Autómatas y Lenguajes Formales
Nota 11. Forma Normal de Greibach y teorema de Chomsky-Schützenberger*

Noé Salomón Hernández S.

1. Forma Normal de Greibach (FNG)

Definición 1.1 Una gramática libre de contexto $G = (V, T, P, S)$ está en **Forma Normal de Greibach (FNG)** si todas sus producciones son de la forma:

$$A \longrightarrow a\alpha$$

donde $A \in V$, $a \in T$ y $\alpha \in V^*$.

Definición 1.2 Una producción $\mathsf{P} \in P$ está en forma de **recursión izquierda directa** si $\mathsf{P} : A \longrightarrow A\alpha$, para algún $A \in V$ y $\alpha \in (V \cup T)^*$.

La forma normal de Greibach tiene importantes aplicaciones, como son: requiere una derivación de n pasos para generar una cadena de tamaño n, pues a cada paso genera un símbolo terminal; es usada en la demostración del teorema de Shamir, del cual se sigue el teorema de Chomsky-Schützenberger (ver Sección 2) que indica que todo lenguaje libre de contexto es esencialmente un lenguaje de paréntesis balanceados modificado de un modo relativamente sencillo; y si a una gramática en FNG se le transforma en un PDA es posible obtener un autómata de pila sin transiciones ε, demostrando así que siempre es posible eliminar tales transiciones ε de un PDA.

Teorema 1.3 *Si $G = (V, T, P, S)$ es una GLC, entonces podemos construir una GLC G_1 en Forma Normal de Greibach tal que $\mathcal{L}(G_1) = \mathcal{L}(G) - \{\varepsilon\}$.*

Demostración. El algoritmo que convierte G a la Forma Normal de Greibach es

1. Transformamos G a la Forma Normal de Chomsky, obteniendo $G' = (V', T, P', S)$ que genera el lenguaje $\mathcal{L}(G') = \mathcal{L}(G) - \{\varepsilon\}$.
2. De la G' anterior en FNC obtenemos una gramática intermedia del siguiente modo:
   a. Asignamos números consecutivos a todas las variables. Las variable ahora son $A_1, A_2, \ldots, A_r$, con $S = A_1$.

*La Sección 2 de esta nota se basa en el libro: D. C. Kozen. *Automata and Computability*, Springer-Verlag, Inc.

b. Buscamos transformar cada regla de manera que si tenemos $A_i \longrightarrow A_j\gamma \in P'$, entonces $j > i$. Si el lado derecho de la regla inicia con un símbolo terminal, no realizamos modificaciones. Procedemos del menor al mayor índice de las variables como sigue:

- Suponemos que las producciones para las primeras k variables han sido modificadas, es decir, para $1 \leq i \leq k$ se tiene $A_i \longrightarrow A_j\gamma \in P'$ con $\gamma \in (V \cup T)^*$ y $j > i$.
- Si para A_{k+1} tenemos la producción $A_{k+1} \longrightarrow A_j\gamma$ con $\gamma \in (V \cup T)^*$ y $j < k+1$, entonces generamos nuevas producciones al reemplazar a A_j por el lado derecho de cada producción para A_j. Repetimos este paso a los más k veces para obtener las reglas de la forma $A_{k+1} \longrightarrow A_p\gamma$, con $p \geq k+1$.
- Eliminamos recursión izquierda directa al reemplazar las reglas

$$A \longrightarrow A\alpha_1 \,|\, A\alpha_2 \,|\, \ldots \,|\, A\alpha_n \,|\, \beta_1 \,|\, \beta_2 \,|\, \ldots \,|\, \beta_m, \quad \alpha_i, \beta_j \in (V \cup T)^*$$

por

$$\begin{aligned} A &\longrightarrow \beta_k \,|\, \beta_k Z, \quad 1 \leq k \leq m; \text{ y} \\ Z &\longrightarrow \alpha_\ell \,|\, \alpha_\ell Z, \quad 1 \leq \ell \leq n. \end{aligned} \qquad (\spadesuit)$$

3. De la gramática intermedia recién generada construimos G_1 del siguiente modo:

a. Cambiamos las producciones de la forma $A_i \longrightarrow A_j\gamma$ por $A_i \longrightarrow a\gamma'$, donde $A_i, A_j \in V'$, $\gamma, \gamma' \in (V' \cup T)^*$ y $a \in T$.

