Capítulo 5

Autómatas de Pila

5.1. Autómatas de Pila Deterministas (AFPD)

Un **Autómata Finito de Pila Determinista** (AFPD), es una 7-upla:

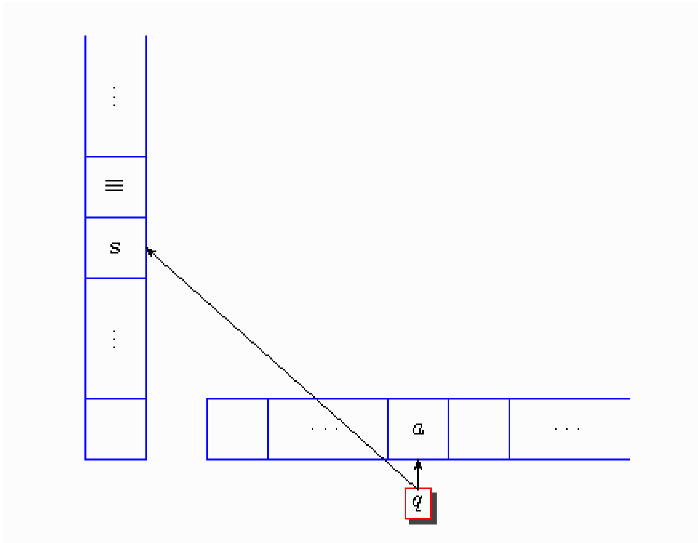
$$M = (\Sigma, Q, q_0, F, \Gamma, s_0, \Delta)$$

cuyos parámetros son:

1. Σ es el alfabeto de cinta.
2. Q es el conjunto (finito) de estados.
3. $q_0 \in Q$ es el estado inicial.
4. F es el conjunto de estados finales o de aceptación, $\emptyset \neq F \subseteq Q$.
5. Γ es el alfabeto de pila.
6. $s_0 \in \Gamma$ es el símbolo inicial de pila.
7. Δ es la función de transición del autómata:

$$\Delta : Q \times (\Sigma \cup \lambda) \times \Gamma \to (Q \times \Gamma^*)$$

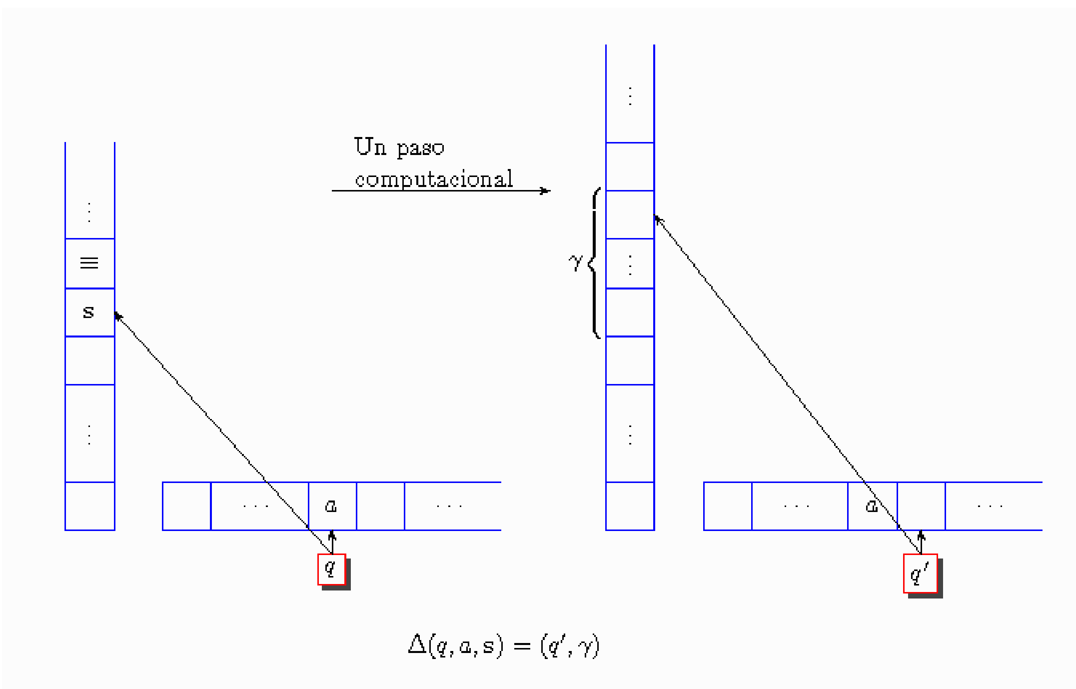
Como en los modelos anteriores, el AFPD procesa palabras sobre una cinta de entrada, pero hay una cinta adicional, llamada pila, que es utilizada por el autómata como lugar de almacenamiento. En un momento determinado, la cabeza lectora del autómata está escaneando un símbolo a sobre la cinta de entrada y el símbolo s en el tope o cima de la pila, como lo muestra la siguiente gráfica:



