aS_2\} \text{ nếu có } S_1 \rightarrow aS_1 \mid a$$

Trước tiên cần chứng minh $L_1 L_2 = L(G_C)$. Nếu $w = w_1 w_2 \in L_1 L_2$ thì $S_1 \xRightarrow[G_1]{*} w_1$ và $S_2 \xRightarrow[G_2]{*} w_2$. Vì $S \rightarrow S_1 S_2$ có trong P_C , nên $S \xRightarrow[G_C]{*} S_1 S_2 \xRightarrow[G_C]{*} w_1 w_2 = w$, suy ra $w \in L(G_C)$. Do đó $L_1 L_2 \subseteq L(G)$.

Để chứng minh chiều ngược lại, hãy xét $w \in L(G_C)$, $S \xRightarrow{G_C} S_1 S_2 \xRightarrow{G_C}^* w$. Vì $V_N^1 \cap V_N^2 = \emptyset$ nên $w = w_1 w_2$ và $S_1 \xRightarrow{G_1}^* w_1$ và $S_2 \xRightarrow{G_2}^* w_2$. Từ đó suy ra $L_1 L_2 = L(G_C)$.

Vấn đề còn lại là chứng minh G_C cùng loại i với G_1 và G_2 .

Nếu G_1 và G_2 cùng loại 0 hoặc 2 thì tất nhiên G_C cũng cùng loại đó.

Trường hợp $\Lambda \notin L_1 \cup L_2$ và G_1, G_2 cùng loại 1, 3 thì hiển nhiên G_C cũng sẽ cùng loại giống như G_1 và G_2 .

Xét trường hợp $\Lambda \in L_1$ hoặc $\Lambda \in L_2$ và G_1, G_2 cùng loại 1 hoặc 3. Đặt $L_1' = L_1 - \{\Lambda\}$, $L_2' = L_2 - \{\Lambda\}$. Khi đó

$$L_1 L_2 = \begin{cases} L_1' L_2' \cup L_2' & \text{nếu } \Lambda \in L_1 \text{ và } \Lambda \notin L_2 \\ L_1' L_2' \cup L_1' & \text{nếu } \Lambda \notin L_1 \text{ và } \Lambda \in L_2 \\ L_1' L_2' \cup L_1' \cup L_2' \cup \{\Lambda\} & \text{nếu } \Lambda \in L_1 \text{ và } \Lambda \in L_2 \end{cases}$$

Như đã chứng minh ở định lý trước, các lớp ngôn ngữ đều đóng với phép hợp, nên $L_1 L_2$ sẽ là loại 1 hay 3 tùy thuộc L_1 và L_2 là loại 1 hay 3 tương ứng. ■

Chú ý:

- Phép ghép các ngôn ngữ không giao hoán, nghĩa là $L_1 L_2 \neq L_2 L_1$.
- Phép ghép có tính kết hợp với phép hợp và phép giao

$$L_1(L_2 \cup L_3) = L_1 L_2 \cup L_1 L_3$$

$$(L_1 \cup L_2)L_3 = L_1 L_3 \cup L_2 L_3$$

$$L_1(L_2 \cap L_3) = L_1 L_2 \cap L_1 L_3$$

$$(L_1 \cap L_2)L_3 = L_1 L_3 \cap L_2 L_3$$

Định lý. Các lớp ngôn ngữ $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$ đóng đối với phép đảo vị.

Chứng minh. Giả sử $G = (\Sigma, V, P, S)$ là văn phạm loại i , $i = 0, 1, 2, 3$ và $L = L(G)$. Chúng ta cần chỉ ra rằng L^T cũng là loại ngôn ngữ i .

Xây dựng văn phạm $G^T = (\Sigma, V, P^T, S)$, trong đó P^T gồm những qui tắc từ P và đảo ngược lại cả vế trái và vế phải của chúng.

$$P^T = \{\alpha^T \rightarrow \beta^T \mid \alpha \rightarrow \beta \in P\}$$

Như thế nếu G là loại 0, 1, 2 thì kiểm tra ngay được G^T cũng là loại 0, 1, 2. Đối với văn phạm chính qui chúng ta chứng minh ở chương sau. ■

Nhận xét: Từ tính chất đóng của các lớp ngôn ngữ đối với phép đảo vị, ta đảo ngược tuyến tính trái sẽ tạo ra tuyến tính phải và ngược lại.

Riêng đối với phép giao và phép lấy phần bù của các ngôn ngữ thì tính đóng là khó xác định hơn. Để tham khảo ở đây chúng ta chỉ nêu kết quả còn phần chứng minh có thể tham khảo tìm hiểu thêm.

Định lý.

- (i) Các lớp ngôn ngữ $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_3$ đóng đối với phép giao.
- (ii) Lớp $\mathcal{L}_3$ đóng với phép lấy phần bù.
- (iii) Lớp các ngôn ngữ phi ngữ cảnh không đóng với phép giao và phép lấy phần bù.

Tính đóng của $\mathcal{L}_3$ đối với phép giao và phép lấy phần bù sẽ được chứng minh ở chương sau.

Ví dụ 2.24 Giao của hai ngôn ngữ phi ngữ cảnh có thể không phải là phi ngữ cảnh. Xét hai ngôn ngữ:

$$L_1 = \{a^n b^n c^i \mid n, i \geq 1\}$$

$$L_2 = \{a^n b^i c^i \mid n, i \geq 1\}$$

L_1, L_2 là hai ngôn ngữ phi ngữ cảnh vì tồn tại hai văn phạm phi ngữ cảnh

$$G_1 = (\{a, b, c\}, \{S, A\}, \{S \rightarrow A \mid Sc, A \rightarrow ab \mid aAb\}, S) \text{ và } L(G_1) = L_1$$

$$G_2 = (\{a, b, c\}, \{S, A, B\}, \{S \rightarrow AB, A \rightarrow a \mid aA, B \rightarrow bc \mid bBc\}, S) \text{ và } L(G_2) = L_2$$

Tuy nhiên, $L_1 \cap L_2 = \{a^n b^n c^n \mid n \geq 1\}$ không phải là ngôn ngữ phi ngữ cảnh (xem Bổ đề Bơm cho NNPNC).

Ví dụ 2.25 Phần bù của ngôn ngữ phi ngữ cảnh có thể không phải là phi ngữ cảnh. Thật vậy, xét hai ngôn ngữ phi ngữ cảnh L_1, L_2 đã cho ở ví dụ 2.24. Nếu ngôn ngữ phi ngữ cảnh đóng với phép lấy phần bù thì L_1^c, L_2^c sẽ là các ngôn ngữ phi ngữ cảnh. Bởi vì lớp các ngôn ngữ phi ngữ cảnh đóng với phép hợp nên $L_1^c \cup L_2^c$ cũng là phi ngữ cảnh. Tương tự, suy ra $(L_1^c \cup L_2^c)^c$ là ngôn ngữ phi ngữ cảnh.

Mặt khác, theo luật De Morgan's thì $L_1 \cap L_2 = (L_1^c \cup L_2^c)^c$, vậy mâu thuẫn với kết luận nêu trên, vì ở ví dụ 2.24 đã khẳng định $L_1 \cap L_2$ không phải là phi ngữ cảnh.

Bài tập về văn phạm và các ngôn ngữ sinh

2.1 Tìm ngôn ngữ sinh bởi các văn phạm sau:

- (a) $S \rightarrow 0S1 \mid 0A1, A \rightarrow 1A \mid 1$
- (b) $S \rightarrow 0S1 \mid 0A \mid 1B \mid 0 \mid 1, A \rightarrow 0A \mid 0, B \rightarrow 1B \mid 1$
- (c) $S \rightarrow 0A \mid 1S \mid 0 \mid 1, A \rightarrow 1A \mid 1S \mid 1$
- (d) $S \rightarrow 0A \mid 1S \mid 0 \mid 1, A \rightarrow 1A0 \mid 1S \mid 1$

2.2 Xây dựng văn phạm để sinh ra các ngôn ngữ sau:

- (a) Tập tất cả các xâu trên $\{0, 1\}$, trong đó số các chữ số 0 bằng số các chữ số 1.
- (b) $\{0^n 1^m 0^{m1^n} \mid m, n \geq 1\}$
- (c) $\{0^n 1^{2n} \mid n \geq 1\}$
- (d) $\{0^n 1^n \mid n \geq 1\} \cup \{1^m 0^m \mid m \geq 1\}$

2.3 Xây dựng văn phạm G sao cho $L(G) = \{a^n b a^m \mid n, m \geq 1\}$.

2.4 Cho G với luật sinh $S \rightarrow aS \mid a$. Hãy chỉ ra rằng $L(G) = \{a\}^+$.

2.5 Tìm văn phạm để sinh ra ngôn ngữ $L = \{a^k b^n c^n \mid n \geq 1, k \geq 0\}$

2.6 Xây dựng

- (a) Văn phạm cảm ngữ cảnh nhưng không phải là phi ngữ cảnh,
- (b) Văn phạm phi ngữ cảnh nhưng không phải là chính qui,
- (c) Văn phạm chính qui để sinh ra $\{a^n \mid n \geq 1\}$.

2.7 Xây dựng văn phạm phi ngữ cảnh để sinh ra:

- (a) $\{0^m 1^n \mid m \neq n \text{ và } m, n \geq 1\}$
- (b) $\{0^m 1^n \mid 1 \leq m \leq n\}$

(c) $\{a^l b^m c^n \mid l + m = n\}$

2.8 Xây dựng văn phạm chính qui để sinh ra:

(a) $\{a^{2m} \mid m \geq 1\}$

(b) Tập tất cả các xâu trên $\{a, b\}$ kết thúc bằng a

(c) Tập tất cả các xâu trên $\{a, b\}$ bắt đầu bằng a

(d) $\{a^l b^m c^n \mid l, m, n \geq 1\}$

2.9 Chứng minh rằng $G_1 = (\{a, b\}, \{S\}, P_1, S)$, trong đó $P_1 = \{S \rightarrow aSb \mid ab\}$ là tương đương với $G_2 = (\{a, b\}, \{S, A, B, C\}, P_2, S)$, với $P_2 = \{S \rightarrow AC, C \rightarrow SB, S \rightarrow AB, A \rightarrow a, B \rightarrow b\}$.

2.10 Xây dựng văn phạm để sinh ra $\{(ab)^n \mid n \geq 1\} \cup \{(ba)^n \mid n \geq 1\}$.

2.11* Cho ngôn ngữ phi ngữ cảnh $L = \{a^n 0^k b^n c^m d^m \mid n > 0, m > 0, k \geq 0\}$

(i) Hãy tìm văn phạm phi ngữ cảnh G sinh ra L .

(ii) Hãy kiểm tra xem các từ $a^3 0^2 b^3 c^2 d^2$, $a^2 0^2 b^2 c^2 d$ có được sinh ra bởi văn phạm G tìm được trong câu (i)?

2.12 Xây dựng văn phạm để sinh ra

(i) $L_1 = \{(ab)^n \mid n > 0\}$, $L_2 = \{(ba)^n \mid n > 0\}$

(ii) $L = L_1 \cup L_2$

(iii) $L = L_1 L_2$

2.13* Xây dựng văn phạm phi ngữ cảnh sinh ra ngôn ngữ

$$L = \{x \in \{a, b\}^+ \mid \text{số lần xuất hiện của } a \text{ lớn hơn số lần xuất hiện } b \text{ trong } x\}$$

2.14* Xây dựng văn phạm phi ngữ cảnh trên $\Sigma = \{a, b, c, d, e\}$ sinh ra ngôn ngữ

1. $L = \{w \in \Sigma^* \mid w \text{ bắt đầu bằng } abc, \text{ kết thúc bằng } cde, \text{ chứa đúng 3 vị trí của } c \text{ và } w \text{ có độ dài chẵn}\}$
2. $L = \{w \in \Sigma^* \mid w \text{ bắt đầu bằng } abc, \text{ kết thúc bằng } cde, \text{ chứa đúng 3 vị trí của } b \text{ và } w \text{ có độ dài lẻ}\}$
3. $L = \{w \in \Sigma^* \mid w \text{ không chứa từ } bcd\}$.

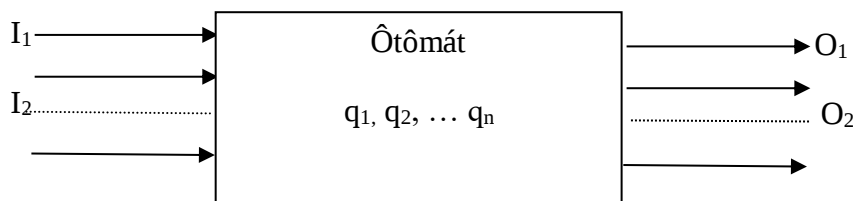
CHƯƠNG 3. LÝ THUYẾT ÔTÔMAT

Chương 3 giới thiệu những khái niệm cơ sở và các tính chất của lý thuyết ôtômát:

- ✓ Các khái niệm cơ bản và các tính chất của ôtômát,
- ✓ Các tính chất của hàm chuyển đổi trạng thái,
- ✓ Sự tương đương của ôtômát hữu hạn đơn định và không đơn định,
- ✓ Một số vấn đề liên quan đến cực tiểu hoá ôtômát hữu hạn.

3.1 Ôtômát hữu hạn

Trong khoa học máy tính, thuật ngữ ôtômát được hiểu là máy xử lý tự động trên dữ liệu rời rạc. Mỗi ôtômát được xem như là một cơ chế biến đổi thông tin gồm một bộ điều khiển, một kênh vào và một kênh ra.



Hình H3-1 Mô hình của ôtômát rời rạc

Trong đó có các thành phần:

1. *Đầu vào (input)*. Ở mỗi thời khoảng (rời rạc) $t_1, t_2, \dots$, các dữ liệu vào $I_1, I_2, \dots$, là những số hữu hạn các giá trị từ bảng chữ cái Σ_i đầu vào.
2. *Đầu ra (Output)*. $O_1, O_2, \dots, O_m$: các kết quả xử lý của hệ thống
3. *Trạng thái*. Tại mỗi thời điểm, ôtômát có thể ở một trong các trạng thái $q_1, q_2, \dots, q_n$.
4. *Quan hệ giữa các trạng thái*. Trạng thái tiếp theo của ôtômát phụ thuộc vào hiện trạng và đầu vào hiện thời, nghĩa là được xác định phụ thuộc vào một hay nhiều trạng thái trước và dữ liệu hiện thời.
5. *Quan hệ kết quả*. Kết quả của ôtômát không những chỉ phụ thuộc vào các hiện trạng mà còn phụ thuộc vào các đầu vào. Như vậy, kết quả (đầu ra,

Ôtômát và ngôn ngữ hình thức

output) của ôtômát được xác định theo đầu vào và các trạng thái thực hiện của nó.

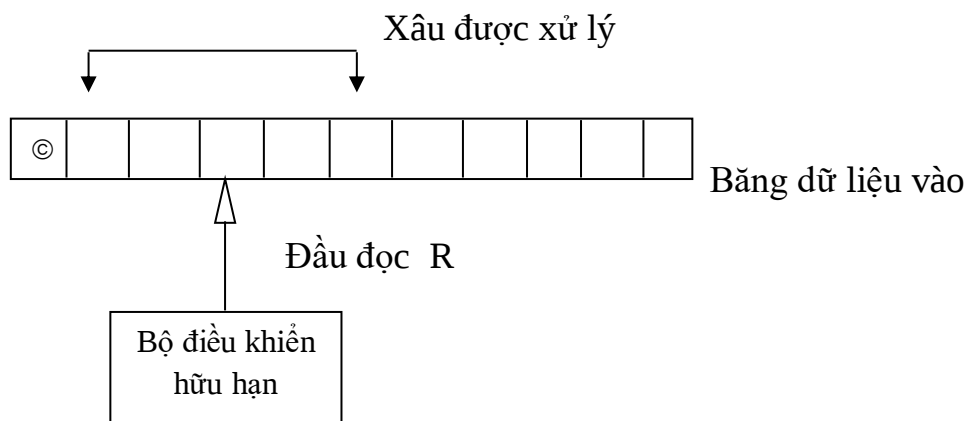
Chúng ta xét định nghĩa hình thức về ôtômát hữu hạn trạng thái gọi tắt là ôtômát hữu hạn.

Một cách hình thức, ôtômát hữu hạn được định nghĩa như sau.

Định nghĩa. Ôtômát hữu hạn đơn định (gọi tắt là ôtômát hữu hạn), ký hiệu là ÔTĐĐ, có thể biểu diễn được bởi bộ 5 thành phần $M = (Q, \Sigma, \sigma, q_0, F)$, trong đó:

- (i) Q là tập hữu hạn khác rỗng các trạng thái,
- (ii) Σ là tập hữu hạn khác rỗng các ký hiệu đầu vào (bảng chữ cái), với giả thiết $Q \cap \Sigma = \emptyset$,
- (iii) σ là ánh xạ từ $Q \times \Sigma$ vào Q và được gọi là *hàm chuyển trạng thái*. Hàm này mô tả sự thay đổi trạng thái của ôtômát và thường được cho biết dưới dạng bảng chuyển trạng thái hay đồ thị chuyển trạng thái.
- (iv) $q_0 \in Q$ là *trạng thái khởi đầu*,
- (v) $F \subseteq Q$ là tập các *trạng thái kết thúc*. Tổng quát: $|F| \geq 1$.

Mô hình trên có thể được mô tả như trong hình H3- 2.



Hình H3-2 Sơ đồ khối của ôtômát hữu hạn

Ôtômát hữu hạn trạng thái gồm các thành phần sau:

- (i) *Băng dữ liệu vào*. Băng dữ liệu vào được chia thành các ô, mỗi ô chứa một ký tự từ bảng chữ cái Σ , trong đó ô đầu được đánh dấu cho sự bắt đầu bằng © và ô cuối dùng \$ để đánh dấu kết thúc. Khi không sử dụng

các ô đánh dấu đó thì băng dữ liệu vào sẽ được xem như có độ dài vô hạn.

(ii) *Đầu đọc R*. Mỗi lần đầu đọc chỉ xem xét một ô và có thể dịch qua phải hoặc qua trái một ô. Để đơn giản, chúng ta giả thiết đầu đọc chỉ dịch qua phải sau mỗi lần xử lý.

(iii) *Bộ điều khiển hữu hạn*. Bộ này điều khiển hoạt động của ôtômát mỗi khi đọc một dữ liệu vào. Khi đọc một ký hiệu vào, ví dụ a , ở trạng thái q có thể cho các kết quả sau:

(a) Đầu đọc R vẫn ở nguyên tại chỗ, không dịch chuyển,

(b) Chuyển sang trạng thái khác được xác định theo $\sigma(q, a)$.

Để hiểu được hoạt động của các ôtômát, chúng ta cần phải tìm hiểu các tính chất, đặc trưng của hàm chuyển trạng thái.

3.1.1 Các tính chất của hàm chuyển trạng thái

Hàm chuyển trạng thái là một ánh xạ xác định trên các cặp: (trạng thái, ký hiệu vào), do vậy không thể sử dụng trực tiếp để đoán nhận các xâu, mà phải mở rộng σ thành ánh xạ

$$\sigma: Q \times \Sigma^* \rightarrow Q$$

nhờ các tính chất sau.

Tính chất.

(i) $\forall q \in Q, \sigma(q, \Lambda) = q$, với Λ là từ rỗng. Nghĩa là ôtômát chỉ thay đổi trạng thái khi có dữ liệu vào. (3.1)

(ii) $\forall w \in \Sigma^*, a \in \Sigma, \sigma(q, aw) = \sigma(\sigma(q, a), w)$ (3.2a)

$\sigma(q, wa) = \sigma(\sigma(q, w), a)$ (3.2b)

Mệnh đề. Với mọi hàm chuyển trạng thái σ và với mọi xâu vào x, y ,

$$\sigma(q, xy) = \sigma(\sigma(q, x), y) \quad (3.3)$$

Chứng minh: Qui nạp theo $|y|$, nghĩa là theo số ký tự trong y .

Cơ sở: Khi $|y| = 1, y = a$, thì $\sigma(q, xa) = \sigma(\sigma(q, x), a)$ đúng theo tính chất (3.2b).

Giả sử rằng (3.3) đúng với mọi xâu x và với những xâu y có n phân tử, $|y| = n$.

Khi xâu y có độ dài là $n+1$, ta có thể viết $y = y_1a$, với $|y_1| = n$. Khi đó

$$\begin{aligned}
 \sigma(q, xy) &= \sigma(q, xy_1a) = \sigma(q, x_1a), \text{ với } x_1 = xy_1 \\
 &= \sigma(\sigma(q, x_1), a), && \text{Theo (3.2b)} \\
 &= \sigma(\sigma(q, xy_1), a) \\
 &= \sigma(\sigma(\sigma(q, x), y_1), a), && \text{Theo giả thiết qui nạp} \\
 &= \sigma(\sigma(q, x), y_1a), && \text{Theo (3.2b)} \\
 &= \sigma(\sigma(q, x), y).
 \end{aligned}$$

Do vậy, tính chất (3.3) đúng với mọi xâu x, y . ■

3.1.2 Các phương pháp biểu diễn ôtômát

Cho trước một ôtômát, thực chất là chỉ cần cho biết hàm chuyển trạng thái của nó. Hàm chuyển trạng thái có thể được cho dưới 2 dạng.