- En la gramática intermedia, el lado derecho de todas las producciones de la última variable, A_n, debe empezar con un símbolo terminal.
- Modificamos todas las producciones de la forma $A_i \longrightarrow A_j\gamma$, reemplazando A_j por el lado derecho de sus producciones, tomando $i = n-1$, luego $i = n-2$, …, $i = 2$, y por último $i = 1$. Al reemplazar A_n en las producciones de A_{n-1}, resulta que las producciones de A_{n-1} empiezan todas con una terminal, pues las producciones de A_n empiezan con una terminal. Procedemos de este modo hasta llegar a A_1, y todas las variables $A_1, A_2, \ldots, A_n$ tendrán producciones que empiezan con una terminal.

b. Modificamos del mismo modo las producciones $\spadesuit$ que se generaron al eliminar la recursión izquierda directa.

Terminamos con una gramática en Forma Normal de Greibach G_1 equivalente a la original. $\dashv$

Ejemplo 1.4 Convierta la siguiente gramática G a la Forma Normal de Greibach.

$$G: \begin{aligned} S &\longrightarrow AB \\ A &\longrightarrow BS \,|\, b \\ B &\longrightarrow SA \,|\, a \end{aligned}$$

El paso 1 ya está hecho porque G está en FNC. Así que comencemos generando la gramática intermedia. Por el paso 2a renombramos las variables

$$\begin{array}{ccc} S & \text{como} & A_1 \\ A & \text{como} & A_2 \\ B & \text{como} & A_3 \end{array}$$

Al cambiar en la gramática los nuevos nombres de las variables tenemos,

$$G: \begin{array}{l} A_1 \longrightarrow A_2A_3 \\ A_2 \longrightarrow A_3A_1 \mid b \\ A_3 \longrightarrow A_1A_2 \mid a \end{array}$$

Ahora realizamos el paso 2b. Vemos que $A_1 \longrightarrow A_2A_3$ y $A_2 \longrightarrow A_3A_1 \mid b$ cumplen con la condición. Pero $A_3 \longrightarrow \mathbf{A_1A_2} \mid a$ tiene un problema, puesto que el índice de A_1 es menor que el de A_3. En las reglas de A_3 reemplazamos la primer presencia de A_1 por el lado derecho de su producción, éste es A_2A_3. Así

$$A_3 \longrightarrow \mathbf{A_2A_3}A_2 \mid a$$

El problema persiste ya que ahora el índice de A_2 es menor que el de A_3. En las reglas de A_3 reemplazamos la primer presencia de A_2 por el lado derecho de sus producciones, éstos son A_3A_1 y b. Obtenemos

$$A_3 \longrightarrow \mathbf{A_3A_1}A_3A_2 \mid \mathbf{b}A_3A_2 \mid a$$

Tenemos que eliminar la recursión izquierda directa para la producción anterior. Aplicando el procedimiento resulta en

$$\begin{array}{l} A_3 \longrightarrow bA_3A_2 \mid a \mid bA_3A_2\mathbf{Z} \mid a\mathbf{Z} \\ \mathbf{Z} \longrightarrow A_1A_3A_2 \mid A_1A_3A_2\mathbf{Z} \end{array}$$

La gramática intermedia depués del ejecutar el paso 2 es

$$\begin{array}{l} A_1 \longrightarrow A_2A_3 \\ A_2 \longrightarrow A_3A_1 \mid b \\ A_3 \longrightarrow bA_3A_2 \mid a \mid bA_3A_2Z \mid aZ \\ Z \longrightarrow A_1A_3A_2 \mid A_1A_3A_2Z \end{array}$$

Las reglas para las variables A_1, A_2, y A_3 no tienen recursión izquierda directa, y para las dos primeras el lado derecho de sus producciones comienza con una variable de índice mayor, además las producciones de A_3 empiezan con un símbolo terminal. Procedemos a cambiar las producciones de A_1 y A_2 para que empiecen con un símbolo terminal, como lo indica el paso 3a. También lo haremos para Z, de acuerdo al paso 3b.

Reemplazamos en las reglas de A_2 la primer presencia de A_3 por el lado derecho de sus producciones, éstas son bA_3A_2, a, bA_3A_2Z y aZ. Por lo que tenemos

$$A_2 \longrightarrow \mathbf{bA_3A_2}A_1 \mid \mathbf{a}A_1 \mid \mathbf{bA_3A_2Z}A_1 \mid \mathbf{aZ}A_1 \mid b$$

Ahora podemos reemplazar en las reglas de A_1 la primer presencia de A_2 por el lado derecho de sus producciones. Lo que resulta en

$$A_1 \longrightarrow \mathbf{bA_3A_2A_1}A_3 \mid \mathbf{aA_1}A_3 \mid \mathbf{bA_3A_2ZA_1}A_3 \mid \mathbf{aZA_1}A_3 \mid \mathbf{b}A_3$$

