Teoría de la Computación

Víctor Melchor

Clase 7

Teoría de Autómatas

Capítulo 18

Equivalencia de Gramáticas

En muchas ocasiones es recomendable simplificar ciertas gramáticas eliminando símbolos o reglas no deseadas.

Para esto definiremos una gramática equivalente a la primera pero que no tenga esos elementos indeseables.

Un mismo lenguaje puede ser generado por varias gramáticas.

Definición: Dos gramáticas G^1 y G^2 definidas ambas sobre Σ_T son equivalentes si generan el mismo lenguaje, esto es $L(G^1) = L(G^2)$

Importancia

El problema de simplificar una gramática de modo que cumpla determinados requisitos, pero sin alterar su lenguaje generado es importante en ciertos contextos, como por ejemplo en la construcción de analizadores sintácticos de lenguajes.

18.1 Elementos Indeseables en Gramáticas

Definición: Una regla innecesaria es una producción de la forma:

$$A ::= A$$

Definición: Sea $G = (\Sigma_T, \Sigma_N, S, P)$ una GLC. Una variable $X \in \Sigma_N$ se llama variable útil si y sólo si existen $u, v \in \Sigma^*$ tales que:

$$S \overset{*}{\to} uXv \text{ y existe } w \in \Sigma_T^* \text{ tal que } uXv \overset{*}{\to} w$$

Definición: Llamaremos símbolo inaccesible o inútil a aquel símbolo no terminal que no aparece en ninguna cadena de símbolos que pueda derivarse a partir del símbolo inicial de la gramática.

Ejemplo: Sea la gramática libre de contexto G, con producciones:

$$S \to AB|a$$
$$B \to b$$
$$C \to c$$

Identificar las variables inútiles.

Solución

- C es inútil. No existen $u, v / S \xrightarrow{*} uCv$
- A es inútil. No existe $w \in \Sigma_T^* / A \xrightarrow{*} w$
- B es inútil.

$$S \xrightarrow{*} uBv \text{ para } u = A \text{ y } v = \varepsilon \text{ pero no existe } w \in \Sigma_t^* / AB \xrightarrow{*} w$$

Definición: Denominaremos símbolo no generativo a aquel símbolo no terminal a partir del cual no puede derivarse ninguna cadena de símbolos terminales

Sea $G^1 = (\Sigma_N^1, \Sigma_T, S, P^1)$ una GLC. Transformaremos G^1 en $G^2 = (\Sigma_N^2, \Sigma_T, S, P^2)$ de modo que $L(G^1) = L(G^2)$ y para todo $A \in \Sigma_N^2$ se obtenga $A \xrightarrow{*} w$ para algún $w \in \Sigma_T^*$

Algoritmo

1. Inicializar Σ_N^2 con las variables A para los que $A \to w$ es una regla de G^1 con $w \in \Sigma_T^*$.
2. Inicializar P^2 con todas las reglas $A \to w$ para las cuales $A \in \Sigma_N^2$ y $w \in \Sigma_T^*$
3. Repetir

    Añadir a Σ_N^2 todas los símbolos no terminales A para los cuales $A \to w$, para algún $w \in (\Sigma_N^2 \cup \Sigma_T)^*$ que sea una producción de P^1 y añadirla a P^2

    Hasta que no se puedan añadir más variables a Σ_N^2

Ejemplo: Sea la gramática G

$$
\begin{aligned}
S &\rightarrow Aa|B|D \\
B &\rightarrow b \\
A &\rightarrow aA|bA|B \\
C &\rightarrow abd
\end{aligned}
$$

use el algoritmo propuesto para eliminar variables.

Solución:

$$\Sigma_N^1 = \{S, A, B, C, D\}$$
$$\Sigma_T = \{a, b, d\}$$

1. $\Sigma_N^2 = \{B, C\}$

2. $$P^2 : \begin{cases} B & \rightarrow b \\ C & \rightarrow abd \end{cases}$$

3. (a) $\Sigma_N^2 = \{B, C, S, A\}$
   Añadimos

$$S \rightarrow B$$
$$A \rightarrow B$$

(b) $\Sigma_N^2 = \{B, C, S, A\}$ Añadimos

$$
\begin{aligned}
S &\rightarrow Aa \\
A &\rightarrow aA|bA
\end{aligned}
$$

Obtenemos así la gramática equivalente G^2, donde:

$$\Sigma_N^2 = \{B, C, S, A\}$$

$$
P^2 \begin{cases}
S & \rightarrow Aa|B \\
A & \rightarrow aA|bA|B \\
B & \rightarrow b \\
C & \rightarrow abd
\end{cases}
$$

Ejemplo: Sea la GLC G dada por:

$$S \rightarrow aA|\varepsilon$$

$$A \rightarrow aA|bB|\varepsilon$$

$$B \rightarrow bB$$

Aplique el algoritmo expuesto para eliminar variables.