Phương pháp biểu diễn bằng bảng chuyển trạng thái

Một ôtômát có thể được cho dưới dạng bảng chuyển trạng thái

Các trạng thái	Ký hiệu vào			
	a_1	a_2	$\dots$	a_n
$q_0 \rightarrow$	$\sigma(q_0, a_1)$	$\sigma(q_0, a_2)$		$\sigma(q_0, a_n)$
q_1	$\sigma(q_1, a_1)$	$\sigma(q_1, a_2)$		$\sigma(q_1, a_n)$
$\dots$	$\dots$			
$q_f \bigcirc$	$\sigma(q_f, a_1)$	$\sigma(q_f, a_2)$		$\sigma(q_f, a_n)$
$\dots$	$\dots$			
q_k	$\sigma(q_k, a_1)$	$\sigma(q_k, a_2)$		$\sigma(q_k, a_n)$

trong đó, $Q = \{q_1, q_2, \dots, q_k\}$, $\Sigma = \{a_1, a_2, \dots, a_n\}$ và F là tập tất cả những trạng thái q_f (những trạng thái nằm trong hình tròn), q_0 là trạng thái bắt đầu

Ôtômát và ngôn ngữ hình thức

Ví dụ 3.2 Xét ôtômát hữu hạn có hàm chuyển trạng thái được xác định theo bảng sau

Bảng B3.1 Bảng chuyển trạng thái của M

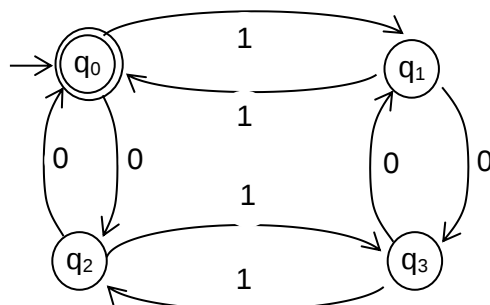
Trạng thái	Dãy vào	
	0	1
$\rightarrow q_0$	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

$Q = \{q_0, q_1, q_2, q_3\}$, $\Sigma = \{0, 1\}$ và $F = \{q_0\}$, q_0 vừa là trạng thái bắt đầu (có mũi tên đi đến) vừa là trạng thái kết thúc (nằm trong một hình tròn).

Phương pháp biểu diễn bằng đồ thị

Hàm chuyển trạng thái có thể biểu diễn dưới dạng đồ thị có hướng, trong đó mỗi trạng thái là một đỉnh. Nếu từ ký tự vào $a \in \Sigma$, và từ trạng thái q ôtômát chuyển sang trạng thái p ($\sigma(q, a) = p$) thì sẽ có cung đi từ q đến p với nhãn là a . Đỉnh ứng với trạng thái bắt đầu (q_0) có mũi tên đến, những đỉnh ứng với các trạng thái kết thúc (thuộc tập F) được khoanh trong vòng tròn kép, những đỉnh còn lại được khoanh trong vòng tròn đơn.

Ví dụ 3.3 Ôtômát hữu hạn có hàm chuyển trạng thái được xác định theo bảng ở ví dụ 3.2 được biểu diễn bằng đồ thị như sau:



Hình H3-3 Biểu diễn đồ thị của ôtômát

3.1.3 Ngôn ngữ đoán nhận được của ôtômát

Định nghĩa. Một chuỗi x được đoán nhận bởi ôtômát hữu hạn M với

$$M = (Q, \Sigma, \sigma, q_0, F), \text{ nếu } \sigma(q_0, x) = q, \text{ với } q \in F, \text{ nghĩa là } \sigma(q_0, x) \in F.$$

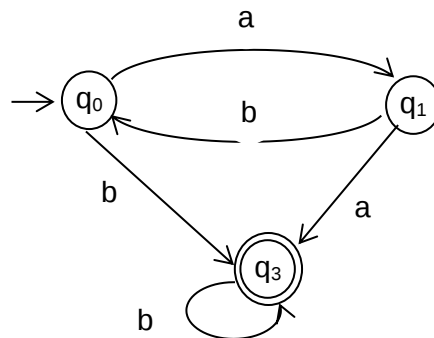
Định nghĩa. Tập tất cả các chuỗi (còn được gọi là ngôn ngữ) được đoán nhận bởi ôtômát hữu hạn $M = (Q, \Sigma, \sigma, q_0, F)$, được ký hiệu là

$$T(M) = \{x \in \Sigma^* \mid \sigma(q_0, x) \in F\}$$

Ví dụ 3.4 Xét ôtômát hữu hạn cho trước ở ví dụ 3.2. Ta dễ dàng kiểm tra chuỗi $x = 110101$ là chấp nhận được bởi M . Thật vậy,

$$\begin{aligned} \sigma(q_0, 110101) &= \sigma(q_1, 10101) = \sigma(q_0, 0101) \\ &= \sigma(q_2, 101) = \sigma(q_3, 01) \\ &= \sigma(q_1, 1) = \sigma(q_0, \Lambda) = q_0 \in F \end{aligned}$$

Ví dụ 3.5 Xét ôtômát hữu hạn M được cho như ở hình sau



Hình H3-4 Biểu diễn đồ thị của ôtômát hữu hạn M cho trước

Dựa vào các tính chất và cách biểu diễn của hàm chuyển trạng thái trên đồ thị định hướng, chúng ta nhận thấy:

- Một xâu (một từ) đoán nhận được bởi M là dãy các ký tự đầu vào (các nhãn trên các cung) trên đường đi từ đỉnh bắt đầu đến một trạng thái kết thúc nào đó.
- Ngôn ngữ đoán nhận được bởi M là tập tất cả các xâu được ghép từ các nhãn trên tất cả các đường đi từ đỉnh bắt đầu đến các đỉnh kết thúc.

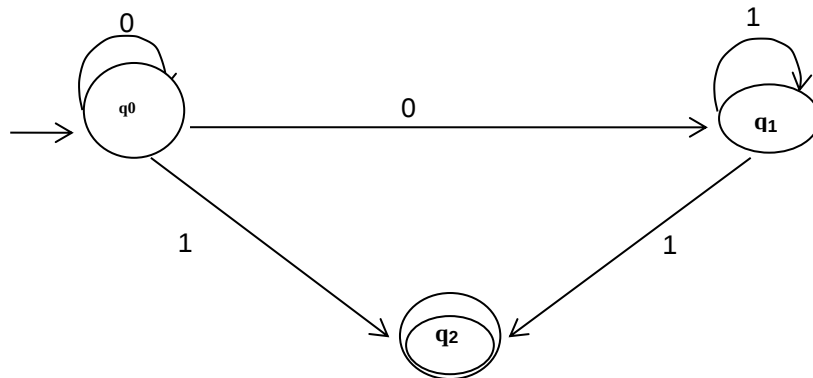
Từ nhận xét trên, ta suy ra ngôn ngữ đoán nhận được bởi M sẽ là

$$T(M) = \{a(ba)^n a, b^m \mid n \geq 0, m > 0\}$$

3.2 Ôtômát hữu hạn không đơn định

Chúng ta đã nghiên cứu những ôtômát mà từ một trạng thái (chưa kết thúc) và một ký tự vào thì trạng thái tiếp theo của chúng được xác định duy nhất. Tiếp theo chúng ta xét trường hợp những ôtômát hữu hạn mà trạng thái tiếp theo như trên không xác định duy nhất.

Chúng ta hãy xét ôtômát được cho bởi đồ thị như trong Hình H3-5.



Hình H3-5 Hệ biến đổi biểu diễn ôtômát hữu hạn không đơn định

Khi ở trạng thái q_0 và dữ liệu vào là 0, thì ôtômát chuyển đến trạng thái nào?. Theo hệ thống trên thì nó có thể hoặc chuyển đến q_1 hoặc quay vòng trở lại chính q_0 . Như vậy, trạng thái tiếp theo của ôtômát không xác định duy nhất khi nhận dữ liệu vào 0. Những ôtômát như vậy được gọi là *ôtômát hữu hạn không đơn định* (không tất định).

Định nghĩa. Một ôtômát hữu hạn không đơn định (ÔTKĐĐ), gọi tắt là *ôtômát không đơn định*, là bộ 5 phần tử $M = (Q, \Sigma, \sigma, q_0, F)$, trong đó

- Q là tập hữu hạn khác rỗng các trạng thái,
- Σ là tập hữu hạn khác rỗng các ký hiệu vào (bảng chữ cái), với $\Sigma \cap Q = \emptyset$,
- σ là ánh xạ từ $Q \times \Sigma$ vào 2^Q (tập tất cả các tập con của Q),

σ là ánh xạ từ $Q \times \Sigma$ vào Q và được gọi là *hàm chuyển trạng thái* (M là ÔTĐĐ)

(iv) $q_0 \in Q$ là trạng thái khởi đầu,

(v) $F \subseteq Q$ là tập các trạng thái kết thúc. Tổng quát thì $|F| \geq 1$.

Chúng ta nhận thấy sự khác nhau giữa đơn định (tắt định) và không đơn định của ôtômát chỉ là ở hàm chuyển trạng thái σ . Đối với ôtômát đơn định (ÔTĐĐ), kết quả của σ chỉ là một trạng thái, là một phần tử của Q ; còn đối với ôtômát không đơn định, kết quả của hàm chuyển trạng thái σ là một tập con các trạng thái của Q .

Tương tự như đối với ôtômát đơn định, hàm chuyển trạng thái được xác định trong định nghĩa ôtômát không đơn định là một ánh xạ xác định trên các cặp trạng thái và ký hiệu vào, do vậy không thể sử dụng trực tiếp để đoán nhận các xâu, mà phải mở rộng σ thành ánh xạ

$$\sigma: 2^Q \times \Sigma^* \rightarrow 2^Q \quad \text{nờ các tính chất sau.}$$

Tính chất.

$$(i) \quad \forall w \in \Sigma^*, \quad \forall q \in Q, \quad \sigma(q, w) = \sigma(\{q\}, w) \quad (3.4)$$

$$(ii) \quad \forall S \subseteq Q, \quad \sigma(S, \Lambda) = S, \quad (3.5)$$

$$(iii) \quad \forall S \subseteq Q, \quad \forall a \in \Sigma, \quad \sigma(S, a) = \bigcup_{q \in S} \sigma(q, a) \quad (3.6)$$

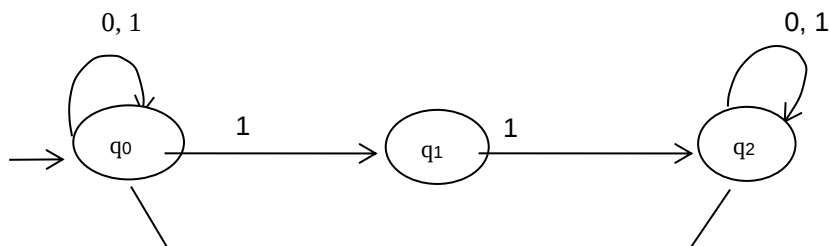
$$(iv) \quad \forall S \subseteq Q, \quad \forall a \in \Sigma, \quad \forall w \in \Sigma^*, \quad \sigma(S, aw) = \sigma(\sigma(S, a), w) \quad (3.7)$$

Định nghĩa. Xâu $w \in \Sigma^*$ *đoán nhận được* bởi một ÔTKĐĐ M nếu $\sigma(q_0, w)$ có chứa một số (ít nhất một) trạng thái kết thúc.

Lưu ý:

- Khi M là ôtômát không đơn định, $\sigma(q_0, w)$ có thể có nhiều hơn một trạng thái. Do vậy, w được đoán nhận bởi M nếu từ q_0 và đọc dữ liệu vào lần lượt theo w (từ trái qua phải) ta có một cách để đạt đến được trạng thái kết thúc.
- Trong đồ thị biểu diễn cho ôtômát không đơn định M , $w \in \Sigma^*$ *đoán nhận được* bởi M cũng chính là dãy các nhãn trên các cung của một đường đi từ đỉnh bắt đầu đến ít nhất một đỉnh kết thúc.

Ví dụ 3.6 Xét ôtômát không đơn định được biểu diễn bằng biểu đồ chuyển trạng thái như hình H3-6.



Hình H3-6 Ôtômát không đơn định M

Lưu ý: Khi ở một đỉnh có nhiều hơn một vòng khuyên (một cung quay lại chính nó) với các nhãn khác nhau thì có thể chỉ cần vẽ một vòng khuyên với các nhãn cách nhau bằng dấu phẩy ‘,’ như ở các đỉnh q_0, q_1, q_4 ở hình trên.

Dãy các trạng thái xác định bởi dãy vào 0100 được mô tả qua hàm chuyển trạng thái như sau:

$$\begin{aligned}\sigma(q_0, 0100) &= \sigma(\{q_0\}, 0100) && \text{Theo (3.4)} \\ &= \sigma(\sigma(\{q_0\}, 0), 100) && \text{Theo (3.7)} \\ &= \sigma(\sigma(q_0, 0), 100) && \text{Theo (3.6)} \\ &= \sigma(\{q_0\}, 100) = \sigma(\sigma(\{q_0\}, 1), 00) \\ &= \sigma(\sigma(\{q_0, q_3\}, 0), 0) \\ &= \sigma(\sigma(q_0, 0) \cup \sigma(q_3, 0), 0) && \text{Theo (3.6)} \\ &= \sigma(\{q_4\} \cup \sigma(q_3, 0), 0) \\ &= \sigma(q_4, 0) \cup \sigma(\sigma(q_3, 0), 0) \\ &= \{q_4\} \cup \sigma(\sigma(q_3, 0), 0)\end{aligned}$$

Vì q_4 là trạng thái kết thúc nên dãy 0100 là đoán nhận được bởi ôtômát không đơn định trên.

Tương tự, dãy 0100 cũng chính là dãy của các nhãn trên đường đi từ q_0, q_0, q_3, q_4 , và kết thúc ở q_4 , nên nó được đoán nhận bởi M (hình H3-6).

Từ đó, chúng ta có tập tất cả các từ đoán nhận bởi một ôtômát được định nghĩa như sau.

Định nghĩa. Tập các xâu (từ) được M (đơn định hoặc không đơn định) *đ đoán nhận* là tập tất cả các xâu dữ liệu (dãy các ký tự) đầu vào được đoán nhận bởi M ; ký hiệu $T(M)$.

$$T(M) = \{w \in \Sigma^* \mid \sigma(q_0, w) \cap F \neq \emptyset\}$$

Tập các xâu được đoán nhận (hay đoán nhận được) bởi một ôtômát thể hiện khả năng tính toán của hệ thống và mô tả hành vi của một ôtômát. Tập đoán nhận được bởi một ôtômát thường còn được gọi là *ngôn ngữ được đoán nhận bởi ôtômát đó*.

3.3 Sự tương đương của ôtômát đơn định và không đơn định

Ở trên chúng ta đã phân biệt hai loại ôtômát đơn định (ÔTĐĐ) và không đơn định (ÔTKĐĐ). Vấn đề quan trọng là chúng có quan hệ với nhau như thế nào?, nghĩa là ngôn ngữ đoán nhận bởi ÔTKĐĐ và ÔTĐĐ có quan hệ gì với nhau hay không?

Một cách trực quan chúng ta nhận thấy:

+ ÔTĐĐ có thể mô phỏng hành vi của ÔTKĐĐ bằng cách gia tăng số các trạng thái. Nói cách khác, ÔTĐĐ $M = (Q, \Sigma, \sigma, q_0, F)$ có thể xem như $M' = \text{ÔTKĐĐ}(Q, \Sigma, \sigma', q_0, F')$ bằng cách định nghĩa $\sigma'(q, a) = \{\sigma(q, a)\}$, nói cách khác ÔTĐĐ là trường hợp đặc biệt của ÔTKĐĐ.

+ ÔTKĐĐ là tổng quát hơn, song có vẻ như không mạnh hơn, nghĩa là tập mà nó đoán nhận không nhiều hơn ÔTĐĐ.

Điều thứ hai được khẳng định bởi định lý sau.

Định lý. Với mọi ÔTKĐĐ M luôn tồn tại ÔTĐĐ M' tương đương theo nghĩa chúng đoán nhận cùng một tập dữ liệu, $T(M) = T(M')$.

Chứng minh: Giả sử ÔTKĐĐ $M = (Q, \Sigma, \sigma, q_0, F)$ đoán nhận L . Chúng ta xây dựng một ôtômát ÔTĐĐ $M' = (Q', \Sigma, \sigma', q'_0, F')$ như sau:

- (i) $Q' = 2^Q$,
- (ii) $q'_0 = \{q_0\}$,
- (iii) F' là tập tất cả các tập con của Q có chứa phần tử của F ,
 $F' = \{S \subseteq Q \mid S \cap F \neq \emptyset\}$

Trước khi xác định σ' , chúng ta hãy khảo sát các đặc điểm của Q' , q_0' và F' . M khởi động từ q_0 và với một dữ liệu vào bất kỳ, ví dụ a , M có thể chuyển sang một trong các trạng thái của $\sigma(q_0, a)$. Để khảo sát hoạt động của M khi xử lý a , chúng ta phải xét tất cả các trạng thái có thể đạt được khi nó xử lý a . Vì thế các trạng thái của M' sẽ là các tập con của Q . Khi M bắt đầu với q_0 , q_0' được định nghĩa như trên sẽ là trạng thái khởi đầu của M' . Mặt khác xâu $w \in T(M) = L$ nếu M đạt đến trạng thái kết thúc khi xử lý w . Do vậy, trạng thái kết thúc của M' (phần tử của F') là tập con của Q và có chứa một số trạng thái kết thúc của M .

Tiếp theo chúng ta định nghĩa σ' :

$$(iv) \quad \sigma'(\{q_1, q_2, \dots, q_i\}, a) = \sigma(q_1, a) \cup \sigma(q_2, a) \cup \dots \cup \sigma(q_i, a)$$

Nghĩa là $\sigma'(\{q_1, q_2, \dots, q_i\}, a) = \{p_1, p_2, \dots, p_k\}$ khi và chỉ khi

$$\sigma(\{q_1, q_2, \dots, q_i\}, a) = \{p_1, p_2, \dots, p_k\} \text{ với } i, k \leq |Q|.$$

Trước khi chứng minh $L = T(M')$, chúng ta chứng minh kết quả bổ trợ:

$$\sigma'(q_0, x) = \{q_1, q_2, \dots, q_i\}, \text{ khi và chỉ khi}$$

$$\sigma(q_0, x) = \{q_1, q_2, \dots, q_i\} \text{ với mọi } x \text{ trong } \Sigma^*. \quad (3.8)$$

Chúng ta chứng minh điều kiện cần của (2.8) qui nạp theo $|x|$.

$$\text{Nếu } \sigma(q_0, x) = \{q_1, q_2, \dots, q_i\} \text{ thì } \sigma'(q_0', x) = \{q_1, q_2, \dots, q_i\} \quad (3.9)$$

Khi $|x| = 0$, hiển nhiên $\sigma(q_0, \Lambda) = \{q_0\}$ và theo định nghĩa σ' , $\sigma'(q_0', \Lambda) = q_0' = \{q_0\}$, nghĩa là (3.9) đúng với $|x| = 0$.

Giả thiết (3.9) đúng với mọi y và $|y| \leq n$. Xét xâu x có độ dài $n + 1$. Ta có thể viết $x = ya$, trong đó $|y| = n$ và $a \in \Sigma$. Giả sử $\sigma(q_0, y) = \{p_1, p_2, \dots, p_k\}$ và $\sigma(q_0, ya) = \{r_1, r_2, \dots, r_l\}$. Bởi $|y| \leq n$ nên theo giả thiết qui nạp ta có

$$\sigma'(q_0', y) = \{p_1, p_2, \dots, p_k\} \quad (3.10)$$

Mặt khác, $\{r_1, r_2, \dots, r_l\} = \sigma(q_0, ya) = \sigma(\sigma(q_0, y), a) = \sigma(\{p_1, p_2, \dots, p_k\}, a)$. Theo định nghĩa của σ' ,

$$\sigma'(\{p_1, p_2, \dots, p_k\}, a) = \{r_1, r_2, \dots, r_l\}. \quad (3.11)$$

Do vậy, $\sigma'(q_0', x) = \sigma'(q_0', ya) = \sigma'(\sigma'(q_0', y), a) = \sigma'(\{p_1, p_2, \dots, p_k\}, a)$

$$= \{r_1, r_2, \dots, r_l\} \text{ theo (3.11).}$$

Suy ra (3.9) được chứng minh với $x = ya$.

Điều kiện đủ chứng minh tương tự.

Từ đó suy ra, $x \in L = T(M)$ khi và chỉ khi $\sigma(q, x)$ chứa một trạng thái của F . Bởi vì $\sigma(q_0, x)$ chứa trạng thái thuộc F khi và chỉ khi $\sigma'(q_0', x)$ nằm trong F' . Điều đó khẳng định $x \in T(M) \leftrightarrow x \in T(M')$, đồng nghĩa với việc M' cũng đoán nhận đúng L . ■

Ví dụ 3.7 Xét ôtômát không đơn định $M = (\{q_0, q_1\}, \{0, 1\}, \sigma, q_0, \{q_0\})$ với hàm chuyển trạng thái σ được xác định theo bảng B3.2.