Todas las producciones para las variables A_1, A_2 y A_3 comienzan con símbolos terminales. Falta que esto pase para Z, así que reemplazamos en las reglas de Z la primer presencia de A_1 por el lado derecho de sus producciones. Esto se ve reflejado a continuación

$$
\begin{aligned}
Z &\longrightarrow \mathbf{bA_3A_2A_1A_3}A_3A_2 \,|\, \mathbf{aA_1A_3}A_3A_2 \,|\, \mathbf{bA_3A_2ZA_1A_3}A_3A_2 \,|\, \mathbf{aZA_1A_3}A_3A_2 \,|\, \mathbf{bA_3}A_3A_2 \\
Z &\longrightarrow \mathbf{bA_3A_2A_1A_3}A_3A_2Z \,|\, \mathbf{aA_1A_3}A_3A_2Z \,|\, \mathbf{bA_3A_2ZA_1A_3}A_3A_2Z \,|\, \mathbf{aZA_1A_3}A_3A_2Z \\
&\quad \,|\, \mathbf{bA_3}A_3A_2Z
\end{aligned}
$$

Por lo tanto, la gramática G_1 en Forma Normal de Greibach que es equivalente a la inicial es

$$
\begin{aligned}
A_1 &\longrightarrow bA_3A_2A_1A_3 \,|\, aA_1A_3 \,|\, bA_3A_2ZA_1A_3 \,|\, aZA_1A_3 \,|\, bA_3 \\
A_2 &\longrightarrow bA_3A_2A_1 \,|\, aA_1 \,|\, bA_3A_2ZA_1 \,|\, aZA_1 \,|\, b \\
A_3 &\longrightarrow bA_3A_2 \,|\, a \,|\, bA_3A_2Z \,|\, aZ \\
Z &\longrightarrow bA_3A_2A_1A_3A_3A_2 \,|\, aA_1A_3A_3A_2 \,|\, bA_3A_2ZA_1A_3A_3A_2 \,|\, aZA_1A_3A_3A_2 \,|\, bA_3A_3A_2 \\
Z &\longrightarrow bA_3A_2A_1A_3A_3A_2Z \,|\, aA_1A_3A_3A_2Z \,|\, bA_3A_2ZA_1A_3A_3A_2Z \,|\, aZA_1A_3A_3A_2Z \\
&\quad \,|\, bA_3A_3A_2Z
\end{aligned}
$$

Ejercicios 1.5

1. Encuentre la FNG de la siguiente gramática libre de contexto.

$$
\begin{aligned}
&S \longrightarrow XA \,|\, BB, \\
&B \longrightarrow b \,|\, SB, \\
&A \longrightarrow a, \qquad X \longrightarrow b.
\end{aligned}
$$

2. Encuentre la FNG de la siguiente gramática libre de contexto.

$$
\begin{aligned}
&S \longrightarrow AB \,|\, BA \,|\, SS \,|\, AC \,|\, BD, \\
&A \longrightarrow a, \qquad B \longrightarrow b, \\
&C \longrightarrow SB, \qquad D \longrightarrow SA.
\end{aligned}
$$

2. Teorema de Chomsky-Schützenberger

Sea PAREN_n el lenguaje que consiste de todas las cadenas de paréntesis de n tipos distintos balanceados. Dicho lenguaje es generado por la gramática:

$$S \longrightarrow \underset{1}{[} S \underset{1}{]} \mid \underset{2}{[} S \underset{2}{]} \mid \ldots \mid \underset{n}{[} S \underset{n}{]} \mid SS \mid \varepsilon$$

El siguiente teorema muestra que el lenguaje de paréntesis PAREN_n juega un papel especial en la teoría de lenguajes libres de contexto: *todo* LLC es esencialmente un lenguaje de paréntesis modificado de una manera relativamente sencilla. En un sentido, los paréntesis balanceados capturan la estructura esencial de los LLC que los diferencia de los lenguajes regulares.

Teorema 2.1 (Chomsky-Schützenberger) *Todo lenguaje libre de contexto es una imagen homomórfica de la intersección de un lenguaje de paréntesis y un lenguaje regular. En otras palabras, para cada LLC A, existe una $n \geq 0$, un lenguaje regular R, y un homomorfismo h tal que*

$$A = h(\text{PAREN}_n \cap R).$$

Recordemos que un homomorfismo es un mapeo $h : \Gamma^* \to \Sigma^*$ tal que $h(xy) = h(x)h(y)$ para toda $x, y \in \Gamma^*$. Se sigue de esta propiedad que $h(\varepsilon) = \varepsilon$ y que h está completamente determinada por sus valores en Γ. La *imagen homomórfica* de un conjunto $B \subseteq \Gamma^*$ bajo h es el conjunto $\{h(x) \mid x \in B\} \subseteq \Sigma^*$, denotado como $h(B)$.

Demostración. Se encuentra en el libro: D. C. Kozen. *Automata and Computability*, Springer-Verlag, Inc.