Solución:

$$\Sigma_N^1 = \{S, A, B\}$$
$$\Sigma_T = \{a, b\} \cup \{\varepsilon\}$$

1. $\Sigma_N^2 = \{S, A\}$

2. $$P^2 : \begin{cases} S & \to \varepsilon \\ A & \to \varepsilon \end{cases}$$

3. (a) $\Sigma_N^2 = \{S, A\}$ Añadimos

$$S \to aA$$
$$A \to aA$$

No se pueden añadir más variables.

Se obtiene así la gramática equivalente G^2, donde:

$$\Sigma_N^2 = \{S, A\}$$

$$P^2 : \begin{cases} S & \rightarrow aA|\varepsilon \\ A & \rightarrow aA|\varepsilon \end{cases}$$

El siguiente algoritmo elimina los terminales y no terminales que no aparezcan en las cadenas que se deriven a partir de S.

Sea $G^1 = (\Sigma_N^1, \Sigma_T^1, S, P^1)$ una GLC. Obtendremos una gramática $G^2 = (\Sigma_N^2, \Sigma_T^2, S, P^2)$ tal que $L(G^1) = L(G^2)$ y para todo $X \in \Sigma_N^2 \cup \Sigma_T^2$ se tenga que $S \rightarrow \alpha X \beta$ para α y $\beta \in (\Sigma_n^2 \cup \Sigma_T^2)^*$.

Algoritmo

1. Inicializar Σ_N^2 tal que contenga a S.

$$P^2 = \{\ \}$$

$$\Sigma_T^2 = \{\ \}$$

2. Repetir

   Para $A \in \Sigma_N^2$, si $A \to w$ es regla de P^1

   (a) Agregar $A \to w$ a P^2

   (b) $\forall B$ de w, agregar B a Σ_N^2

   (c) $\forall \sigma$ de w, agregar σ a Σ_T^2

   Hasta que no se puedan agregar más reglas.

Observaciones

1. Si un terminal o no terminal no puede ser conseguido a partir de S, nunca será incluido en Σ_T^2 o Σ_N^2
2. Si un no terminal no es accesible, entonces todas las reglas que lo tengan en su lado izquierdo serán excluidas.

Ejemplo: Sea la GLC G dada por:

$$S \rightarrow Aa|B$$
$$B \rightarrow b$$
$$A \rightarrow aA|bA|B$$
$$C \rightarrow abd$$

Aplique el algoritmo expuesto para obtener una gramática equivalente.

Solución:

1.

$$\Sigma_N^2 = \{S\}$$
$$\Sigma_T^2 = \{\ \}$$
$$P^2 = \{\ \}$$

2. (a) i. Agregamos $S \to Aa|B$
   ii. $\Sigma_N^2 = \{S, A, B\}$
   iii. $\Sigma_T^2 = \{a\}$
   (b) i. agregamos

$$B \to b$$
$$A \to aA|bA|B$$

ii. $\Sigma_N^2 = \{S, A, B\}$

iii. $\Sigma_T^2 = \{a, b\}$

No se pueden agregar más reglas.

La gramática equivalente G^2 es:

$$P^2 \begin{cases} S & \rightarrow Aa|B \\ B & \rightarrow b \\ A & \rightarrow aA|bA|B \end{cases}$$

Es importante tener en cuenta el orden en que son aplicados los algoritmos a la gramática.

Ejemplo: Sea la GLC G dada por:

$$S \to AB|a$$
$$A \to a$$

Aplique

1. el algoritmo A seguido del algoritmo B
2. el algoritmo B seguido del algoritmo A

Solución

I) i) Aplicando el algoritmo A

$$\Sigma_N = \{S, A, B\} \qquad \Sigma_T = \{a\}$$

1. $\Sigma_n^2 = \{S, A\}$
2. $$P^2 \begin{cases} S & \to a \\ A & \to a \end{cases}$$

3.1 No hay variables que añadir. Se tiene la gramática

$$S \to a$$

$$A \to a$$

ii) Aplicando el algoritmo B

1. $\Sigma_N^2 = \{S\}$ $\Sigma_T^2 = \{\ \}$ $P = \{\ \}$
2.