Bảng B3.2 Bảng các trạng thái của M

Trạng thái \ Σ	0	1
$\Rightarrow \textcircled{q_0}$	q_0	q_1
q_1	q_1	q_0, q_1

Xây dựng ôtômát đơn định tương đương M' theo định lý 3.1, có các thành phần:

- (i) $Q' = \{\emptyset, \{q_0\}, \{q_1\}, \{q_0, q_1\}\}$,
- (ii) $q_0' = \{q_0\}$,
- (iii) $F' = \{\{q_0\}, \{q_0, q_1\}\}$, những tập con các trạng thái có chứa q_0 ,
- (iv) σ' được định nghĩa như trong bảng B3.3

Bảng B3.3 Bảng các trạng thái của M'

Trạng thái \ Σ	0	1
$\emptyset$	$\emptyset$	$\emptyset$
$\longrightarrow \{q_0\}$	$\{q_0\}$	$\{q_1\}$
$\{q_1\}$	$\{q_1\}$	$\{q_0, q_1\}$

$\{q_0, q_1\}$

$\{q_0, q_1\}$

$\{q_0, q_1\}$

Nhận xét: Khi M có n trạng thái, M' tương ứng sẽ có thể có 2^n trạng thái. Tuy nhiên, chúng ta không nhất thiết phải xây dựng ôtômát có tất cả 2^n trạng thái mà chỉ cần những trạng thái đạt đến từ trạng thái khởi đầu và đi đến được trạng thái kết thúc.

Trong bảng chuyển trạng thái B3.3 trạng thái $\emptyset$ là không đạt đến được từ trạng thái bắt đầu $\{q_0\}$ nên có thể loại bỏ đi. Ngoài ra, để cho tiện lợi chúng ta có thể ký hiệu lại tập các trạng thái, ví dụ $s_0 = \{q_0\}$, $s_1 = \{q_1\}$, $s_2 = \{q_0, q_1\}$ và $Q' = \{s_0, s_1, s_2\}$. Khi đó, ôtômát có hàm chuyển trạng thái như trên sẽ được viết thành

Bảng B3.4 Bảng các trạng thái của M'

Trạng thái \ Σ	0	1
$\rightarrow \textcircled{s_0}$	s_0	s_1
s_1	s_1	s_2
$\textcircled{s_2}$	s_2	s_2

Tổng quát hóa quá trình trên, chúng ta có thể xây dựng thuật toán thiết lập M' như sau.

Thuật toán. Xây dựng hàm chuyển trạng thái của M' tương đương với M (Phép đơn định hóa)

Input: Cho trước ôtômát $M = (Q, \Sigma, \sigma, q_0, F)$ không đơn định.

Output: Hàm chuyển trạng thái σ' của ôtômát đơn định tương đương.

1. Xây dựng σ' bắt đầu từ $\{q_0\}$,
2. Xác định trạng thái đạt được từ những trạng thái (tập con các trạng thái của M) tương ứng với các cột dữ liệu vào đã xác định từ trước,
3. Lặp lại bước 2 cho đến khi không có một trạng thái mới xuất hiện trong các cột ứng với dữ liệu vào.

Ví dụ 3.8 Tìm ôtômát đơn định tương đương với

$$M = (\{q_0, q_1, q_2\}, \{a, b\}, \sigma, q_0, \{q_2\})$$

và σ được cho trong bảng B3.5.

Bảng B3.5 Bảng các trạng thái của M

Trạng thái \ Σ	a	b
$\rightarrow q_0$	$\{q_0, q_1\}$	q_2
q_1	q_0	q_1
$\odot q_2$		$\{q_0, q_1\}$

Ôtômát đơn định tương đương với M là $M' = (2^Q, \{a, b\}, \sigma', \{q_0\}, F')$, trong đó

$F' = \{\{q_2\}, \{q_0, q_2\}, \{q_1, q_2\}, \{q_0, q_1, q_2\}\}$. Thực hiện theo thuật toán 3.1 ta thu được σ' như ở bảng B3.6.

Bảng B3.6 Bảng các trạng thái của M'

Trạng thái / Σ	a	b
$\rightarrow \{q_0\}$	$\{q_0, q_1\}$	$\{q_2\}$
$\{q_2\}$	$\emptyset$	$\{q_0, q_1\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_1, q_2\}$
$\{q_1, q_2\}$	$\{q_0\}$	$\{q_0, q_1\}$

3.4 Cực tiểu hoá ôtômát hữu hạn

Trong phần này chúng ta xét vấn đề tìm *ôtômát có số trạng thái cực tiểu tương đương* (cùng đoán nhận một ngôn ngữ) với ôtômát cho trước.

Trên tập các trạng thái Q chúng ta định nghĩa một số quan hệ tương đương.

Định nghĩa. Hai trạng thái q_1 và q_2 được gọi là tương đương, ký hiệu $q_1 \equiv q_2$, nếu cả hai $\sigma(q_1, x)$ và $\sigma(q_2, x)$ đều là những trạng thái kết thúc hoặc cả hai đều không kết thúc với mọi $x \in \Sigma^*$.

Số các xâu (từ) được xây dựng từ bảng ký tự vào Σ thường là khá lớn (có khi là vô hạn), nhưng trong phần này chúng ta chỉ xét những trường hợp hữu hạn.

Định nghĩa. Hai trạng thái q_1 và q_2 được gọi là *k-tương đương* ($k \geq 0$), ký hiệu $q_1 \equiv_k q_2$, nếu cả hai $\sigma(q_1, x)$ và $\sigma(q_2, x)$ đều là những trạng thái kết thúc hoặc cả hai đều không kết thúc với mọi $x \in \Sigma^*$ có độ dài nhỏ hơn k .

Hiển nhiên, hai trạng thái kết thúc hoặc hai trạng thái không kết thúc đều là 0-tương đương.

Các quan hệ trên có một số tính chất như sau.

Tính chất. Hai quan hệ $\equiv$ và $\equiv_k$ là các quan hệ tương đương (có tính phản xạ, bắc cầu và đối xứng).

Tính chất. Các lớp tương đương của hai quan hệ $\equiv$ và $\equiv_k$ sẽ tạo ra hai phân hoạch π , π_k của Q ; các phần tử của π_k là các lớp k-tương đương.

Tính chất. Nếu $q_1 \equiv q_2$ thì $q_1 \equiv_k q_2$, với mọi $k \geq 0$.

Tính chất. Nếu $q_1 \equiv_{(k+1)} q_2$ thì $q_1 \equiv_k q_2$.

Tính chất. Tồn tại n để $\pi_n = \pi_{n+1}$.

Mệnh đề sau sẽ là cơ sở để xây dựng ôtômát cực tiểu.

Mệnh đề. Hai trạng thái q_1 và q_2 được gọi là $(k+1)$ -tương đương nếu

(a) Chúng là k-tương đương,

(b) $\sigma(q_1, a)$ và $\sigma(q_2, a)$ cũng là k-tương đương với mọi $a \in \Sigma$.

Chứng minh: Chúng ta chứng minh bằng phản chứng. Giả sử q_1 và q_2 không phải là $(k+1)$ -tương đương. Khi đó tồn tại $w = aw_1$ với độ dài $k+1$ sao cho $\sigma(q_1, aw_1)$ là trạng thái kết thúc nhưng $\sigma(q_2, aw_1)$ lại không phải là trạng thái kết thúc. Như vậy, $\sigma(q_1, aw_1) = \sigma(\sigma(q_1, a), w_1)$ là trạng thái kết thúc và $\sigma(q_2, aw_1) = \sigma(\sigma(q_2, a), w_1)$ không phải là trạng thái kết thúc. Từ đó suy ra $\sigma(q_1, a)$ và $\sigma(q_2, a)$ không phải là k-tương đương, mâu thuẫn với giả thiết (b). ■

Dựa vào các tính chất trên chúng ta có thể xây dựng các lớp $(k+1)$ -tương đương khi biết các lớp k-tương đương trên tập các trạng thái và từ đó xây dựng được thuật toán cực tiểu hoá ôtômát (có số trạng thái cực tiểu) tương đương với ôtômát cho trước.

Thuật toán. Xây dựng ôtômát cực tiểu hoá

Input: Cho trước ôtômát $M = (Q, \Sigma, \sigma, q_0, F)$ thường là không đơn định.

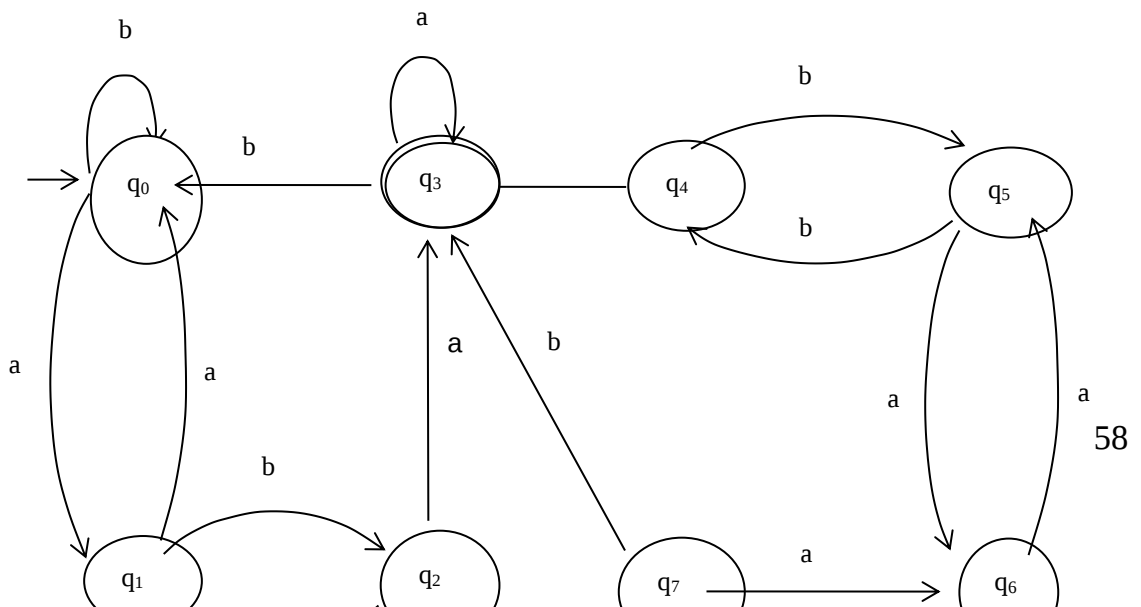
Output: Ôtômát đơn định và cực tiểu $M' = (Q', \Sigma, \sigma', q_0', F')$

1. Thiết lập phân hoạch π_0 . Theo định nghĩa 0-tương đương, $\pi_0 = \{Q_1^0, Q_2^0\}$, trong đó $Q_1^0 = F$ (tập các trạng thái kết thúc), $Q_2^0 = Q - Q_1^0$.
2. Xây dựng π_{k+1} từ π_k . Xây dựng $Q_i^k, i = 1, 2, \dots$ là các tập con của π_k và là các lớp $(k+1)$ -tương đương. q_1 và q_2 nằm trong Q_i^k nếu chúng là $(k+1)$ -tương đương nghĩa là $\sigma(q_1, a)$ và $\sigma(q_2, a)$ là k -tương đương, với mọi a trong bảng chữ vào. Điều này xảy ra khi $\sigma(q_1, a)$ và $\sigma(q_2, a)$ nằm trong cùng lớp tương đương của π_k . Do vậy, Q_i^k là lớp $(k+1)$ -tương đương. Thực hiện như trên cho đến khi các tập con Q_i^k tạo thành một phân hoạch π_{k+1} của Q mịn hơn π_k .
3. Lặp lại bước 2 để thiết lập π_k với $k = 1, 2, \dots$ cho đến khi $\pi_k = \pi_{k+1}$.
4. Xây dựng ôtômát cực tiểu. Các trạng thái của ôtômát cực tiểu chính là các lớp tương đương được xác định như trong bước 3, đó là các phần tử của π_k . Bảng chuyển trạng thái thu được bằng cách thay trạng thái q bằng lớp tương đương tương ứng $[q]$.

Lưu ý:

1. Dựa vào bảng chuyển trạng thái cho trước ta dễ dàng xây dựng được các lớp $\pi_0 = \{Q_1^0, Q_2^0\}$, $Q_1^0 = F$, $Q_2^0 = Q - F$;
2. Giả sử đã xây dựng được $\pi_k, k = 0, 1, \dots$. Với $q_1, q_2 \in Q_i^k, Q_i^k \in \pi_k, i = 1, 2, \dots$ xét các trạng thái ở các cột tương ứng với dữ liệu vào $a \in \Sigma$, nếu $\sigma(q_1, a)$ và $\sigma(q_2, a)$ cùng thuộc một tập con nào đó của π_k thì là $(k+1)$ -tương đương, nghĩa là q_1, q_2 cùng nằm trong một phân hoạch của π_{k+1} ; ngược lại sẽ không phải là $(k+1)$ -tương đương, nghĩa là q_1, q_2 nằm trong hai phân hoạch khác nhau của π_{k+1} .
3. Lặp lại bước 2 cho đến khi $\pi_k = \pi_{k+1}$.

Ví dụ 3.9 Xây dựng ôtômát cực tiểu tương đương với ôtômát có đồ thị chuyển trạng thái như trong hình H3-9.



Hình H3-9 Đồ thị chuyển trạng thái của ví dụ 3.9

Từ đồ thị chuyển trạng thái ở hình H3.9 chúng ta có bảng chuyển trạng thái.

Bảng B3.7 Bảng các trạng thái của M

Trạng thái \ Σ	a	b
$\rightarrow q_0$	q_1	q_0
q_1	q_0	q_2
q_2	q_3	q_1
$\textcircled{q_3}$	q_3	q_0
q_4	q_3	q_5
q_5	q_6	q_4
q_6	q_5	q_6
q_7	q_6	q_3

- Theo thuật toán trên ta có $Q_1^0 = \{q_3\}$, $Q_2^0 = \{q_0, q_1, q_2, q_4, q_5, q_6, q_7\}$ và $\pi_0 = \{\{q_3\}, \{q_0, q_1, q_2, q_4, q_5, q_6, q_7\}\}$.

- Chúng ta xét tiếp quan hệ 1-tương đương để tính π_1 . Trước tiên ta có $Q_1^1 = \{q_3\}$. Dựa vào bảng B2.7 chúng ta dễ kiểm tra được q_0 là 1-tương đương với q_1, q_5, q_6 vì $\sigma(q_0, t)$ và $\sigma(q_i, t)$, $i = 1, 5, 6$ và $t = a, b$ là cùng kết thúc hoặc cùng không phải là trạng thái kết thúc. Vậy $Q_2^1 = \{q_0, q_1, q_5, q_6\}$. Tương tự ta có thể kiểm tra được q_2 -tương đương với q_4 và $Q_3^1 = \{q_2, q_4\}$. Hiên nhiên còn lại $Q_4^1 = \{q_7\}$. Từ đó chúng ta có

$$\pi_1 = \{\{q_3\}, \{q_0, q_1, q_5, q_6\}, \{q_2, q_4\}, \{q_7\}\}.$$

- Phân hoạch tiếp π_1 để xây dựng π_2 . $Q_1^2 = \{q_3\}$ và vì $\sigma(q_0, t)$ và $\sigma(q_6, t)$, $t = a, b$ đều thuộc cùng lớp 1-tương đương (lớp $\{q_0, q_1, q_5, q_6\}$) nên q_0 và q_6 là 2-tương đương. Vậy $Q_2^2 = \{q_0, q_6\}$. Nhưng q_0 và q_1 (q_0 và q_5) không phải là 2-tương đương vì $\sigma(q_0, b) = q_0$ ($\sigma(q_1, b) = q_2$, $\sigma(q_0, b) = q_0$, $\sigma(q_1, b) = q_2$) và chúng lại không cùng lớp 1-tương đương. Kiểm tra ta thấy q_1 là 2-tương đương với q_5 , nên $Q_3^2 = \{q_1, q_5\}$. Tương tự q_2 vẫn là 2-tương đương với q_4 , vì thế $Q_4^2 = \{q_2, q_4\}$ và còn lại $Q_5^2 = \{q_7\}$. Vậy

$$\pi_2 = \{\{q_3\}, \{q_0, q_6\}, \{q_1, q_5\}, \{q_2, q_4\}, \{q_7\}\}$$

- Tiếp tục xây dựng π_3 . Tương tự như trên, chúng ta thấy $Q_1^3 = \{q_3\}$; $Q_2^3 = \{q_0, q_6\}$ vì q_0 là 3-tương đương với q_6 ; tương tự $Q_3^3 = \{q_1, q_5\}$; $Q_4^3 = \{q_2, q_4\}$ và $Q_5^3 = \{q_7\}$. Nghĩa là $\pi_3 = \{\{q_3\}, \{q_0, q_6\}, \{q_1, q_5\}, \{q_2, q_4\}, \{q_7\}\}$.

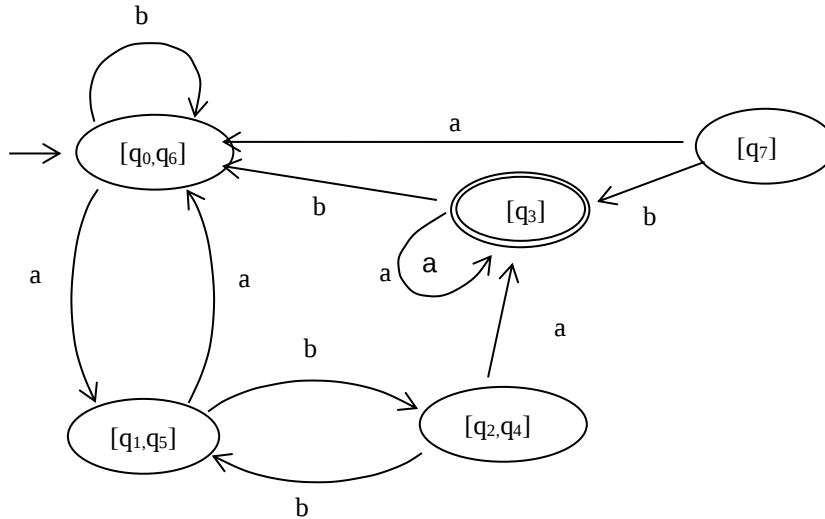
Vì $\pi_2 = \pi_3$, nên các lớp tương đương trong π_2 sẽ làm cơ sở để xây dựng ôtômát cực tiểu $M' = (Q', \{a, b\}, \sigma', q_0', \{q_3\})$. Trong đó $Q' = \pi_2 = \{\{q_3\}, \{q_0, q_6\}, \{q_1, q_5\}, \{q_2, q_4\}, \{q_7\}\}$.

Hàm chuyển trạng thái σ' được xác định như trong bảng B3.8.

Bảng B3.8 Bảng các trạng thái của ôtômát cực tiểu

Trạng thái \ Σ	a	b
$\rightarrow \{q_0, q_6\}$	$\{q_1, q_5\}$	$\{q_0, q_6\}$
$\{q_1, q_5\}$	$\{q_0, q_6\}$	$\{q_2, q_4\}$
$\{q_2, q_4\}$	$\{q_3\}$	$\{q_1, q_5\}$
$\{q_3\}$	$\{q_3\}$	$\{q_0, q_6\}$
$\{q_7\}$	$\{q_0, q_6\}$	$\{q_3\}$

Đồ thị chuyển trạng thái của ôtômat cực tiểu thu được sẽ là



Hình H3-10 Ôtômat cực tiểu của ôtômat ở hình H3-9

Bài tập về ôtômat hữu hạn

3.1. Chứng minh rằng trong ôtômat hữu hạn, nếu $\sigma(q, x) = \sigma(q, y)$ thì $\sigma(q, xz) = \sigma(q, yz)$ với mọi z trong Σ^+ .

3.2. Cho trước ôtômat hữu hạn $M = (Q, \Sigma, \sigma, q_0, F)$. Giả sử R là một quan hệ trên Q được định nghĩa như sau $q_1 R q_2$ nếu $\sigma(q_1, a) = \sigma(q_2, a)$ với $a \in \Sigma$. Hỏi R có phải là quan hệ tương đương hay không?

3.3. Xây dựng một ôtômat không đơn định đoán nhận $\{ab, ba\}$, và sử dụng nó để tìm ôtômat đơn định đoán nhận cùng tập đó.

3.4. Xây dựng một hệ biến đổi đoán nhận được các xâu có chứa *cat* hoặc *rat* từ bảng chữ $a, b, c, \dots$

3.5. Xây dựng một ôtômat không đơn định đoán nhận được tập tất cả các xâu từ tập $\{a, b\}$ kết thúc bằng xâu con *aba*.

3.6. Xây dựng ôtômat đơn định tương đương với
 $M = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \sigma, q_0, \{q_3\})$ với σ được cho trong bảng

Bảng B

Trạng thái \ Σ	a	b
-----------------------	---	---

Ôtômát và ngôn ngữ hình thức

→

q_0	q_0, q_1	q_0
q_1	q_2	q_1
$\textcircled{q_2}$	q_3	q_3
q_3		q_2

3.7. $M = (\{q_1, q_2, q_3\}, \{0, 1\}, \sigma, q_1, \{q_3\})$ là ôtômát không đơn định, trong đó σ được cho bởi

$$\sigma(q_1, 0) = \{q_2, q_3\}, \sigma(q_1, 1) = \{q_1\}$$

$$\sigma(q_2, 0) = \{q_1, q_2\}, \sigma(q_2, 1) = \emptyset$$

$$\sigma(q_3, 0) = \{q_2\}, \quad \sigma(q_3, 1) = \{q_1, q_2\}$$

Tìm ôtômát đơn định tương đương với M .

3.8. Xây dựng ôtômát đơn định trên $\{0, 1\}$ để đoán nhận tất cả các xâu chứa chẵn lần các chữ số 1.

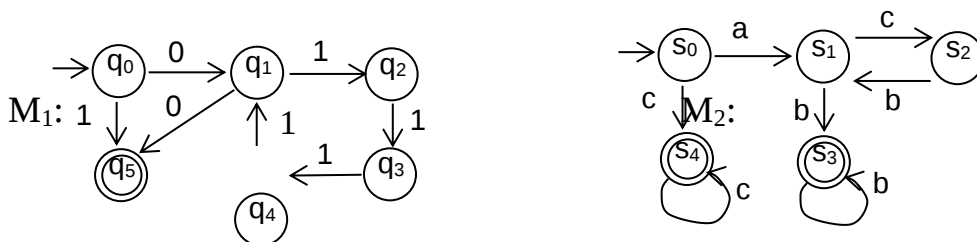
3.9. Xây dựng ôtômát đơn định trên $\Sigma = \{a_1, a_2, \dots, a_n\}$ để đoán nhận các ngôn ngữ sau:

a) $L_1 = \{x \in \Sigma^* \mid x \text{ có độ dài lẻ}\}$

b) $L_2 = \{x \in \Sigma^* \mid x \text{ có độ dài chẵn}\}$

c) $L_3 = \{x \in \Sigma^* \mid x \text{ có độ dài chẵn và lớn hơn 1}\}$

3.10* Cho hai ôtômát sau đây:



a/ Tìm ngôn ngữ $T(M_1)$ đoán nhận được bởi M_1 và $T(M_2)$ đoán nhận được bởi M_2

b/ Xây dựng ôtômát M từ M_1 và M_2 sao cho $T(M) = T(M_1)T(M_2)$

c/ Xây dựng ôtômát M từ M_1 và M_2 sao cho $T(M) = T(M_1) \cup T(M_2)$.

CHƯƠNG 4. TẬP CHÍNH QUI VÀ VĂN PHẠM CHÍNH QUI

Chương bốn đề cập đến:

- ✓ Tập chính qui và ôtômát hữu hạn trạng thái,
- ✓ Bỏ đề Bơm cho các tập chính qui và ứng dụng,
- ✓ Quan hệ giữa tập chính qui, văn phạm chính qui, ôtômát hữu hạn

4.1 Các biểu thức chính qui

Biểu thức chính qui được sử dụng để biểu diễn tập các xâu trong các cấu trúc đại số. Biểu thức chính qui mô tả những ngôn ngữ đoán nhận được bởi ôtômát hữu hạn trạng thái.

Định nghĩa. *Biểu thức chính qui* trên bảng chữ Σ được định nghĩa đệ qui như sau:

1. Mọi ký hiệu kết thúc $a \in \Sigma$, Λ và $\emptyset$ (tập rỗng) đều là biểu thức chính qui. Khi một ký hiệu $a \in \Sigma$ thì biểu thức chính qui tương ứng sẽ được ký hiệu là **a**.
2. Hợp của hai biểu thức chính qui R_1 và R_2 , ký hiệu là $R_1 + R_2$, cũng là biểu thức chính qui,
3. Ghép hai biểu thức chính qui R_1 và R_2 , ký hiệu là $R_1.R_2$, cũng là biểu thức chính qui,
4. Bao đóng của một biểu thức chính qui R , ký hiệu là R^* , cũng là biểu thức chính qui,
5. Nếu R là một biểu thức chính qui thì (R) cũng là biểu thức chính qui.
6. Biểu thức chính qui trên bảng chữ Σ là tập tất cả các hạng thức được xây dựng một cách đệ qui trên cơ sở chỉ áp dụng các qui tắc từ 1 – 5 nêu trên.

Chú ý:

- (i) Chúng ta ký hiệu **x** (in đậm) cho biểu thức chính qui để phân biệt với ký hiệu (hoặc xâu) x thông thường.

(ii) Cặp ngoặc đơn ‘(’ và ‘)’ trong qui tắc 5 được sử dụng để xác định thứ tự thực hiện các phép toán của các biểu thức chính qui.

(iii) Khi không có các ngoặc đơn thì thứ tự thực hiện trong biểu thức chính qui được qui định như sau: *Bao đóng, phép ghép rồi đến phép hợp.*

Định nghĩa. Tập biểu diễn cho một biểu thức chính qui được gọi là tập chính qui. Nếu $a, b \in \Sigma$ thì

1. Biểu thức chính qui a xác định tập $\{a\}$, nói cách khác tập $\{a\}$ được biểu diễn bởi a ,
2. Biểu thức chính qui $a + b$ xác định tập $\{a, b\}$, hoặc $\{a, b\}$ được biểu diễn bởi biểu thức $a + b$
3. Biểu thức chính qui ab xác định tập $\{ab\}$, hoặc $\{ab\}$ được biểu diễn bởi biểu thức ab
4. Biểu thức chính qui a^* xác định tập $\{\Lambda, a, aa, aaa, \dots\}$, hoặc $\{\Lambda, a, aa, aaa, \dots\}$ được biểu diễn bởi a^*
5. Biểu thức chính qui $(a + b)^*$ xác định tập $\{a, b\}^*$, hoặc tập $\{a, b\}^*$ được biểu diễn bởi $(a + b)^*$.

Để hiểu rõ hơn việc tính toán các biểu thức chính qui chúng ta cần xét kỹ hơn ba phép toán nêu trên. Giả sử R_1 và R_2 là hai biểu thức chính qui.

Một xâu trong $R_1 + R_2$ là xâu trong R_1 hoặc R_2 .

Một xâu trong R_1R_2 là xâu trong R_1 được ghép với xâu của R_2 .

Một xâu trong R^* là xâu được xây dựng từ phép ghép n các phần tử của R , với $n \geq 0$.

Tóm lại:

(i) Tập được biểu diễn bởi $R_1 + R_2$ là hợp của hai tập biểu diễn bởi R_1 và bởi R_2 .

Tập được biểu diễn bởi R_1R_2 là ghép của hai tập biểu diễn bởi R_1 và bởi R_2 .

Tập được biểu diễn bởi R^* là $\{w_1w_2 \dots w_n \mid w_i \text{ là tập con biểu diễn bởi } R \text{ và } n \geq 0\}$.

Ví dụ 4.1 Xác định các biểu thức chính qui biểu diễn cho các tập sau:

(ii) Tập $\{101\}$ được biểu diễn bởi biểu thức **101**,

(iii) $\{abba\}$ được biểu diễn bởi biểu thức **abba**,

(iv) $\{01, 10\}$ được biểu diễn bởi biểu thức **01 + 10**,

(v) $\{\Lambda, ab\}$ được biểu diễn bởi biểu thức **$\Lambda + ab$** ,

(vi) $\{abb, a, b, bba\}$ được biểu diễn bởi biểu thức **$abb + a + b + bba$** ,

- (vii) $\{\Lambda, 0, 00, 000, \dots\}$ được biểu diễn bởi biểu thức 0^* ,
(viii) $\{1, 11, 111, \dots\}$ được biểu diễn bởi biểu thức $1(1)^*$.

Ví dụ 4.2 Xác định các biểu thức chính qui biểu diễn cho các tập (ngôn ngữ) sau:

- (ix) L_1 là tập tất cả các xâu chứa 0, 1 và kết thúc bằng 00,
(x) L_2 là tập tất cả các xâu chứa 0, 1 được bắt đầu bằng 0 và kết thúc bằng 1,
(xi) $L_3 = \{\Lambda, 11, 1111, 111111, \dots\}$

Giải:

- (ix) Một xâu bất kỳ của L_1 sẽ được xây dựng từ phép ghép của xâu xác định trên $\{0, 1\}$ với xâu 00. $\{0, 1\}$ biểu diễn cho $0 + 1$. Vậy, L_1 được biểu diễn bởi $(0+1)^*00$.
(x) Mỗi phần tử của L_2 đều nhận được từ phép ghép 0 với các xâu xác định trên $\{0, 1\}$ và ghép tiếp với 1, nên L_2 sẽ được biểu diễn bởi $0(0+1)^*1$.
(xi) Mỗi phần tử của L_3 hoặc là Λ , hoặc là xâu chứa chẵn lần các số 1, nghĩa là những xâu có dạng $(11)^n, n \geq 0$. Vậy L_3 có thể được biểu diễn bởi $(11)^*$.

4.2 Sự tương đương của các biểu thức chính qui

Định nghĩa. Hai biểu thức chính qui P, Q được gọi là *tương đương*, (ta viết là $P = Q$) nếu P và Q cùng biểu diễn cho cùng một tập các xâu.

Các biểu thức chính qui có các tính chất sau:

$$I_1 \quad \emptyset + R = R$$

$$I_2 \quad \emptyset R = R\emptyset = \emptyset$$

$$I_3 \quad \Lambda R = R\Lambda = R$$

$$I_4 \quad \Lambda^* = \Lambda \text{ và } \emptyset^* = \Lambda$$

$$I_5 \quad R + R = R$$

$$I_6 \quad R^*R^* = R^*$$

$$I_7 \quad RR^* = R^*R$$

$$I_8 \quad (R^*)^* = R^*$$

$$I_9 \quad \Lambda + RR^* = \Lambda + R^*R = R^*$$

$$I_{10} \quad (PQ)^*P = P(QP)^*$$

$$I_{11} \quad (P + Q)^* = (Q^*P^*)^* = (P^* + Q^*)^*$$

$$I_{12} \quad (P + Q)R = PR + QR \text{ và } R(P+Q) = RP + RQ$$

Lưu ý: Cần phân biệt Λ với $\emptyset$, Λ ký hiệu cho một xâu (từ) rỗng (một số tài liệu khác sử dụng ε để ký hiệu cho xâu rỗng), còn $\emptyset$ ký hiệu của tập rỗng).

Định lý. (Định lý Arden) Giả sử P và Q là hai biểu thức chính qui trên Σ . Nếu P không chứa Λ thì phương trình sau

$$R = Q + RP \quad (4.1)$$

có nghiệm duy nhất là $R = QP^*$

Chứng minh: Thay $R = QP^*$ vào vế phải và theo tính chất I_9 chúng ta có

$$Q + (QP^*)P = Q(\Lambda + P^*P) = QP^*$$

Nghĩa là $R = QP^*$ thoả mãn đẳng thức (4.1).

Để chứng minh tính duy nhất của R , chúng ta hãy thế R bằng $Q + RP$ một số lần như sau.

$$\begin{aligned} Q + RP &= Q + (Q + RP)P \\ &= Q + QP + RPP \\ &= Q + QP + RP^2 \\ &\dots \\ &= Q + QP + QP^2 + \dots + QP^i + RP^{i+1} \\ &= Q(\Lambda + P + P^2 + \dots + P^i) + RP^{i+1} \end{aligned}$$

Vậy từ (4.1) chúng ta có

$$R = Q(\Lambda + P + P^2 + \dots + P^i) + RP^{i+1} \quad (4.2)$$

Nếu R thoả mãn (4.1) thì nó cũng thoả mãn (4.2). Giả sử w là một xâu có độ dài là i trong tập R . Khi đó w sẽ thuộc tập được xác định bởi $Q(\Lambda + P + P^2 + \dots + P^i) + RP^{i+1}$. Bởi vì P không chứa Λ nên RP^{i+1} sẽ không có các xâu có độ dài i . Do vậy, w phải thuộc $Q(\Lambda + P + P^2 + \dots + P^i)$, nghĩa là thuộc QP^* .

Ngược lại, xét xâu w thuộc QP^* , tức là w thuộc QP^k , với $k \geq 0$, từ đó suy ra nó cũng thuộc $Q(\Lambda + P + P^2 + \dots + P^i)$.

Như vậy chúng ta đã chứng minh được là R và QP^* biểu diễn cho cùng một tập hợp. Điều này khẳng định tính duy nhất của lời giải phương trình (4.1).

4.3 Ôtômát hữu hạn và biểu thức chính quy

Định nghĩa. Otomat $M = (Q, \Sigma, \sigma, q_0, F)$ chấp nhận (đoán nhận) xâu w trong Σ^* nếu

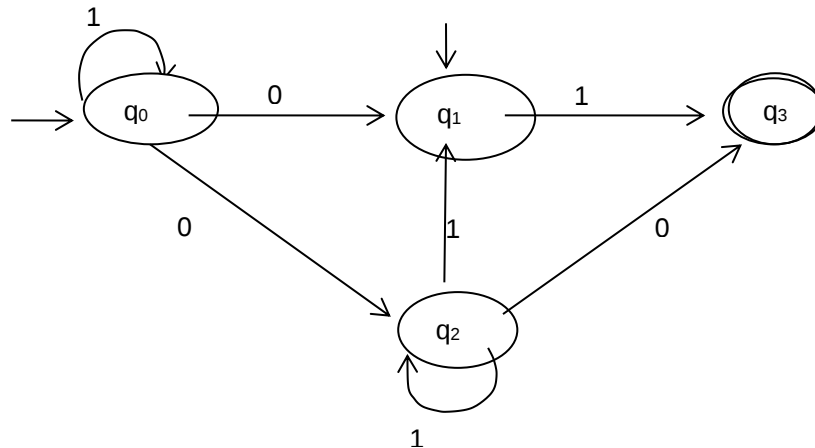
- (a) Có một đường đi trên đồ thị trạng thái, bắt đầu từ một đỉnh khởi đầu và kết thúc tại một đỉnh kết thúc (đỉnh thuộc F),
- (b) Các nhãn trên các cung dọc đường đi đó ghép lại tạo thành w .

Định nghĩa. Ngôn ngữ đoán nhận được bởi $M = (Q, \Sigma, \sigma, q_0, F)$, ký hiệu là $T(M)$, là tập tất cả các xâu đoán nhận được bởi M .

$$T(M) = \{w \in \Sigma^* \mid \sigma(q_0, w) \cap F \neq \emptyset\}$$

$T(M)$ là tập tất cả các dãy ký tự vào, trên tất cả các đường đi từ một đỉnh bắt đầu đến một đỉnh kết thúc.

Ví dụ 4.5 Cho trước hệ chuyển trạng thái được mô tả như hình H4-1.



Hình H4-1 Đồ thị chuyển trạng thái của Otomat M

Trong đó $Q_0 = \{q_0, q_1\}$ và $F = \{q_3\}$. Dễ nhận thấy hệ thống trên đoán nhận xâu $w = 1100$, vì có đường đi từ q_0, q_0, q_2, q_3 và các nhãn ghép lại thành w .

Ngôn ngữ đoán nhận được bởi M là $T = \{1^n 0 1^m 0, 1^n 0 1^m 11, 1^n 0 1, 1 \mid n, m \geq 0\}$ mà tập T này được biểu diễn bởi biểu thức chính qui $1^* 0 1^* 0 + 1^* 0 1^* 11 + 1^* 0 1 + 1$.

Nhận xét: Nếu P là biểu thức chính qui thì luôn xây dựng được một ôtômát hữu hạn trạng thái M để đoán nhận biểu thức chính qui đó và ngược lại: Mọi ngôn ngữ đoán nhận bởi ôtômát hữu hạn M đều biểu diễn được bởi một biểu thức chính qui

Mối quan hệ giữa ôtômát hữu hạn và biểu thức chính qui.

4.3.1 Cho trước Ôtômát hữu hạn M , tìm biểu thức chính quy tương đương.

Cho trước một Ôtômát hữu hạn M , dùng phương pháp đại số ứng dụng Định lý Arden, chúng ta tìm được *biểu thức chính qui* – là tập đoán nhận được bởi M .

Xét ôtômát hữu hạn được biểu diễn bằng đồ thị chuyển trạng thái với những giả thiết như sau:

- (i) Không có Λ -dịch chuyển trong đồ thị chuyển trạng thái
- (ii) Có một trạng thái bắt đầu là q_1
- (iii) Tập các đỉnh là $q_1, q_2, \dots, q_n$
- (iv) Ký hiệu $q_i, i = 1, 2, \dots, n$ là những biểu thức chính qui đại diện cho các dãy đoán nhận được bởi ôtômát khi đi đến trạng thái kết thúc q_i .
- (v) a_{ij} ký hiệu cho biểu thức chính qui đại diện cho tập các nhãn trên các cung từ q_i tới $q_j, i, j = 1, 2, \dots, n$. Nếu không có cung nào nối giữa chúng thì $a_{ij} = \emptyset$.

Dựa vào đồ thị chuyển trạng thái, chúng ta có thể xác định được hệ phương trình theo $q_1, q_2, \dots, q_n$, với q_1 là trạng thái bắt đầu như sau:

$$q_1 = q_1 a_{11} + q_2 a_{21} + \dots + q_n a_{n1} + \Lambda$$

$$q_2 = q_1 a_{12} + q_2 a_{22} + \dots + q_n a_{n2}$$

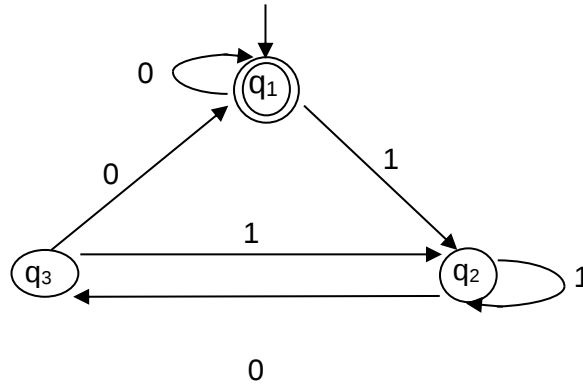
...

$$q_n = q_1 a_{1n} + q_2 a_{2n} + \dots + q_n a_{nn}$$

Thực hiện lặp lại một số lần phép thế và sử dụng định lý Arden (định lý 4.1), chúng ta dễ dàng biểu diễn được q_i bằng các biểu thức a_{ij} và các phép toán $(+, \cdot, *)$.

Nhận xét: Tập đoán nhận được bởi ôtômát hữu hạn sẽ tương đương với biểu thức chính qui là hợp của tất cả các biểu thức q_i ứng với tất cả các trạng thái kết thúc của ôtômát.

Ví dụ 4.7 Tìm biểu thức chính qui tương ứng Otomat với biểu đồ trạng thái sau:



Hình H4-2 Cho trước đồ thị trạng thái

Dựa vào đồ thị chuyển trạng thái, chúng ta có thể xác định được các phương trình theo các trạng thái như sau:

$$q_1 = q_1 0 + q_3 0 + \Lambda, \quad (4.6)$$

$$q_2 = q_1 1 + q_2 1 + q_3 1, \quad (4.7)$$

$$q_3 = q_2 0 \quad (4.8)$$

Thay (4.8) vào (4.7) chúng ta được

$$q_2 = q_1 1 + q_2 1 + (q_2 0) 1 = q_1 1 + q_2 (1 + 01)$$

Áp dụng định lý 4.1

$$q_2 = q_1 1 (1 + 01)^* \quad (4.9)$$

Tương tự $q_1 = q_1 0 + q_3 0 + \Lambda = q_1 0 + q_2 00 + \Lambda$. Thay (4.9) vào ta có

$$= q_1 0 + q_1 1 (1 + 01)^* 00 + \Lambda$$

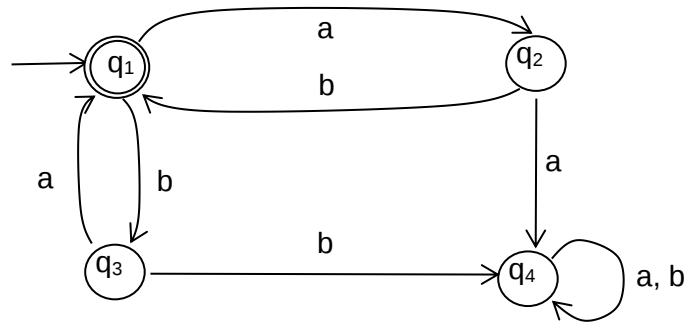
$$= q_1 (0 + 1 (1 + 01)^* 00) + \Lambda$$

Lại áp dụng định lý 4.1

$$q_1 = \Lambda (0 + 1 (1 + 01)^* 00)^* = (0 + 1 (1 + 01)^* 00)^*$$

Bởi vì, q_1 là trạng thái kết thúc nên biểu thức chính qui tương ứng với hệ chuyển trạng thái có đồ thị trạng thái như Hình H4-2 sẽ là: $(0 + 1(1 + 01)^*00)^*$.

Ví dụ 4.8 Cho trước ôtômát hữu hạn trạng thái M được mô tả bởi đồ thị chuyển trạng thái như hình H4-3. Chứng minh rằng: M đoán nhận tất cả các xâu trên $\{a, b\}$ có số các chữ a bằng số các chữ b và trong mọi dãy đầu của mỗi xâu thì hoặc số các chữ a hơn số các chữ b nhiều nhất là 1 hoặc số các chữ b hơn số các chữ a nhiều nhất là 1.