   2.1) añadir: $S \rightarrow a$

   2.2) $\Sigma_N = \{\ \}$

   2.3) $\Sigma_T = \{a\}$

No hay más reglas que agregar.

Finalmente la gramática equivalente es:

$$P^* \left\{ S \quad \rightarrow a \right.$$

$$\Sigma_N^2 = \{S\}$$
$$\Sigma_T^2 = \{a\}$$

II) i) Aplicando el algoritmo B

1) $\Sigma_N^2 = \{S\}$ $\Sigma_T^2 = \{\ \}$ $P^2 = \{\ \}$

2)

2.1.1 1) $P^2 : S \to AB|a$

2) $\Sigma_N^2 = \{S, A, B\}$

3) $\Sigma_T^2 = \{a\}$

2.1.2 1) $P^2 : A \to a$

2) $\Sigma_N^2 = \{S, A, B\}$

3) $\Sigma_T^2 = \{a\}$

No se pueden agregar más reglas.

La gramática resultante es:

$$\begin{cases} A \to a \\ S \to AB|a \end{cases}$$

(a) Aplicando el Algoritmo A

$$\Sigma_N = \{A, S, B\} \qquad \Sigma_T = \{a\}$$

1) $\Sigma_N^2 = \{A, S\}$

2)

$$P^2 : \begin{cases} A \to a \\ S \to a \end{cases}$$

3) No se pueden añadir variables.

La gramática equivalente es:

$$P^F \begin{cases} S & \to a \\ A & \to a \end{cases}$$

$$\Sigma_N = \{S, A\}$$
$$\Sigma_T = \{a\}$$

Definición: Denominamos producción ε a una regla de la forma:

$$A \to \varepsilon$$

Definición: Una variable $A \in \Sigma_N$ en una GLC se dirá anulable si $A \xrightarrow{*} \varepsilon$

Notación:
El conjunto de todos los no terminales anulables en una GLC se denotará por $\mathcal{N}$

El siguiente algoritmo determina el conjunto de las variables anulables en una GLC:

Algoritmo

1. Inicializar $\mathcal{N}$ con todos los $A \in \Sigma_N / A \to \varepsilon$
2. Repetir

   Si $B \to w$ para $w \in \Sigma^*$ y todos los símbolos de w están en $\mathcal{N}$, añadir B a $\mathcal{N}$

   Hasta que no se puedan añadir más variables a $\mathcal{N}$

A continuación se presenta un algoritmo para la eliminación de reglas ε

Algoritmo

Sea $G^1 = (\Sigma_N, \Sigma_T, S, P^1)$ una GLC. Se obtendrá una gramática equivalente $G^2 = (\Sigma_N, \Sigma_T, S, P^2)$ sin reglas ε, excepto $S \to \varepsilon$

1. Obtener $\mathcal{N}$
2. $\mathcal{P}^2 \leftarrow \emptyset$
3. Para cada regla $X \to w \in P^1$ hacer

   Para cada representación de w como $x_1\Psi_1 x_2 \ldots \Psi_n x_{n+1}$ con $\Psi_1, \ldots, \Psi_n \in \mathcal{N}$

   $$\mathcal{P}^2 \leftarrow \mathcal{P}^2 \cup \{\mathcal{X} \to x_1 x_2 \ldots x_{n+1}\}$$

   Fin_Para

4. Devolver $G^2 = (\Sigma_N, \Sigma_T, S, \mathcal{P}^2 - \{\mathcal{X} \to \varepsilon / \mathcal{X} \neq P\})$

Ejemplo: Elimina las producciones ε de la gramática G

$$S \to aA$$
$$A \to aA|\varepsilon$$

Solución

1. $\mathcal{N} = \{A\}, \varepsilon\}$
2. $\mathcal{P}^2 = \{\ \}$

añadir

$$S \rightarrow a|aA$$
$$A \rightarrow a|aA$$

La gramática resultante G^2 será:

$$S- \rightarrow aA|a$$
$$A \rightarrow aA|a$$

A partir de una producción de $\mathcal{P}^1$

$$B \rightarrow X_1 X_2 \ldots X_n$$

se pueden conseguir nuevas producciones en $\mathcal{P}^2$

Ejemplo: Suponga que X_1 y X_2 son anulables y sea:

$$B \rightarrow X_1 X_2$$

Se pueden obtener las siguientes reglas:

$$B \rightarrow X_1 | X_2 | X_1 X_2$$

Ejemplo: Elimina las producciones ε de la gramática

$$S \to ASB|C$$

$$A \to AaaA|\varepsilon$$

$$B \to BBb|C$$

$$C \to cC|\varepsilon$$

Solución

Las variables anulables son $\mathcal{N} = \{A, C, B, S\}$

$$
\begin{aligned}
S &\to A|\cancel{S}|B|AS|AB|SB|ASB|C|\varepsilon \\
A &\to aaA|Aaa|AaaA|aa \\
B &\to Bb|BBb|b|C \\
C &\to c|cC|
\end{aligned}
$$

Se han eliminado las producciones $A \to \varepsilon$ y $C \to \varepsilon$.