Hình H4-3 Ôtômát hữu hạn M

Giải: Áp dụng phương pháp nêu trên, chúng ta có hệ phương trình sau:

$$q_1 = q_2\mathbf{b} + q_3\mathbf{a} + \Lambda$$

$$q_2 = q_1\mathbf{a}$$

$$q_3 = q_1\mathbf{b}$$

$$q_4 = q_2\mathbf{a} + q_3\mathbf{b} + q_4(\mathbf{a} + \mathbf{b})$$

Bởi vì q_1 là trạng thái kết thúc, vậy ta sẽ giải hệ phương trình trên để tìm q_1 . Thay q_2, q_3 vào phương trình đầu tiên chúng được:

$$q_1 = q_1\mathbf{ab} + q_1\mathbf{ba} + \Lambda$$

Áp dụng định lý Arden, ta có

$$q_1 = \Lambda(\mathbf{ab+ba})^* = (\mathbf{ab} + \mathbf{ba})^*$$

Ôtômát và ngôn ngữ hình thức

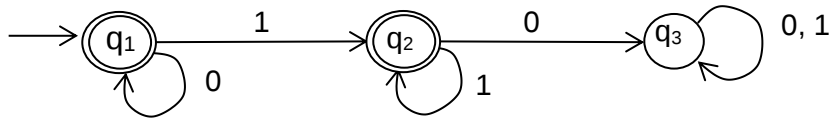
Đây chính là biểu thức chính qui biểu diễn cho tất cả các xâu đoán nhận được bởi M , Thật vậy, nếu dãy x được đoán nhận bởi M thì nó sẽ được tạo ra từ các dãy **ab** và **ba** ghép với nhau, trong đó số các chữ **a** luôn bằng số các chữ **b**.

Mọi tiếp đầu ngữ x_1 của xâu x thỏa mãn các tính chất sau:

- Có số các ký hiệu là chẵn vì chúng có số các chữ **a** luôn bằng số các chữ **b**,
- Hoặc có số các ký hiệu là lẻ thì ta có thể viết $x_1 = \mathbf{ay}$, hay $x_1 = \mathbf{by}$. Hiển nhiên y có số các ký hiệu là chẵn, nên y có số các chữ **a** luôn bằng số các chữ **b**.

Từ đó suy ra mọi dãy đầu của mỗi xâu được đoán nhận bởi M , thì hoặc số các chữ a hơn số các chữ b nhiều nhất là 1 hoặc số các chữ b hơn số các chữ a cũng nhiều nhất chỉ là 1.

Ví dụ 4.9 Tìm biểu thức chính qui đoán nhận bởi ôtômát



Hình H4-4 Ôtômát M có hai trạng thái kết thúc

Tương tự như trên, ta thiết lập được hệ phương trình sau:

$$q_1 = q_1\mathbf{0} + \Lambda$$

$$q_2 = q_1\mathbf{1} + q_2\mathbf{1}$$

$$q_3 = q_2\mathbf{0} + q_3(\mathbf{0} + \mathbf{1})$$

Áp dụng định lý 4.1 vào phương trình đầu tiên, ta thu được

$$q_1 = q_1\mathbf{0} + \Lambda = \mathbf{0}^*$$

Thay vào phương trình thứ hai

$$q_2 = q_1\mathbf{1} + q_2\mathbf{1} = \mathbf{0}^*\mathbf{1} + q_2\mathbf{1} = (\mathbf{0}^*\mathbf{1})\mathbf{1}^*$$

Phương trình ứng với q_3 không cần thiết phải giải vì nó không đạt đến được từ trạng thái kết thúc.

Vậy, biểu thức chính qui đoán nhận được bởi ôtômát M ở hình H4-4 là:

$$q_1 + q_2 = \mathbf{0}^* + (\mathbf{0}^*\mathbf{1})\mathbf{1}^* = \mathbf{0}^* + \mathbf{0}^*(\mathbf{11}^*) = \mathbf{0}^*(\Lambda + \mathbf{11}^*) = \mathbf{0}^*\mathbf{1}^* \text{ (ứng dụng } I_9)$$

Nói cách khác, các từ đoán nhận được bởi M là tất cả các xâu có các tiếp đầu ngữ gồm các chữ số 0 với số lượng bất kỳ sau đó là dãy các chữ số 1 tùy ý (có thể là rỗng).

4.3.2 Thiết lập ôtômát hữu hạn tương đương với biểu thức chính qui

Cho trước một biểu thức chính qui, chúng ta có thể xây dựng một ôtômát hữu hạn trạng thái tương đương, nghĩa là ngôn ngữ nó đoán nhận biểu diễn cho biểu thức chính qui cho trước (dành cho độc giả).

4.4 Bổ đề Bơm đối với các tập chính qui

Tiếp theo chúng ta xét điều kiện cần để kiểm tra xem một tập cho trước có phải là chính qui hay không. Đó chính là “*bổ đề Bơm*”, nó còn được sử dụng để chỉ ra những tập hợp không phải là tập chính qui.

Lưu ý: Mọi tập hữu hạn luôn là tập chính qui.

Định lý. (Bổ đề Bơm) Giả thiết $M = (Q, \Sigma, \delta, q_0, F)$ là ôtômát có n trạng thái và T là tập chính qui đoán nhận được bởi M . Giả sử $w \in T$, $|w| \geq m$. Nếu $m \geq n$, thì tồn tại x, y, z sao cho $w = xyz$ với $y \neq \Lambda$ và $xy^iz \in T$ với mọi $i \geq 0$. Đây chính là điều kiện cần để đoán nhận ngôn ngữ vô hạn là chính qui.

Chứng minh: Giả sử $w = a_1a_2 \dots a_m$, $m \geq n$

$$\delta(q_0, a_1a_2 \dots a_i) = q_i \text{ với } i = 1, 2, \dots, m; Q_1 = \{q_0, q_1, q_2, \dots, q_m\}$$

Trong đó, Q_1 là dãy các trạng thái tuần tự trên đường dẫn để sinh ra $w = a_1a_2 \dots a_m$. Vì M chỉ có n trạng thái khác nhau, vậy theo nguyên lý “*Chuồng bồ câu*” (nguyên lý Dirichlet) phải có ít nhất hai trạng thái của Q_1 trùng nhau. Trong số các cặp trùng nhau đó, ta chọn cặp đầu tiên, q_j và q_k ($q_j = q_k$). Không mất tính tổng quát chúng ta có thể giả thiết $0 \leq j < k \leq n$.

Xâu w do vậy có thể tách ra thành ba xâu con:

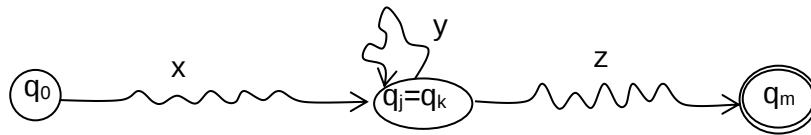
$$x = a_1a_2 \dots a_j$$

$$y = a_{j+1}a_{j+2} \dots a_k, \text{ vì } j < k \text{ nên } y \neq \Lambda.$$

$$z = a_{k+1}a_{k+2} \dots a_m$$

Bởi vì $k \leq n$ nên $|xy| \leq n$.

Đường dẫn để sinh ra w có thể minh họa như Hình H4-15.



Hình H4-5 Các xâu đoán nhận được bởi M

Dễ nhận thấy là từ trạng thái bắt đầu q_0 đi đến q_j xác định được x , duyệt tiếp theo y sẽ đạt được đến trạng thái q_k ($q_j = q_k$). Như vậy, nếu cứ tiếp tục duyệt theo y^i , với $i \geq 0$ sẽ vẫn đi đến q_j , sau đó có thể tiếp tục duyệt theo z để đi đến trạng thái kết thúc q_m . Điều này khẳng định $xy^iz \in L$ với mọi $i \geq 0$. ■

4.5 Ứng dụng của bổ đề “Bơm” (điều kiện cần của ngôn ngữ chính qui)

Định lý 4.5 có thể ứng dụng để kiểm chứng những tập cho trước không phải là tập chính qui. Để khẳng định một tập (vô hạn) không phải là chính qui, chúng ta có thể thực hiện theo thuật toán sau.

Thuật toán. Kiểm tra một tập không phải là chính qui

1. Giả sử L là chính qui và n là số trạng thái của ôtômát hữu hạn đoán nhận L .
2. Chọn một xâu $w \in L$ sao cho $|w| \geq n$. Sử dụng bổ đề Bơm để viết $w = xyz$, với $|xy| \leq n$ và $|y| > 0$.
3. Tìm một số nguyên i sao cho $xy^iz \notin L$. Điều này mâu thuẫn với giả thiết, từ đó suy ra L không phải là chính qui.

Ví dụ 4.10 Chứng minh rằng $L = \{a^{n^2} \mid n \geq 1\}$ không phải là tập chính qui.

Giải:

- (i) Giả sử L là chính qui và tồn tại ôtômát hữu hạn có n trạng thái để đoán nhận L .
- (ii) Đặt $w = a^{n^2}$. Hiển nhiên $|w| = n^2 > n$. Theo bổ đề Bơm chúng ta có thể viết $w = xyz$ với $|xy| \leq n$ và $|y| > 0$.
- (iii) Xét xy^2z ($i = 2$ trong thuật toán 4.4). Hiển nhiên, $|xy^2z| = |x| + 2|y| + |z| > |x| + |y| + |z|$ vì $|y| > 0$. Nghĩa là

$$n^2 = |xyz| = |x| + |y| + |z| < |xy^2z|.$$
Bởi vì $|xy| \leq n$ và $|y| \leq n$ nên

$$|xy^2z| = |x| + 2|y| + |z| = (|x| + |y| + |z|) + |y| \leq n^2 + n.$$

$$n^2 < |xy^2z| \leq n^2 + n < n^2 + n + n + 1 = (n + 1)^2$$

Vì $n^2 < |xy^2z| < (n+1)^2$, suy ra số các ký tự của xy^2z không thể là bình phương của một số nguyên, nghĩa là $xy^2z \notin L$, điều này mâu thuẫn với giả thiết.

Ví dụ 4.11 Chứng minh rằng $L = \{a^p \mid p \text{ là số nguyên tố}\}$ không phải là tập chính qui.

Giải:

- (i) Giả sử L là chính qui và tồn tại ôtômát hữu hạn có n trạng thái để đoán nhận L .
- (ii) Chọn p là số nguyên tố lớn hơn n . Đặt $w = a^p$. Hiển nhiên $|w| = p > n$. Theo bổ đề Bơm chúng ta có thể viết $w = xyz$ với $|xy| \leq n$ và $|y| > 0$. Các xâu x, y, z được thiết lập từ một phần tử a , nên có thể viết $y = a^m$ với $m \geq 1$ và $m \leq n$.
- (iii) Đặt $i = p + 1$. $|xy^iz| = |xyz| + |y^{i-1}| = p + (i - 1)m = p(1 + m)$. Theo bổ đề Bơm thì $xy^iz \in L$. Nhưng $|xy^iz| = p(1+m)$ và $p(n + 1)$ không phải là nguyên tố nên $xy^iz \notin L$, điều này mâu thuẫn với giả thiết.

Ví dụ 4.12 Ngôn ngữ $L = \{a^{2^n} \mid n \geq 1\}$ có phải là chính qui không?

Giải: Ta có thể viết $a^{2^n} = a(a^2)^i a$, với $i = n - 1$. Biết rằng $\{(a^2)^i \mid i \geq 0\}$ biểu diễn cho $(a^2)^*$ là chính qui. Vậy, ngôn ngữ L biểu diễn bởi biểu thức $a(P)^*a$, là chính qui, trong đó P biểu diễn cho tập chính qui $\{a^2\}$.

4.6 Các tập chính qui và văn phạm chính qui

Chúng ta đã chứng minh được rằng các tập chính qui là đoán nhận được bởi ôtômát hữu hạn đơn định. Bây giờ, chúng ta chỉ ra rằng lớp các tập chính qui trên Σ cũng chính là các ngôn ngữ chính qui trên bảng chữ Σ .

4.6.1 Xây dựng văn phạm chính qui tương đương với ÔTĐĐ cho trước.

Cho trước $M = (Q, \Sigma, \delta, q_0, F)$ là ôtômát đơn định hữu hạn trạng thái. Vì Q là tập hữu hạn nên ta có thể viết $Q = \{q_0, q_1, q_2, \dots, q_n\}$. Nếu w là một từ trong $T(M)$ thì nó sẽ nhận được từ các phép ghép các nhãn của một số các phép chuyển trạng thái từ trạng thái q_0 đến một trạng thái kết thúc nào đó trong F .

Do đó chúng ta xây dựng văn phạm G để sinh ra $L(M)$ như sau:

Cho trước $M = (Q, \Sigma, \delta, q_0, F)$, với $Q = \{q_0, q_1, q_2, \dots, q_n\}$.

Ta xây dựng Văn phạm tương ứng G như sau:

Đặt $G = (\Sigma, V, P, S)$, với $V = Q$; $S = q_0$ và P gồm các luật sinh:

- (i) Nếu $\delta(q, a) = p \notin F \quad \forall a \in \Sigma, p, q \in Q$, thì $(q \rightarrow ap) \in P$

(ii) Nếu $\delta(q, a) = p \in F$, thì $(q \rightarrow ap) \in P$ và $(q \rightarrow a) \in P$

Thật vậy, theo cách thành lập P thì:

$q \Rightarrow ap$ khi và chỉ khi $\delta(q, a) = p$

$q \Rightarrow a$ khi và chỉ khi $\delta(q, a) = p \in F$

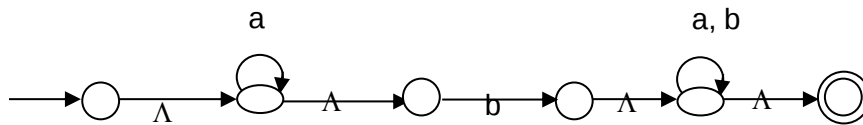
Do vậy, $q_0 \Rightarrow a_1q_1 \Rightarrow a_1a_2q_2 \Rightarrow \dots \Rightarrow a_1a_2\dots a_{k-1}q_k \Rightarrow a_1a_2\dots a_{k-1}a_k$ khi và chỉ khi

$\delta(q_0, a_1) = q_1, \delta(q_1, a_2) = q_2, \delta(q_2, a_3) = q_3, \dots, \delta(q_{k-1}, a_k) \in F$.

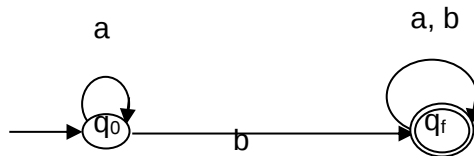
Điều này khẳng định $w = a_1a_2\dots a_k \in L(G)$ khi và chỉ khi $\delta(q_0, a_1a_2\dots a_k) \in F$, nghĩa là khi và chỉ khi $w \in T(M)$.

Ví dụ 4.13 Cho tập chính qui được biểu diễn bởi $E = \mathbf{a^*b(a+b)^*}$. Xây dựng văn phạm G để $L(G) = E$.

Giải: Trước tiên chúng ta xây dựng ôtômát hữu hạn M đoán nhận E.



Chúng ta được ôtômát đơn định hữu hạn như ở hình sau:



Hình H4-6 Ôtômát hữu hạn M đoán nhận $\mathbf{a^*b(a+b)^*}$

Đặt $G = (\{a, b\}, \{A_0, A_1\}, P, A_0)$, trong đó P gồm:

$A_0 \rightarrow aA_0, A_0 \rightarrow bA_1, A_0 \rightarrow b, A_1 \rightarrow aA_1, A_1 \rightarrow bA_1, A_1 \rightarrow a, A_1 \rightarrow b$

Hiển nhiên, G là văn phạm chính qui, sinh ra ngôn ngữ chính qui và ngôn ngữ đó cũng chính là tập chính qui E.

4.6.2 Xây dựng ÔTĐĐ hữu hạn tương đương với văn phạm chính qui G

Cho văn phạm chính qui $G = (\Sigma, \{A_0, A_1, A_2, \dots, A_n\}, P, A_0)$. Ôtômát hữu hạn $M = (\{A_0, A_1, A_2, \dots, A_n, q_f\}, \Sigma, \delta, A_0, \{q_f\})$ tương đương được xây dựng như sau:

- (i) Tập các trạng thái là tập các biến (ký hiệu chưa kết thúc),
- (ii) Trạng thái khởi đầu là A_0 ,
- (iii) Nếu $A_i \rightarrow aA_j \in P$, thì $\delta(A_i, a) = A_j \notin F$ và
Nếu $A_i \rightarrow a \in P$, thì $\delta(A_i, a) = q_f$.

Từ cách xây dựng như trên dễ nhận thấy:

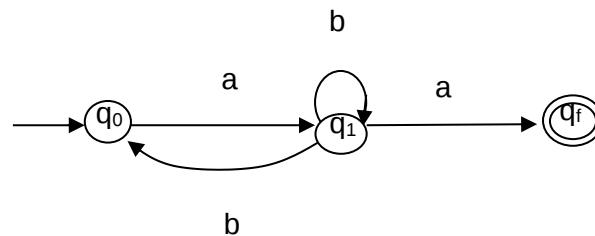
$A_0 \Rightarrow a_1A_1 \Rightarrow a_1a_2A_2 \Rightarrow \dots \Rightarrow a_1a_2\dots a_{k-1}A_k \Rightarrow a_1a_2\dots a_{k-1}a_k$ trong văn phạm G khi và chỉ khi tồn tại một đường dẫn trong đồ thị M đi từ A_0 đến trạng thái kết thúc q_f để có được $a_1a_2\dots a_{k-1}a_k$. Từ đó suy ra $L(G) = T(M)$.

Ví dụ 4.14 Cho trước $G = (\{a, b\}, \{A_0, A_1\}, P, A_0)$, trong đó

$$P = \{A_0 \rightarrow aA_1, A_1 \rightarrow bA_1, A_1 \rightarrow a, A_1 \rightarrow bA_0\}$$

Xây dựng ôtômát hữu hạn M để đoán nhận $L(G)$.

Giải: Đặt $M = (\{q_0, q_1, q_f\}, \{a, b\}, \delta, q_0, \{q_f\})$, trong đó q_0, q_1 là hai trạng thái tương ứng với các biến A_0, A_1 và q_f là trạng thái kết thúc mới được bổ sung. Theo cách xây dựng như trên chúng ta có ôtômát M như sau:



Hình H4-7 Ôtômát đoán nhận $L(G)$

Dễ nhận ra $L(G) = a(ba + b)^*a$.

Bài tập:

4.1. Xây dựng ôtômát đơn định trên $\{0, 1\}$ để đoán nhận tất cả các xâu chứa chẵn lần các chữ số 1.

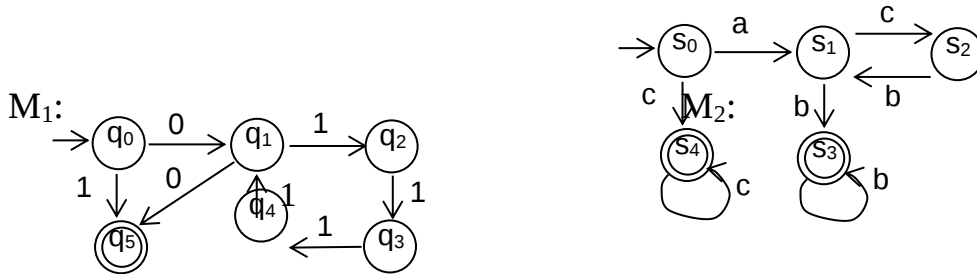
4.2. Xây dựng ôtômát đơn định trên $\Sigma = \{a_1, a_2, \dots, a_n\}$ để đoán nhận các ngôn ngữ sau:

d) $L_1 = \{x \in \Sigma^* \mid x \text{ có độ dài lẻ}\}$

e) $L_2 = \{x \in \Sigma^* \mid x \text{ có độ dài chẵn}\}$

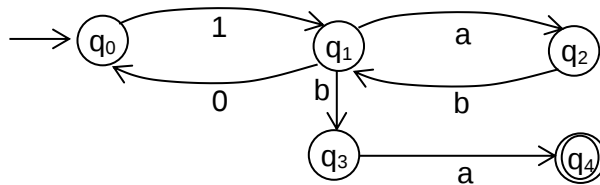
f) $L_3 = \{x \in \Sigma^* \mid x \text{ có độ dài chẵn và lớn hơn } 1\}$

4.3. Cho hai ôtômát sau đây:



Tìm ngôn ngữ $T(M_1)$ đoán nhận được bởi M_1 và $T(M_2)$ đoán nhận được bởi M_2

4.4* Cho trước ôtômát M được biểu diễn dưới dạng đồ thị chuyển trạng thái như hình sau:



a/ Tìm ngôn ngữ đoán nhận được bởi M

b/ Xây dựng văn phạm chính qui G để sinh ra ngôn ngữ được đoán nhận bởi ôtômát nêu trên.

c/ Xâu $w = 1abba$ có được sinh ra trong G hay không?, Tại sao?

4.5 Biểu diễn các tập sau bằng các biểu thức chính qui

(i) $\{0, 1, 2\}$

(ii) $\{w \in \{a, b\}^* \mid w \text{ chỉ chứa một lần xuất hiện } a\}$

(iii) $\{a^2, a^5, a^8, \dots\}$

(iv) $\{a^n \mid n \text{ chia hết cho } 2, 3 \text{ hoặc } n = 5\}$

(v) Tập tất cả các xâu trên $\{a, b\}$ mà bắt đầu và kết thúc bằng a .

4.6 Tìm tất cả các xâu có độ dài bằng hoặc nhỏ hơn 5 trong tập chính qui được biểu diễn bằng:

(i) $(ab + a)^*(aa + b)$

(ii) $(a^*b + b^*a)^*a$

(iii) $a^* + (ab + a)^*$

4.7 Xây dựng ôtômát hữu hạn tương ứng với các biểu thức chính qui sau:

(i) $(ab + c^*)^*b$

(ii) $a + bb + bab^*a$

(iii) $(a + b)^*abb$

4.8 Tìm các biểu thức chính qui biểu diễn cho các tập sau:

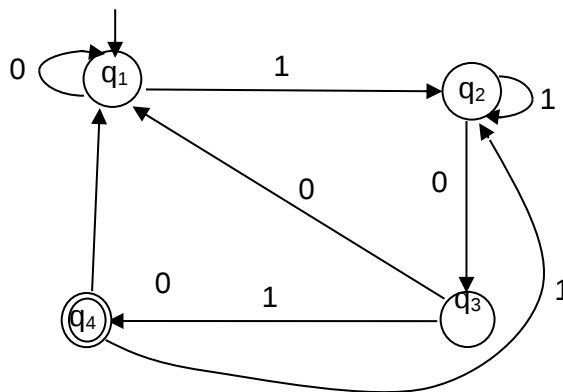
(a) Tập tất cả các xâu trên $\{0, 1\}$ có nhiều nhất một cặp số 0 hoặc nhiều nhất một cặp số 1.

(b) Tập tất cả các xâu trên $\{a, b\}$, trong đó số các lần xuất hiện của a chia hết cho 3.

(c) Tập tất cả các xâu trên $\{a, b\}$, trong đó có ít nhất hai lần xuất hiện của b giữa hai ký tự a .

(d) Tập tất cả các xâu trên $\{a, b\}$ kết thúc bằng 00 và bắt đầu bằng 1.

4.9 Tìm biểu thức chính qui tương ứng với đồ thị trạng thái sau



4.10 Chứng minh rằng $L = \{ww \mid w \in \{a, b\}^*\}$ không phải là tập chính qui.

4.11 Chứng minh rằng các tập sau không phải là tập chính qui

(i) $\{a^n b^{2n} \mid n > 0\}$

(ii) $\{a^n b^m \mid 0 < n < m\}$

(iii) $\{a^n b^n \mid n > 0\}$

(iv) $\{a^n b^m \mid n, m \text{ là nguyên tố cùng nhau, } \text{UCLN}(m, n) = 1\}$

4.12 Cho trước văn phạm G có các qui tắc dẫn xuất $S \rightarrow aS \mid a$, tìm ôtômát M để đoán nhận $L(G)$.

4.13 Xây dựng một ôtômát hữu hạn để đoán nhận $L(G)$, khi G có $P: S \rightarrow aS \mid bA \mid b, A \rightarrow aA \mid bS \mid a$

4.14 Xây dựng ôtômát đơn định hữu hạn tương đương với văn phạm có các luật sinh sau: $S \rightarrow aS \mid aA, A \rightarrow bB, B \rightarrow aC, C \rightarrow \Lambda$

4.15 Cho ngôn ngữ chính qui trên bảng chữ cái $\{0, 1\}$, gồm tất cả các từ chứa hai ký hiệu 0 đứng liền nhau.

(i) Hãy biểu diễn ngôn ngữ đó bởi biểu thức chính qui

(ii) Hãy tìm văn phạm chính qui sinh ra ngôn ngữ đó

(iii) Hãy xây dựng ôtômát hữu hạn đoán nhận cùng ngôn ngữ trên.

4.16 Xây dựng ôtômát hữu hạn đơn định tương đương với biểu thức chính qui $10+(0+11)^*1$.

4.17 Cho ngôn ngữ L trên bảng chữ $\{0, 1\}$ có các từ chứa chẵn lần số 0 và chẵn lần số 1.

(i) Tìm biểu thức chính qui biểu diễn cho L .

(ii) Xây dựng văn phạm sinh ra ngôn ngữ L .

4.18* Cho ngôn ngữ $L = \{x^n(10)^m y^k \mid n, k \geq 1, m \geq 0\}$

(i) Hãy tìm văn phạm chính qui G sinh ra L ($L(G) = L$).

(ii) Xây dựng ôtômát hữu hạn trạng thái M sao cho $T(M) = L(G)$.

(iii) Hãy kiểm tra xem các xâu $xx10y, x101$ có được sinh ra bởi văn phạm G và được đoán nhận bởi ôtômát M ở trong câu (i), (ii)?

4.19* Cho văn phạm chính qui $G = (\Sigma, V, P, S)$, với tập các qui tắc

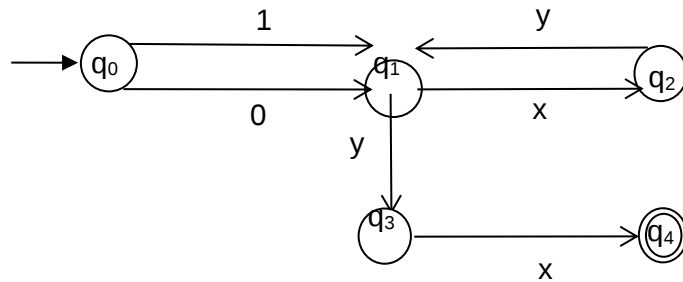
$$P = \{S \rightarrow aA \mid bE, A \rightarrow xB \mid b, B \rightarrow yC, C \rightarrow zA, E \rightarrow aF, F \rightarrow b\}$$

(i) Tìm ngôn ngữ $L(G)$ sinh ra bởi G .

(ii) Xây dựng ôtômát hữu hạn trạng thái M sao cho $T(M) = L(G)$.

(iii) Cho xâu $w = axyzb$. Hãy chỉ ra G sinh ra xâu w và w còn được đoán nhận bởi M .

4.20* Cho ôtômát hữu hạn M với hàm chuyển trạng thái được cho như hình sau:



(i) Xây dựng văn phạm chính qui G để $L(G) = T(M)$.

(ii) Hãy chỉ ra G sinh ra $w_1 = 1xyyx$, $w_2 = 0yx$ và M cũng đoán nhận được w_1 , w_2 .

4.21* Cho trước bảng chữ cái $\Sigma = \{a, b\}$.

(i) Xây dựng ôtômát hữu hạn trạng thái M trên Σ để đoán nhận tất cả các xâu có các chữ a là số chẵn và lớn hơn 0.

(ii) Xây dựng văn phạm chính qui tương đương với M mà anh (chị) đã chỉ ra.

(iv) Xác định biểu thức chính qui tương đương với ngôn ngữ sinh bởi G .

CHƯƠNG 5. NGÔN NGỮ PHI NGỮ CẢNH VÀ ÔTÔMÁT ĐẨY XUỐNG

Nội dung của Chương 5 đề cập đến:

- ✓ Cây dẫn xuất và các phương pháp xác định văn phạm phi ngữ cảnh,
- ✓ Tính nhập nhằng của văn phạm phi ngữ cảnh,
- ✓ Bổ đề Bơm đối với Ngôn ngữ phi ngữ cảnh,
- ✓ Ôtômát đẩy xuống,
- ✓ Hai kiểu đoán nhận ngôn ngữ của ôtômát đẩy xuống,
- ✓ Các kết quả đoán nhận của ôtômát đẩy xuống và ngôn ngữ phi ngữ cảnh.

5.1 Ngôn ngữ phi ngữ cảnh và cây dẫn xuất

Văn phạm $G = (\Sigma, V, P, S)$ là phi ngữ cảnh nếu mọi luật sinh đều có dạng:

$$A \rightarrow \alpha, \text{ trong đó } A \in V \text{ và } \alpha \in (V \cup \Sigma)^*$$

Ví dụ 5.1 Xây dựng văn phạm phi ngữ cảnh G sinh ra tất cả các số nguyên.

Giải. Giả sử $G = (\Sigma, V, P, S)$, trong đó

$$V = \{S, \langle \text{dau} \rangle, \langle \text{chu so} \rangle, \langle \text{So nguyen} \rangle\}$$

$$\Sigma = \{0, 1, 2, \dots, 9, +, -\}$$

$$P: \quad S \rightarrow \langle \text{dau} \rangle \langle \text{So nguyen} \rangle,$$

$$\langle \text{dau} \rangle \rightarrow + \mid -,$$

$\langle \text{Số nguyên} \rangle \rightarrow \langle \text{chu số} \rangle \langle \text{Số nguyên} \rangle \mid \langle \text{chu số} \rangle$

$\langle \text{chu số} \rangle \rightarrow 0 \mid 1 \mid 2 \mid \dots \mid 9.$

Kiểm chứng được rằng: $L(G)$ là tập tất cả các số nguyên.

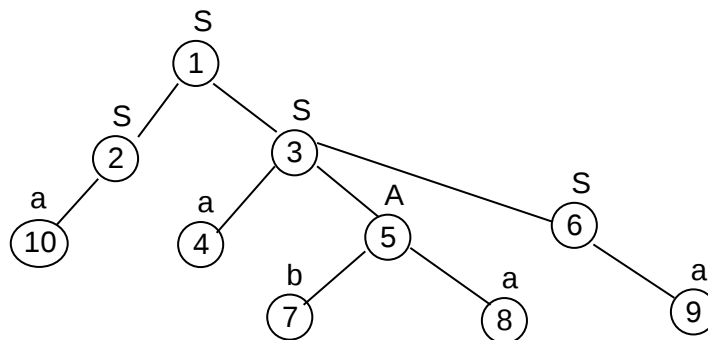
Định nghĩa. Cây dẫn xuất (còn gọi là *cây phân tích cú pháp*, *cây cú pháp*) cho văn phạm phi ngữ cảnh $G = (\Sigma, V, P, S)$, là cây thoả mãn các tính chất sau:

- (i) Các đỉnh đều được gán nhãn là: biến, ký tự kết thúc hoặc ϵ .
- (ii) Gốc có nhãn S ,
- (iii) Nếu có 1 nút có nhãn là A là nút trong, thì A là một biến.
- (iv) Nếu nút (đỉnh) n có nhãn là A và các nút $n_1, n_2, \dots, n_k$ là các con của n , theo thứ tự từ trái qua phải và lần lượt mang các nhãn $X_1, X_2, \dots, X_k$ tương ứng, thì $A \rightarrow X_1 X_2 \dots X_k$ phải là một luật sinh trong P .
- (v) Đỉnh n là lá nếu nhãn của nó là $a \in \Sigma$ hoặc là ký hiệu rỗng ϵ .

Ví dụ 5.2 Cho văn phạm $G = (\{a, b\}, \{S, A\}, P, S)$, với P :

$S \rightarrow aAS \mid a \mid SS, A \rightarrow SbA \mid ba$

Cây dẫn xuất sinh ra xâu $w = aabaa$ trong G sẽ là:



Hình H5-1 Cây dẫn xuất cho văn phạm G sinh ra xâu $w = aabaa$

Sắp xếp các đỉnh (lá) từ trái qua phải

+ Các đỉnh con của đỉnh gốc được sắp xếp từ trái qua phải, nghĩa là các đỉnh ở mức 1 được sắp xếp từ bên trái,

+ Nếu hai đỉnh v_1, v_2 ở mức 1 và v_1 ở bên trái của v_2 thì ta nói rằng v_1 bên trái của tất cả các đỉnh con của v_2 và các đỉnh con của v_1 cũng là ở bên trái của v_2 , đồng thời cũng ở bên trái của các đỉnh con của v_2 .

+ Lặp lại quá trình trên cho mức 2, 3, ..., k.

Định nghĩa. Kết quả của một cây dẫn xuất là dãy các nhãn được ghép của các lá từ trái qua phải.

Ví dụ: Kết quả của cây dẫn xuất ở Hình H5-1 là *aabaa*.

Sử dụng các luật sinh của văn phạm G:

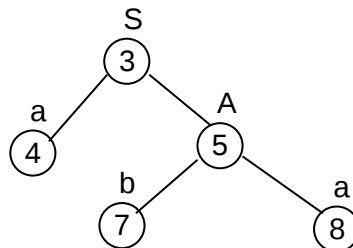
$$S \Rightarrow SS \Rightarrow aS \Rightarrow aaAS \Rightarrow aabaS \Rightarrow aabaa$$

Vậy kết quả của một cây dẫn xuất cũng là một từ được sinh ra bởi văn phạm G.

Định nghĩa. Cây con của cây dẫn xuất T là một cây có

- (i) Gốc của nó là một đỉnh v của cây T ,
- (ii) Các đỉnh của nó cũng là con cháu của v cùng với các nhãn tương ứng,

Ví dụ: Cây trong hình H5-2 là cây con của cây ở hình H5-1



Hình H5-2 Một cây con của cây T ở hình H5-1

Lưu ý: Cây con cũng là một cây dẫn xuất, chỉ khác nhau là nhãn của cây con đó có thể không phải là ký hiệu bắt đầu S . Nếu nhãn của gốc của cây con là A thì cây con đó được gọi là A -cây.

Định lý. Giả sử $G = (\Sigma, V, P, S)$ là văn phạm phi ngữ cảnh. Khi đó

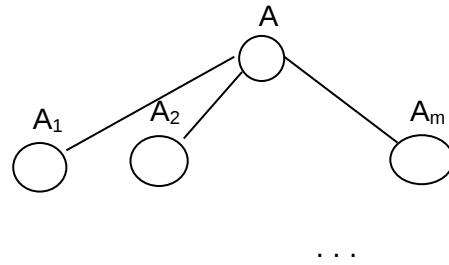
$$S \xRightarrow{*} \alpha \text{ khi và chỉ khi tồn tại cây dẫn xuất } T \text{ của } G \text{ để } T\text{-cây có kết quả là } \alpha.$$

Chứng minh:

Chỉ cần chứng minh rằng $A \xRightarrow{*} \alpha$ khi và chỉ khi A -cây có kết quả là α . (5.1)

Giả sử α là kết quả A-cây T_1 . Chúng ta chứng minh $A \Rightarrow^* \alpha$ bằng phương pháp qui nạp theo số các đỉnh bên trong của T_1 .

Cơ sở qui nạp: Nếu T_1 chỉ có một đỉnh bên trong, nghĩa là tất cả các đỉnh con của gốc đều là lá. Khi đó cây T_1 có dạng:



Hình H5-3 (a) A-Cây có một đỉnh bên trong

Theo (iv) trong định nghĩa trên suy ra $A \rightarrow A_1A_2...A_m = \alpha$ là một qui tắc trong G . Vậy $A \Rightarrow \alpha$.

Giả thiết qui nạp: Giả sử (5.1) đúng với các cây con có $k - 1$ đỉnh bên trong, $k > 1$.

Chúng ta xét A-cây T_1 có k đỉnh bên trong ($k \geq 2$). Giả thiết gốc của T_1 có $v_1, v_2, \dots, v_m$ là các đỉnh con được gán nhãn tương ứng $X_1, X_2, \dots, X_m$. Theo (iv) trong định nghĩa, suy ra $A \rightarrow X_1X_2...X_m$ là một qui tắc trong P . Do vậy,

$$A \Rightarrow X_1X_2...X_m \quad (5.2)$$

Bởi vì $k \geq 2$ nên trong số các đỉnh con của A có ít nhất một đỉnh là đỉnh bên trong. Nếu v_i là một đỉnh bên trong thì lại xét tiếp cây con của T_1 có gốc chính là v_i . Hiển nhiên là số các đỉnh bên trong của cây con gốc v_i là nhỏ hơn k , vậy theo giả thiết qui nạp suy ra $X_i \Rightarrow^* \alpha_i$, X_i là nhãn của v_i .

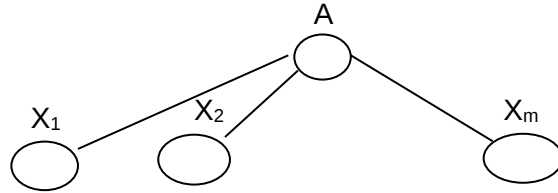
Nếu v_i không phải là đỉnh bên trong, thì $X_i = \alpha_i$. Và chúng ta nhận được:

$$A \Rightarrow X_1X_2...X_m \xRightarrow{*} \alpha_1X_2...X_m \dots \xRightarrow{*} \alpha_1\alpha_2 \dots \alpha_m = \alpha$$

Nghĩa là $A \Rightarrow^* \alpha$.

Để chứng minh chiều ngược lại, chúng ta giả thiết rằng $A \Rightarrow^* \alpha$. Chúng ta đi xây dựng A-cây để có kết quả là α . Thực hiện điều này bằng phương pháp qui nạp theo số bước trong dẫn xuất $A \Rightarrow^* \alpha$.

Khi $A \Rightarrow \alpha$, $A \rightarrow \alpha$ là qui tắc trong P. Nếu $\alpha = X_1X_2...X_m$, A-cây cho kết quả α có thể xây dựng dễ dàng và có dạng như hình H5-3 (b).



Hình H5-3 (b) Cây dẫn xuất cho dẫn xuất một bước

Giả sử dẫn xuất $A \Rightarrow^* \alpha$ thực hiện qua k bước, $k > 1$. Khi đó ta có thể chia nó thành

$A \Rightarrow X_1X_2...X_m \xRightarrow{k-1} \alpha$. Dẫn xuất $A \Rightarrow X_1X_2...X_m$ kéo theo $A \rightarrow X_1X_2...X_m$ là một qui tắc trong P. Còn trong dẫn xuất $X_1X_2...X_m \xRightarrow{k-1} \alpha$ thì:

- (i) Hoặc X_i không chuyển trạng thái suốt trong quá trình dẫn xuất, $X_i = \alpha_i$
- (ii) Hoặc X_i biến đổi trong quá trình dẫn xuất. Giả sử α_i là xâu nhận được từ X_i , $X_i \Rightarrow^* \alpha_i$.

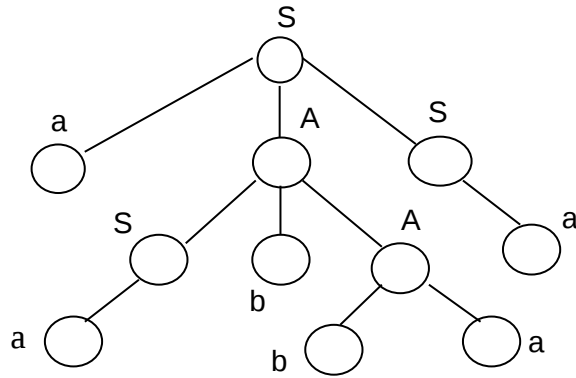
Cây dẫn xuất cho kết quả $\alpha = \alpha_1\alpha_2 \dots \alpha_m$ được xây dựng như sau:

Khi $A \rightarrow X_1X_2...X_m$ là một qui tắc trong P, ta xây dựng cây có m lá: $X_1, X_2, \dots, X_m$. Từ X_i suy dẫn ra α_i được thực hiện nhỏ hơn k bước, vậy theo giả thiết qui nạp thì X_i -cây sẽ cho kết quả là α_i . Do đó, A-cây sẽ cho kết quả là α . ■

Ví dụ 5.3 Xét văn phạm với P: $S \rightarrow aAS \mid a$, $A \rightarrow SbA \mid SS \mid ba$. Chỉ ra rằng $S \xRightarrow{*} aabbbaa$ và tìm cây dẫn xuất để cho kết quả $aabbbaa$.

Giải: $S \Rightarrow aAS \Rightarrow aSbAS \Rightarrow aabAS \Rightarrow a^2bbaS \Rightarrow a^2b^2a^2$, nghĩa $S \xRightarrow{*} a^2b^2a^2$.

Cây dẫn xuất cho kết quả trên là:



Hình H5-4 Cây dẫn xuất cho kết quả $a^2b^2a^2$

Trong các dẫn xuất ở trên chúng ta nhận thấy, luật sinh P luôn áp dụng cho biến trái nhất.

Định nghĩa. Dẫn xuất $A \Rightarrow^* w$ là *dẫn xuất trái nhất* nếu chỉ áp dụng luật sinh cho biến trái nhất trong mọi bước dẫn xuất.

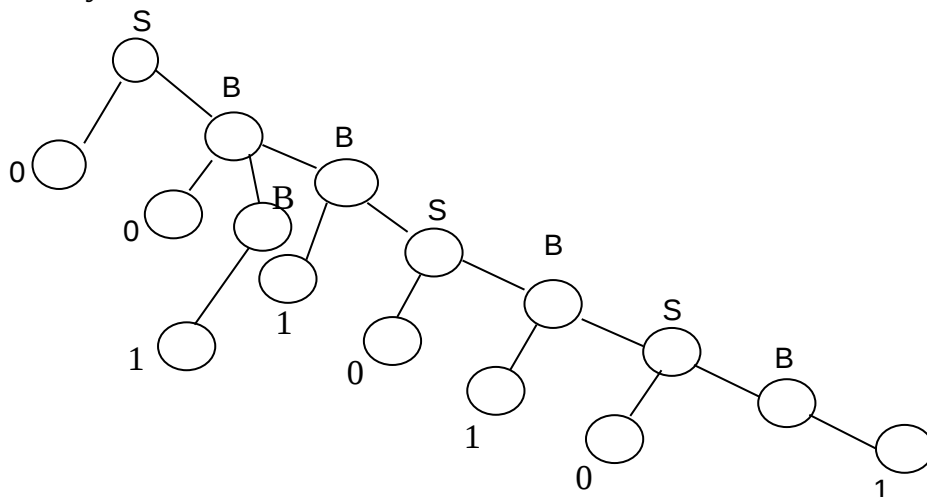
Định nghĩa. Dẫn xuất $A \Rightarrow^* w$ là *dẫn xuất phải nhất* nếu chỉ áp dụng luật sinh cho biến phải nhất trong mọi bước dẫn xuất.