La gramática resultante G^2 será:

$$\begin{cases} S & \rightarrow ASB|AS|AB|SB|A|B|C|\varepsilon \\ A & \rightarrow AaaA|aaA|Aaa|aa \\ B & \rightarrow BBb|Bb|b|C \\ C & \rightarrow cC|c \end{cases}$$

Definición: Una regla se llama producción unitaria o no generativa si es de la forma:

$$A \rightarrow B \qquad \text{con } A, B \in \Sigma_N$$

Las producciones unitarias hacen que la GLC se vuelva compleja

Ejemplo: En la GLC G dada por:

$$A \rightarrow B$$
$$B \rightarrow w|C$$

donde $w \in \Sigma^*$

podemos eliminar la producción unitaria $A \rightarrow B$, de modo que:

$$A \rightarrow w|C$$

aquí se añadió la producción unitaria $A \rightarrow C$.

Definición: Sea G una GLC. Para un símbolo $A \in \Sigma_N$ se define:

$$Unit(A) = \{B \in \Sigma_N / A \stackrel{*}{\rightarrow} B \text{ usando solo producciones unitarias}\}$$

Observación: $A \in Unit(A)$ ya que $A \stackrel{*}{\rightarrow} A$ en 0 producciones unitarias.

Ejemplo: Dada la gramática G:

$$A \rightarrow B$$
$$B \rightarrow C|w_1$$
$$C \rightarrow D$$
$$D \rightarrow w_2$$

Obtener $Unit(A)$, $Unit(B)$, $Unit(C)$, $Unit(D)$

Solución

$$
\begin{aligned}
Unit(A) &= \{A, B, C, D\} \\
Unit(B) &= \{B, C, D\} \\
Unit(C) &= \{C, D\} \\
Unit(D) &= \{D\}
\end{aligned}
$$

Sea $G^1 = (\Sigma_N, \Sigma_T, S, P^1)$ una GLC sin producciones ε. El siguiente algoritmo obtiene una gramática equivalente $G^2 = (\Sigma_N, \Sigma_T, S, P^2)$ donde P^2 no tiene producciones unitarias.

Algoritmo:

1. Inicializar $P^2 \leftarrow \emptyset$
2. Para cada $A \in \Sigma_N$ obtener $Unit(A)$
3. Para cada A tal que $Unit(A) \neq \{A\}$

   Para cada $B \in Unit(A)$

   Para cada producción no unitaria $B \rightarrow w$ de P^1

   añadir $A \rightarrow w$ a P^2
4. Añadir a P^2 las producciones no unitarias restantes de P^1

Ejemplo: Sea G la GLC dada por:

$$
\begin{aligned}
S &\to A|Aa \\
A &\to B \\
B &\to C|b \\
C &\to D|ab \\
D &\to b
\end{aligned}
$$

Eliminar todas las producciones unitarias de G.

Solución:

Tenemos $\Sigma_N = \{S, A, B, C, D\}$

2.

$$
\begin{aligned}
Unit(S) &= \{S, A, B, C, D\} \\
Unit(A) &= \{A, B, C, D\} \\
Unit(B) &= \{B, C, D\} \\
Unit(C) &= \{C, D\} \\
Unit(D) &= \{D\}
\end{aligned}
$$

3.

$$S \rightarrow b|ab$$
$$A \rightarrow b|ab$$
$$B \rightarrow ab|b$$
$$C \rightarrow b|ab$$

Aquí se eliminan las producciones

$$S \rightarrow A, \quad A \rightarrow B, \quad B \rightarrow C, \quad C \rightarrow D$$

La gramática resultante es:

$$
\begin{aligned}
S &\rightarrow b|ab|Aa \\
A &\rightarrow b|ab \\
B &\rightarrow ab|b \\
C &\rightarrow b|ab \\
D &\rightarrow b
\end{aligned}
$$