Ví dụ 5.4 Cho văn phạm G có P: $S \rightarrow 0B \mid 1A$, $A \rightarrow 0 \mid 0S \mid 1AA$, $B \rightarrow 1 \mid 1S \mid 0BB$ và $w = 00110101$.

- (i) Tìm dẫn xuất trái nhất của w,
- (ii) Tìm dẫn xuất phải nhất của w,
- (iii) Xác định cây dẫn xuất của w.

Giải:

- (i) $S \Rightarrow 0B \Rightarrow 00BB \Rightarrow 0011S \Rightarrow 0^21^20B \Rightarrow 0^21^201S \Rightarrow 0^21^2010B \Rightarrow 0^21^20101$
- (ii) $S \Rightarrow 0B \Rightarrow 00BB \Rightarrow 00B1S \Rightarrow 0^2B10B \Rightarrow 0^2B101S \Rightarrow 0^2B1010B \Rightarrow 0^21B10101 \Rightarrow 0^21^20101$
- (iii) Cây dẫn xuất cho 0^21^20101 là



Hình H5-5 Cây dẫn xuất cho kết quả 0^21^20101

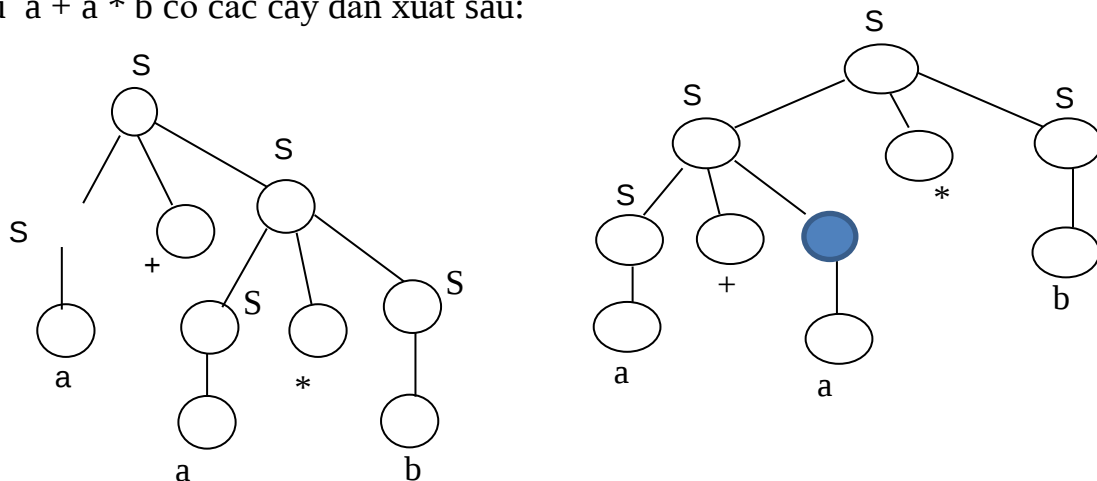
5.2 Sự nhập nhằng trong văn phạm phi ngữ cảnh

Định nghĩa. Văn phạm phi ngữ cảnh G là nhập nhằng nếu có $w \in L(G)$ là nhập nhằng (có nhiều hơn 01 cây dẫn xuất trái).

(Một câu được hiểu theo nhiều hơn 01 nghĩa.)

Ví dụ 5.5 Xét $G = (\{a, b, +, *\}, \{S\}, P, S)$ với $P: S \rightarrow S + S \mid S * S \mid a \mid b$.

Xâu $a + a * b$ có các cây dẫn xuất sau:



Hình H5-6 Hai cây dẫn xuất cùng cho kết quả $a + a * b$

5.3 Dạng chuẩn Chomsky

Định nghĩa Văn phạm phi ngữ cảnh G ở *dạng chuẩn Chomsky* (viết tắt là CNF) nếu các qui tắc đều có dạng:

$A \rightarrow a$ hoặc $A \rightarrow BC$, và $S \rightarrow \Lambda$ nếu $\Lambda \in L(G)$, với $S, A, B, C \in V_N, a \in \Sigma$.

Khi Λ thuộc $L(G)$ thì S không xuất hiện ở vế phải của mọi qui tắc.

Ví dụ 5.6: Văn phạm G có các qui tắc $S \rightarrow AB \mid \Lambda, A \rightarrow a, B \rightarrow b$ là ở dạng CNF.

Nhận xét: Ở dạng CNF, các vế phải của mọi qui tắc chỉ có thể là một ký hiệu kết thúc (hoặc ký hiệu rỗng Λ) hoặc hai biến.

Vấn đề quan trọng đặt ra là có thể chuyển một văn phạm phi ngữ cảnh bất kỳ về CNF hay không? Ta có Định lý sau (không chứng minh):

Định lý (Lược giản về dạng chuẩn Chomsky). Với mọi văn phạm phi ngữ cảnh G , đều tồn tại văn phạm G_2 tương đương ở dạng chuẩn Chomsky.

5.4 Bổ đề Bơm cho ngôn ngữ phi ngữ cảnh

Có nhiều cách để chứng minh một ngôn ngữ cho trước có là phi ngữ cảnh hay không. Một trong các cách đó là xây dựng một văn phạm phi ngữ cảnh (hay xây dựng ôtômát đẩy xuống) để sinh ra (đoán nhận) ngôn ngữ đó.

Song, muốn chứng minh một ngôn ngữ không phải là phi ngữ cảnh thì khó hơn nhiều. Ta không thể xét được tất cả các văn phạm phi ngữ cảnh để chứng tỏ ngôn ngữ đã cho không được sinh bởi các văn phạm phi ngữ cảnh. Để khắc phục khó khăn này, người ta đưa ra một số “tiêu chuẩn đặc trưng” của ngôn ngữ phi ngữ cảnh và dựa vào đó để xem xét khả năng thỏa mãn của ngôn ngữ cho trước.

Sau đây chúng ta xét một số tính chất quan trọng của ngôn ngữ phi ngữ cảnh dựa vào bổ đề Bơm. Theo đó, chúng ta có thể khẳng định khá nhiều ngôn ngữ không phải là phi ngữ cảnh.

Bổ đề. Giả sử G là văn phạm phi ngữ cảnh ở dạng CNF và T là một cây dẫn xuất của G . Nếu độ dài của đường đi dài nhất trong T nhỏ hơn hoặc bằng k thì mọi kết quả sinh ra của T đều có độ dài nhỏ hơn hoặc bằng 2^{k-1} .

Chứng minh: Qui nạp theo độ dài k , độ dài của đường đi dài nhất trong số tất cả các *A-cây dẫn xuất* (cây dẫn xuất với gốc có nhãn là A).

Bước cơ sở qui nạp: Khi $k = 1$, tức là gốc của cây dẫn xuất chỉ có một đỉnh con mà nhãn của nó là ký hiệu kết thúc. Nếu đỉnh gốc có 2 đỉnh con thì nhãn của chúng sẽ phải là biến bởi G là ở dạng chuẩn Chomsky. Từ đó suy ra kết quả sinh ra có độ dài là $1 \leq 2^{1-1} = 1$.

Bước giả thiết qui nạp: Giả sử kết luận trên đúng với mọi $k - 1$ ($k > 1$).

Ta cần chứng minh nó đúng với k , tức nếu T là *A-cây dẫn xuất* có các đường đi dài nhất nhỏ hơn hoặc bằng k thì mọi kết quả sinh ra của T đều có độ dài nhỏ hơn hoặc bằng 2^{k-1} . Bởi vì $k > 1$ nên đỉnh gốc của T có đúng hai cây con A_1 và A_2 có các đường đi dài nhất là nhỏ hơn hoặc bằng $k - 1$.

Theo giả thiết qui nạp mọi kết quả w_1, w_2 sinh ra trên hai cây dẫn xuất A_1 và A_2 tương ứng đều thỏa mãn $|w_1| \leq 2^{k-2}, |w_2| \leq 2^{k-2}$. Kết quả sinh ra bởi T khi đó có thể là w_1w_2 và $|w_1w_2| \leq 2^{k-2} + 2^{k-2} = 2^{k-1}$. Đó là điều cần phải chứng minh. ■

Định lý. Nếu L là ngôn ngữ phi ngữ cảnh thì sẽ tồn tại số tự nhiên n sao cho:

- (i) $\forall z \in L$ với $|z| \geq n$ đều có thể viết được dưới dạng: $z = uvwxy$, với u, v, x, y là các xâu nào đó
- (ii) $|vx| \geq 1$
- (iii) $|vwx| \leq n$
- (iv) $uv^kwx^ky \in L$ với mọi $k \geq 0$.

Chứng minh:

Từ hệ quả 5.2 ta có thể kiểm tra được $\Lambda \in L$ hay không. Khi $\Lambda \in L$ thì ta xây dựng văn phạm phi ngữ cảnh $G = (V_N, \Sigma, P, S)$ ở dạng chuẩn Chomsky để sinh ra $L - \{\Lambda\}$ (trường hợp $\Lambda \notin L$ thì xây dựng văn phạm $G = (V_N, \Sigma, P, S)$ ở dạng chuẩn Chomsky để sinh ra L).

Giả sử $|V_N| = m$ và đặt $n = 2^m$. Ta chỉ ra rằng n là số cần tìm ở bổ đề trên.

Với $z \in L, |z| \geq n$ ta đi xây dựng cây dẫn xuất cho z . Nếu độ dài đường đi dài nhất trong T nhiều nhất là m thì theo bổ đề 5.3, ta có $|z| \leq 2^{m-1}$. Nhưng $|z| \geq n = 2^m > 2^{m-1}$. Vậy T sẽ có đường đi, ví dụ Γ , có độ dài lớn hơn hoặc bằng $m + 1$. Γ có ít nhất $m + 2$ đỉnh và chỉ có đỉnh cuối cùng là lá. Như thế trên Γ sẽ có $m + 1$ đỉnh có nhãn là các biến. Bởi vì $|V_N| = m$ nên một số nhãn sẽ phải lặp lại.

Chúng ta chọn đỉnh có nhãn bị lặp lại như sau: bắt đầu từ lá đi ngược lên đến khi gặp đỉnh đầu tiên có nhãn bị lặp lại, ví dụ đó là nhãn B . Giả sử v_1 và v_2 là hai đỉnh có cùng nhãn B , với v_1 ở gần gốc của cây T hơn. Trên đoạn đường đi Γ từ v_1 đến lá chỉ chứa đúng một đỉnh có nhãn B bị lặp 1 lần, do vậy độ dài lớn nhất của đoạn đó là $m + 1$.

Giả sử T_1 và T_2 là hai cây con tương ứng với hai đỉnh gốc v_1 và v_2 , T_1 sinh ra z_1 và T_2 sinh ra w . Theo bổ đề trên, suy ra $|z_1| \leq 2^m$.

Bởi vì z là xâu được sinh ra bởi cây T và T_1 là cây con thực sự của T , nên có thể viết $z = uz_1y$. Mặt khác, T_2 lại là cây con thực sự của T_1 , nên ta viết được $z_1 = vwx$. $|vwx| > |w|$ suy ra $|vx| \geq 1$. Do vậy, $z = uvwxy$ với $|vwx| \leq n$ và $|vx| \geq 1$. Đến đây ta đã chứng minh được các điều (i) – (iii).

Bởi T là S -cây dẫn xuất và T_1, T_2 là hai B -cây dẫn xuất, nghĩa là $S \xRightarrow{*} uBy$, $B \xRightarrow{*} vBx$ và $B \xRightarrow{*} w$. Vì $S \xRightarrow{*} uBy \Rightarrow uwy = uv^0wx^0y \in L$. Với $k \geq 1$, $S \xRightarrow{*} uBy \xRightarrow{*} uv^kBx^ky \xRightarrow{*} uv^kwx^ky \in L$. Điều này chứng minh điều kiện (iv) của bổ đề Bơm. ■

Ví dụ 5.7 Chỉ ra rằng $L = \{a^p \mid p \text{ là số nguyên tố}\}$ không phải là ngôn ngữ phi ngữ cảnh.

Giải: Giả sử L là phi ngữ cảnh. Theo bổ đề Bơm thì sẽ tìm được số n để thỏa mãn các điều kiện của định lý 5.10.

$z = a^p$, p là số nguyên tố và $p \geq n$. Ta có thể viết nó dưới dạng $z = uvwxy$. Chọn $k = 0$, theo bổ đề Bơm ta có $uv^0wx^0y \in L$. Suy ra $|uwy| = q$ là số nguyên tố. Đặt $|vx| = m$ và xét tiếp dãy $uv^qwx^qy \in L$. Nó có độ dài $|uv^qwx^qy| = q + q * m = q(1 + m)$, lại không phải là số nguyên tố, do đó mâu thuẫn với giả thiết.

Bổ đề Bơm được sử dụng để chỉ ra một ngôn ngữ không phải là phi ngữ cảnh. Trước tiên, giả sử L là ngôn ngữ phi ngữ cảnh. Áp dụng bổ đề Bơm thì dẫn đến mâu thuẫn.

Thuật toán áp dụng bổ đề Bơm để kiểm tra tính phi ngữ cảnh của L .

1. Giả sử L là phi ngữ cảnh và n là số tự nhiên nhận được từ bổ đề Bơm.
2. Chọn $z \in L$ sao $|z| \geq n$. Viết $z = uvwxy$ theo bổ đề.
3. Tìm một số k sao cho $uv^kwx^ky \notin L$. Điều này là mâu thuẫn, do vậy L không phải là phi ngữ cảnh.

Ví dụ 5.8 Hãy chỉ ra rằng $L = \{a^n b^n c^n \mid n \geq 1\}$ không phải là ngôn ngữ phi ngữ cảnh mà là ngôn ngữ cảm ngữ cảnh.

Giải:

Bước 1. Giả sử L là phi ngữ cảnh và n là số nguyên như trong bổ đề Bơm nêu trên.

Bước 2. Chọn $z = a^n b^n c^n$. Khi đó $|z| = 3n > n$. Theo bổ đề ta viết được $z = uvwxy$, trong đó $|vx| > 1$, suy ra ít nhất là v hoặc x sẽ khác rỗng.

Bước 3. $uvwxy = a^n b^n c^n$. Vì $|uwx| \leq n$ nên suy ra $1 \leq |ux| \leq n$. Dễ dàng thấy rằng, v và x không thể chứa tất cả ba chữ a, b, c . Khi đó ta có

1. v hoặc x có dạng $a^i b^j$ (hoặc $b^i c^j$) với $i + j \leq n$, $i, j \geq 1$.

2. v hoặc x chỉ chứa một trong ba chữ a, b, c .

Trường hợp 1. Nếu $v = a^i b^j$ (hoặc $x = a^i b^j$) thì $v^2 = a^i b^j a^i b^j$ (tương tự $x^2 = a^i b^j a^i b^j$) với $i + j \leq n$. Bởi vì v^2, x^2 là hai xâu con của uv^2wx^2y , nên $ua^i b^j a^i b^j wa^i b^j a^i b^j y$ không thể viết thành dạng $a^m b^m c^m$ được, nghĩa là $uv^2wx^2y \notin L$. Tương tự đối với trường hợp $v = b^i c^j$ (hoặc $x = b^i c^j$).

Trường hợp 2. Nếu v, x chỉ chứa một trong ba chữ a, b, c (ví dụ $v = a^i$ và $x = b^j$, hoặc $v = b^i$ và $x = c^j$ với $1 \leq i, j \leq n$), thì xâu uwy sẽ chứa chữ thứ ba còn lại, gọi là a_1 . Hiển nhiên a_1^n sẽ là xâu con của uwy vì a_1 không xuất hiện trong v và x . Biết rằng $uvwxy = a^n b^n c^n$ nên số các hai chữ khác với a_1 trong uwy là nhỏ hơn n , nghĩa là $uwy \notin L$. Mặt khác $uv^0wx^0y = uwy \notin L$.

Vậy L không phải là phi ngữ cảnh.

5.5 Ôtômát đẩy xuống (Push-Down Automata) và ứng dụng.

Mô hình toán học hữu hạn đã nghiên cứu trong các chương trước không có khả năng đoán nhận ngay cả một ngôn ngữ đơn giản như $L = \{a^n b^n \mid n \geq 1\}$ vì nó không thể đếm được số chữ a đọc được trước khi chữ b đầu tiên được tính đến. Không thể xây dựng ôtômát hữu hạn trạng thái để đoán nhận được L bởi vì các xâu của nó có dạng $a^n b^n$ với n là tùy ý và để sinh ra chúng thì phải có vô hạn trạng thái. Tuy nhiên, việc mở rộng trực tiếp ôtômát hữu hạn bằng cách cho phép nó có vô số các trạng thái rõ ràng không phải là mô hình tính toán phù hợp. Vậy ôtômát loại nào thích hợp để đoán nhận được những ngôn ngữ như thế? Đó chính là *ôtômát đẩy xuống*. Một ôtômát đẩy xuống là một ôtômát hữu hạn kèm theo một băng đẩy xuống vô hạn tiềm năng.

Ôtômát đẩy xuống hoạt động như sau: Nó đọc băng chữ (ký hiệu) vào, có trạng thái điều khiển, trạng thái bắt đầu, kết thúc giống như ôtômát hữu hạn. Ngoài ra, nó còn có kho (bộ nhớ) đẩy xuống. Mỗi lần thay đổi trạng thái nó có thể đọc, lấy ra (từ đỉnh xuống đáy) hoặc ghi một ký hiệu vào kho đẩy xuống.

Định nghĩa: Ôtômát đẩy xuống (PDA) là bộ bảy:

$$A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F), \text{ trong đó}$$

- (i) Q là tập hữu hạn khác rỗng các trạng thái,
- (ii) Σ là tập hữu hạn khác rỗng các ký hiệu vào, băng chữ cái,
- (iii) Γ là tập hữu hạn khác rỗng các ký hiệu ngăn xếp,

- (iv) q_0 là trạng thái bắt đầu, $q_0 \in Q$,
- (v) $Z_0 \in \Gamma$ gọi là ký hiệu khởi đầu (đáy ngăn xếp),
- (vi) $F \subseteq Q$ là tập các trạng thái kết thúc,
- (vii) $\delta: \Gamma \times Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow X \subseteq \Gamma^* \times Q$

Ví dụ 5.9 $A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, trong đó $Q = \{q_0, q_1, q_f\}$, $\Sigma = \{a, b\}$,

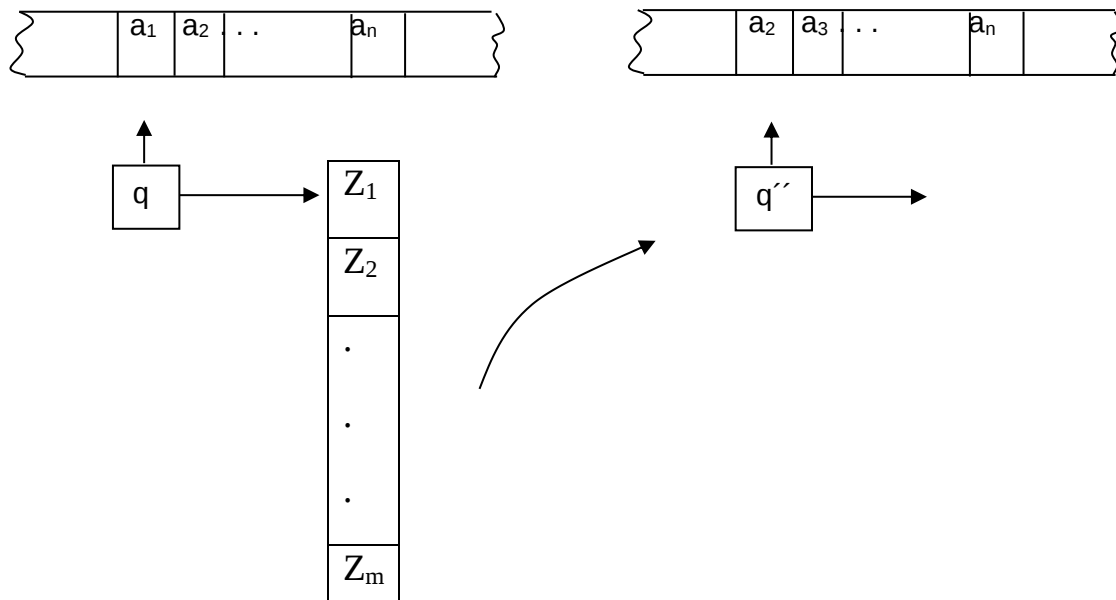
$\Gamma = \{a, Z_0\}$, $F = \{q_f\}$ và δ được xác định như sau:

$$\delta(Z_0, q_0, a) = \{(aZ_0, q_0)\}, \quad \delta(a, q_1, b) = \{(\varepsilon, q_1)\}$$

$$\delta(a, q_0, a) = \{(aa, q_0)\}, \quad \delta(Z_0, q_1, \varepsilon) = \{(\varepsilon, q_1)\}$$

$$\delta(a, q_0, b) = \{(\varepsilon, q_1)\}$$

5.5.1. Cấu tạo và hoạt động của PDA: Ôtômát đẩy xuống (PDA) gồm 1 băng vào, 1 ngăn xếp và 1 bộ điều khiển.



PDA cũng giống FA: Có bộ điều khiển là 1 tập hữu hạn trạng thái, đầu đọc cho phép đọc lần lượt các ký hiệu trên băng vào từ trái sang phải.

PDA khác FA : PDA có 1 Stack hoạt động theo nguyên tắc LIFO, làm tăng khả năng nhớ của PDA.

Hoạt động của Stack: Khi đưa ký tự vào Stack, thì ký hiệu đó nằm ở đầu (đỉnh) của Stack; Ngược lại, khi đọc và đưa ra khỏi Stack: Ký hiệu trên cùng được đọc và khi ký hiệu đó đã được đọc xong thì nó bị loại ra khỏi ngăn xếp.

Một bước chuyển của PDA: Khi thực hiện bước chuyển, PDA căn cứ vào:

- + Trạng thái hiện thời của bộ điều khiển (Q);
- + Ký hiệu quan sát được trên băng vào;
- + Ký hiệu trên đầu (đỉnh) của Stack.

Để đưa ra các hành động thích hợp sau:

- Bộ điều khiển chuyển sang một trạng thái mới nào đó (Q);
- Ghi một xâu ký hiệu nào đó vào đầu Stack;
- Đầu đọc có thể chuyển sang ô bên phải.

(Nếu đầu đọc chuyển sang phải 1 ô, thì quá trình trên được gọi là một bước chuyển. Ngược lại- Đầu đọc đứng yên, thì được gọi là bước chuyển mù).

5.5.2. **Đoán nhận xâu vào của PDA:**

Có 02 cách đoán nhận xâu vào của PDA.

- +) Xâu vào được đọc xong và PDA đến được 1 trong các trạng thái cuối ($\in F$);
- +) Xâu vào được đọc xong và Stack trở thành rỗng.

Quy ước: Ký hiệu vào: $a, b, c, \dots \in \Sigma$; Xâu vào: $u, v, x, y, \dots$; $\alpha, \beta, \gamma, \dots \in \Gamma$.

Khái niệm 1. Bước chuyển được nhìn dưới dạng hàm chuyển:

Có 2 dạng tương ứng với 2 cách đoán nhận:

$\delta(z, q, a) = \{(\gamma_1, q_1), (\gamma_2, q_2), \dots, (\gamma_m, q_m)\}$ - Đa trị, trong đó $z \in \Gamma$; $q, q_i \in Q$; $a \in \Sigma$; $\gamma_i \in \Gamma^*$. Có nghĩa là: Ở trạng thái q , z ở đỉnh Stack, nhìn lên ký hiệu $a \in \Sigma$ (trên băng) thì PDA chuyển sang trạng thái q_i , thay z ở đỉnh Stack bằng γ_i và đầu đọc chuyển sang phải 1 ô.

(Vẽ hình mô tả)

Nếu $a = \varepsilon$ tức là: $\delta(z, q, \varepsilon) = \{(\gamma_1, q_1), (\gamma_2, q_2), \dots, (\gamma_m, q_m)\}$, thì được gọi là bước chuyển mù (bước chuyển rỗng). Trong trường hợp này, ký hiệu vào không ảnh hưởng tới bước chuyển (trong bước chuyển đó đầu đọc vẫn đứng yên tại chỗ).

Có nghĩa là: Ở trạng thái q , z ở đỉnh Stack, PDA chuyển trạng thái q về q_i và thay z ở đỉnh Stack bằng γ_i , còn đầu đọc không dịch chuyển.

Khái niệm 2. Hình trạng của PDA: Gồm 3 thành phần: $\alpha q x$

Trong đó $q \in Q$; $x \in \Sigma^*$ - Đây là phần xâu chưa được đọc; $\alpha \in \Gamma^*$ (Stack) – Ký hiệu trái nhất ở dưới. Lúc đó, q nhìn lên ký tự trái nhất của x trên băng vào.

Hình trạng đầu của PDA: $z_0 q_0 w$.

Vậy, quá trình đoán nhận xâu w bởi PDA là quá trình biến đổi các hình trạng.

Một xâu w được đoán nhận bởi PDA, nếu **tồn tại** một quá trình đoán nhận nó.

Chú ý: Khi đưa một xâu γ vào Stack, ký hiệu bên trái nhất của γ nằm ở dưới, ký hiệu phải nhất sẽ nằm ở đỉnh Stack.

Nếu hiện thời $z \in \Gamma$ và $\gamma = Y_1 Y_2 \dots Y_k$ thì việc đưa γ vào Stack được mô tả như:

Y_k
Y_{k-1}
$\dots$
Y_1
z

5.5.3. Ngôn ngữ đoán nhận của PDA:

Ngôn ngữ đoán nhận bởi PDA (M) theo kiểu trạng thái cuối (kết thúc):

$$L(M) = \{ w \in \Sigma^* \mid z_0 q_0 w \Rightarrow^* \gamma p \text{ với } p \in F, \gamma \in \Gamma^* \}.$$

Ngôn ngữ đoán nhận bởi PDA (M) theo kiểu Stack rỗng:

$$N(M) = \{ w \mid z_0 q_0 w \Rightarrow^* p \text{ với } p \in Q \}.$$

Ôtômat và ngôn ngữ hình thức

Định lý: Tập các xâu đoán nhận được theo trạng thái cuối của một PDA(M) chính bằng tập các xâu đoán nhận được bằng kiểu Stack rỗng ($L(M) = N(M)$).

$$L(M) = \{w \in \Sigma^* \mid (Z_0, q_0, w) \Rightarrow^* (\gamma, q_f, \Lambda) \text{ với } q_f \in F \text{ và } \gamma \in \Gamma^*\}$$

Ví dụ 5.10 (Đoán nhận):

Cho $M = (\{q_0, q_1, q_f\}, \{a, b, c\}, \{a, b, Z_0\}, \delta, q_0, Z_0, \{q_f\})$, với δ :

$$\delta(Z_0, q_0, a) = \{(Z_0 a, q_0)\}, \quad \delta(Z_0, q_0, b) = \{(Z_0 b, q_0)\},$$

$$\delta(a, q_0, a) = \{(aa, q_0)\}, \quad \delta(a, q_0, b) = \{(ab, q_0)\},$$

$$\delta(b, q_0, a) = \{(ba, q_0)\}, \quad \delta(b, q_0, b) = \{(bb, q_0)\},$$

$$\delta(a, q_0, c) = \{(a, q_1)\}, \quad \delta(b, q_0, c) = \{(b, q_1)\},$$

$$\delta(Z_0, q_0, c) = \{(Z_0, q_1)\}$$

$$\delta(a, q_1, a) = \delta(b, q_1, b) = \{(\epsilon, q_1)\}$$

$$\delta(Z_0, q_1, \epsilon) = \{(Z_0, q_f)\}$$

Xét việc đoán nhận xâu $w = \mathbf{bacab}$. Quy trình :

1. Đặt xâu cần đoán nhận vào Hình trạng đầu ;
2. Hoạt động của Hàm chuyển trạng thái ;
3. Dựa vào Hình trạng cuối đạt được để kết luận : thừa nhận hoặc không.

Hình trạng đầu: $Z_0 q_0 \mathbf{bacab}$ (luôn luôn Xem xét bộ 3 ký tự mà q là ký tự ở giữa)

$$Z_0 q_0 \mathbf{bacab} \Rightarrow Z_0 b q_0 a c a b$$

$$\Rightarrow Z_0 b a q_0 c a b$$

$$\Rightarrow Z_0 b a q_1 a b$$

$$\Rightarrow Z_0 b q_1 b$$

$$\Rightarrow Z_0 q_1 \epsilon$$

$$\Rightarrow Z_0 q_f \epsilon$$

Nghĩa là $Z_0 q_0 \mathbf{bacab} \Rightarrow^* Z_0 q_f \Lambda$, do đó $\mathbf{bacab} \in L(M)$. (**Stack rỗng**)

Tương tự chúng ta có thể chứng minh được rằng $wcw^T \in L(M)$.

Vậy $L = \{wcw^T \mid w \in \{a, b\}^*\} \subseteq L(M)$.

Để chứng minh chiều ngược lại, $L(M) \subseteq L$ ta chỉ cần chỉ ra rằng $L^C \subseteq L(M)^C$ (phần bù của chúng cũng bao nhau). Giả sử $x \in L^C$. Xét tiếp hai trường hợp:

Trường hợp 1: x không chứa ký tự c. Khi đó PDA A không thể chuyển đổi từ trạng thái q_0 đến q_1 và để đến q_f được. Vậy $x \in L(M)^C$.

Trường hợp 2: x có chứa ký tự c, $x = w_1cw_2$, $w_1^T \neq w_2$.

$$(Z_0, q_0, w_1cw_2) \Rightarrow^* (Z_0w_1^T, q_0, cw_2) \Rightarrow (Z_0w_1^T, q_1, w_2)$$

Bởi vì $w_1^T \neq w_2$ nên $(Z_0w_1^T, q_1, w_2)$ không thể chuyển về được (Z_0, q_1, Λ) để kết thúc ở (Z_0, q_f, Λ) . Từ đó cũng suy ra $x \in L(M)^C$.

Kết luận $L(M) = L$.

5.5.4. Mối quan hệ giữa Ôtômát đẩy xuống và Văn phạm phi ngữ cảnh.

Định lý: (Sự tương đương giữa PDA và Văn phạm Phi ngữ cảnh G).

L là một ngôn ngữ phi ngữ cảnh $\Leftrightarrow \exists$ PDA(M) mà $L = N(M)$.

Nghĩa là, khi cho trước một Văn phạm phi ngữ cảnh G, chúng ta luôn tìm được một Ôtômát đẩy xuống M để $L(G) = N(M)$ và ngược lại.

Cho $G = (\Sigma, V, P, S)$ là VP PNC sinh ra $L(G) \Leftrightarrow \exists$ PDA(M) mà $L(G) = N(M)$.

Chứng minh: (**Chiều thuận:** Có Văn phạm G, xây dựng PDA(M) tương ứng):

Giả sử ta có: $G = (\Sigma, V, P, S)$ là VP PNC sinh ra $L(G)$.

Ta xây dựng M như sau: Đặt $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, trong đó ta lấy:

$$\Sigma = \Sigma_G; Q = \{q\} \text{ và } q = q_0; F = \emptyset; \Gamma = \Sigma \cup V; Z_0 = S;$$

Hàm chuyển δ được xác định như sau:

$$\delta: \Gamma \times Q \times (\Sigma \cup \{\epsilon\}) \rightarrow X \subseteq \Gamma^* \times Q$$

$$+). \forall A \in V \text{ và } (A \rightarrow \alpha) \in P \text{ trong } G, \text{ thì } \delta(A, q, \epsilon) = (\alpha^R, q);$$

$$+). \delta(a, q, a) = (\varepsilon, q); \forall a \in \Sigma.$$

Ví dụ 5.11: Cho $G = (\Sigma, V, P, S)$ với $\Sigma = \{a, b\}$; $V = \{A, B\}$; $S = A$;

$$P: A \rightarrow B|AB; B \rightarrow ab|aBb.$$

1. Hãy mô tả đáng điệu các xâu $w \in L(G)$.
2. Tìm cây dẫn xuất đối với xâu $w_1 = aaabbb$; $w_2 = abababab$.
3. Xây dựng PDA(M) mà $N(M) = L(G)$.

Áp dụng sự đoán nhận của M đối với aabb.

Giải (3): $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$. Với:

$$Q = \{q\}; \Sigma = \{a, b\}; \Gamma = \Sigma \cup V = \{a, b, A, B\}; Z_0 = A.$$

Hàm chuyển δ được xác định như sau:

$$\delta(A, q, \varepsilon) = (B, q) \mid (BA, q);$$

$$\delta(B, q, \varepsilon) = (ba, q) \mid (bBa, q);$$

$$\delta(a, q, a) = (\varepsilon, q);$$

$$\delta(b, q, b) = (\varepsilon, q).$$

Tìm đường dẫn của $w = aabb$.

$$A \rightarrow B \rightarrow aBb \rightarrow aabb = w \in L(G)..$$

Đ đoán nhận của PDA (M):

$$\begin{aligned} Aqaabb &\Rightarrow Bqaabb \Rightarrow bBaqaabb \Rightarrow bBqabb \Rightarrow bbaqabb \Rightarrow bbqbb \\ &\Rightarrow bqbb \Rightarrow q \text{ (Stack rỗng)}. \text{ Vậy, } w \in N(M) \end{aligned}$$

Ví dụ 5.12: Cho $\Sigma = \{x, y, z\}$; $V = \{A, B\}$; $S = A$;

$$P: A \rightarrow zyz|Bz; B \rightarrow Bxy|yz;$$

Hãy lập PDA(M) mà $N(M) = L(G)$; Áp dụng đoán nhận cho:

$$w_1 = zyxyz; \quad w_2 = zyxyxyz.$$

Ví dụ 5.13: Xây dựng ôtômát đẩy xuống M tương đương với văn phạm phi ngữ cảnh có các luật sinh: $S \rightarrow 0BB, B \rightarrow 0S \mid 1S \mid 0$. Kiểm tra xem $010^4 \in N(M)$?

Giải: Xây dựng PDA(M) như sau:

$M = (\{q\}, \{0, 1\}, \{S, B, 0, 1\}, \delta, q, S, \emptyset)$ và δ được định nghĩa như sau:

$$R_1: \quad \delta(S, q, \varepsilon) = \{(BB0, q)\}$$

$$R_2: \quad \delta(B, q, \varepsilon) = \{(S0, q), (S1, q), (0, q)\}$$

$$R_3: \quad \delta(0, q, 0) = \{(\varepsilon, q)\}$$

$$R_4: \quad \delta(1, q, 1) = \{(\varepsilon, q)\}$$

Ta kiểm tra xem $010^4 \in N(M)$?

$$Sq \ 010^4 \Rightarrow BB0q \ 010^4 \text{ theo qui tắc } R_1$$

$$\Rightarrow BBq \ 10^4 \text{ theo qui tắc } R_3$$

$$\Rightarrow BS1q \ 10^4 \text{ theo qui tắc } R_2 \text{ bởi vì } (S1, q) \in \delta(B, q, \varepsilon)$$

$$\Rightarrow BSq \ 0^4 \text{ theo qui tắc } R_4$$

$$\Rightarrow BBB0q \ 0^4 \text{ theo qui tắc } R_1$$

$$\Rightarrow BBBq \ 0^3 \text{ theo qui tắc } R_3$$

$$\Rightarrow *000q \ 0^3 \text{ theo qui tắc } R_2 \text{ bởi vì } (0, q) \in \delta(B, q, \varepsilon)$$

$$\Rightarrow *(\varepsilon, q), \text{ theo qui tắc } R_3$$

Vậy, $010^4 \in N(M)$.

Ví dụ 5.14. Xây dựng PDA để đoán nhận được tập các xâu trên $\{a, b\}$, trong đó số chữ a bằng số các chữ b.

Giải: $B = (\{q\}, \{a, b\}, \{Z_0, a, b\}, \delta, q, Z_0, \emptyset)$, trong đó

$$\delta(Z_0, q, a) = \{(Z_0 a, q)\}, \quad \delta(Z_0, q, b) = \{(Z_0 b, q)\},$$

$$\delta(a, q, a) = \{(aa, q)\}, \quad \delta(b, q, b) = \{(bb, q)\}$$

$$\delta(b, q, a) = \{(\varepsilon, q)\}, \quad \delta(q, b, a) = \{(\varepsilon, q)\}$$

$$\delta(Z_0, q, \epsilon) = \{(\epsilon, q)\}$$

Chứng minh được rằng $N(A) = L$ là tập tất cả các xâu trên $\{a, b\}$ trong đó số các chữ a bằng số các chữ b.

Bài tập

5.1 Cho văn phạm G có $P: S \rightarrow S + S \mid S * S, S \rightarrow a \mid b$. Tìm cây dẫn xuất của xâu $w = a * b + a * b$ thuộc $L(G)$.

5.2 Cho văn phạm G có $P: S \rightarrow 0S0 \mid 1S1 \mid A, A \rightarrow 2B3, B \rightarrow 2B3 \mid 3$. Tìm $L(G)$?

5.3 Cho văn phạm G có $P: S \rightarrow aB \mid bA, A \rightarrow aS \mid bAA \mid a, B \rightarrow bS \mid aBB \mid b$. Đối với xâu $w = aaabbabbba$, hãy tìm:

- (i) Dẫn xuất trái nhất của w ,
- (ii) Dẫn xuất phải nhất của w ,
- (iii) Cây dẫn xuất của w .

5.4 Chỉ ra rằng văn phạm G có $P: S \rightarrow SbS \mid a$ là nhập nhằng.

5.5 Chỉ ra rằng văn phạm G có $P: S \rightarrow a \mid abS \mid aAb, A \rightarrow bS \mid aAAb$ là nhập nhằng.

5.6 Xây dựng PDA để đoán nhận bằng kho đẩy xuống rỗng các ngôn ngữ sau:

- (a) $\{a^n b^m c^n \mid m, n \geq 1\}$
- (b) $\{a^n b^{2n} \mid n \geq 1\}$
- (c) $\{a^m b^m c^n \mid m, n \geq 1\}$
- (d) $\{a^m b^n \mid m > n \geq 1\}$

5.7 Xây dựng PDA để đoán nhận bằng trạng thái kết thúc các ngôn ngữ như trong bài 5.6.

5.8 Xây dựng văn phạm phi ngữ cảnh sinh ra các ngôn ngữ sau và sau đó xây dựng các PDA để đoán nhận chúng bằng kho đẩy xuống rỗng:

- (a) $\{a^n b^n \mid n \geq 1\} \cup \{a^m b^{2m} \mid m \geq 1\}$
- (b) $\{a^n b^m c^n \mid m, n \geq 1\} \cup \{a^n c^n \mid n \geq 1\}$
- (c) $\{a^n b^m c^m d^n \mid m, n \geq 1\}$

- 5.9** Chỉ ra rằng tập tất cả các xâu trên $\{a, b\}$ có số ký tự a bằng số ký tự b , được đoán nhận bởi một ôtômát đẩy xuống đơn định.
- 5.10** Chỉ ra rằng $\{a^n b^n \mid n \geq 1\} \cup \{a^m b^{2m} \mid m \geq 1\}$ không thể đoán nhận bởi một ôtômát đẩy xuống đơn định.
- 5.11*** Cho ngôn ngữ phi ngữ cảnh $L = \{a^n c b^m c d^k \mid n > m > 0, k > 0\}$
- (i) Tìm văn phạm phi ngữ cảnh G để sinh ra L ($L(G) = L$).
 - (ii) Chỉ ra một dẫn xuất chứng tỏ từ $a^3 c b^2 c d^2$ được sinh ra bởi G mà anh (chị) đã chỉ ra ở trên.
 - (iii) Xây dựng cây cú pháp của từ $a^3 c b^2 c d^2$.
 - (iv) Xây dựng ôtômát đẩy xuống M để đoán nhận L cho trước ở trên.
 - (v) Chỉ ra một dẫn xuất chứng tỏ từ $a^3 c b^2 c d^2$ được đoán nhận bởi M mà anh (chị) đã chỉ ra ở trên.
- 5.12*** Cho văn phạm phi ngữ cảnh G có tập các qui tắc $P = \{S \rightarrow aSa \mid bSb \mid cSc \mid d\}$
- (i) Tìm ngôn ngữ sinh $L = L(G)$.
 - (ii) Xây dựng ôtômát đẩy xuống M để đoán nhận L cho trước ở trên theo ngăn xếp rỗng, $L(G) = L(M)$.
 - (iii) Cho $w_1 = aababbccdcbbabaa$ và $w_2 = ababbcccdbbabaa$. Chỉ ra rằng $w_1 \in L(G)$ và $w_1 \in L(M)$, còn $w_2 \notin L(G)$ và $w_2 \notin L(M)$.
 - (iv) Xây dựng cây cú pháp của xâu w_1 đối với văn phạm G nêu trên.

TÀI LIỆU THAM KHẢO.

- [1] Nguyễn Văn Ba, Lý thuyết ngôn ngữ và tính toán, NXB ĐHQG Hà Nội 2006.
- [2] Đỗ Đức Giáo, Toán rời rạc, NXB ĐHQG Hà Nội 2002, 2004.
- [3] Hopcroft J. and Ullman J., *Formal languages and their relation to automata*, Addison Wesley, London, 1969.
- [4] Mishra K. and Chandrasekaran, *Theory of Computer Science (Automata, Languages and Computation)*, Prentice Hall, 2001.
- [5] Rosen K. H., *Discrete Mathematics and Its Application*, Mc Graw- Hill, NY 1994.