La transición

$$\Delta(q, a, s) = (q', \gamma)$$

representa un **paso computacional**: la cabeza lectora pasa al estado q' y se mueve a la derecha; además, borra el símbolo s que está en el tope de la pila y escribe la palabra γ (palabra que pertenece a Γ^*). La siguiente gráfica ilustra un paso computacional:



En un paso computacional, el autómata sólo tiene acceso al símbolo que está en el tope de la pila. El contenido de la pila siempre se lee desde arriba (el tope) hacia abajo; esa es la razón por la cual la pila se dibuja verticalmente.

Casos especiales de transiciones:

1. $\Delta(q, a, s) = (q', s)$. En este caso, el contenido de la pila no se altera.

2. $\Delta(q, a, s) = (q', \lambda)$. El símbolo s en el tope de la pila se borra y el control finito pasa a escanear el nuevo tope de la pila, que es el símbolo inmediatamente debajo de s.

3. $\Delta(q, \lambda, s) = (q', \gamma)$. Ésta es una transición λ: el símbolo sobre la cinta de entrada no se consume y la cabeza lectora no se mueve a la derecha, pero el tope s de la pila es reemplazado por la palabra γ. Para garantizar el determinismo, $\Delta(q, a, s)$, con $a \in \Sigma$, y $\Delta(q, \lambda, s)$ no pueden estar simultáneamente definidos (de lo contrario el autómata tendría una opción no-determinista). Las transiciones λ en un AFPD permiten que el autómata cambie el contenido de la pila, sin consumir símbolos sobre la cinta de entrada.

Configuración o descripción instantánea. Es una tripla $(q, au, s\beta)$ que representa lo siguiente: el autómata está en el estado q, au es la parte no procesada de la palabra de entrada, y la cabeza lectora está escaneando el símbolo a. La cadena $s\beta$ es el contenido *total* de la pila; siendo s el símbolo que está en el tope. Para una palabra de entrada $w \in \Sigma^*$, la configuración inicial es (q_0, w, s_0).

Ésta es una notación muy cómoda: para representar el paso computacional de la figura anterior escribimos simplemente

$$(q, au, s\beta) \vdash (p, u, \gamma\beta)$$

Aquí el autómata utilizó la transición $\Delta(q, a, s) = (p, \gamma)$.

La notación

$$(q, u, \beta) \overset{*}{\vdash} (p, v, \gamma)$$

significa que el autómata pasa de la configuración instantánea (q, u, β) a la configuración instantánea (p, v, γ) en cero, uno o más pasos computacionales.

Configuración de aceptación. La configuración (p, λ, β), siendo p un estado final o de aceptación, se llama configuración de aceptación. Esto significa que, para ser aceptada, una palabra de entrada debe ser procesada completamente, con el control finito en un estado de aceptación. La palabra β que queda en la pila puede ser cualquier cadena de símbolos en Γ^*.

Lenguaje aceptado por un AFPD. El lenguaje aceptado por un AFPD M se define como

$$L(M) := \{w \in \Sigma^* : (q_0, w, s_0) \overset{*}{\vdash} (p, \lambda, \beta),\ p \in F\}$$

Es decir, una palabra es aceptada si se puede ir desde es la configuración inicial a una configuración de aceptación, en cero, uno o más pasos.

Observaciones:

1. En el modelo AFPD se permite que la transición $\Delta(q, a, s)$ no esté definida, para algunos valores $q \in Q$, $a \in \Sigma$, $s \in \Gamma$. En tal caso, el cómputo de la palabra de entrada se aborta, es decir, el autómata se detiene.
2. No se debe confundir la tripla que aparece en la función de transición $\Delta(q, a, s)$ con la tripla (q, u, β) que representa una configuración instantánea.
3. La definición de la función de transición Δ requiere que haya por lo menos un símbolo en la pila. No hay cómputos con pila vacía.
4. No hay diagramas de estado (que sean útiles) para los AFPD ya que el procesamiento de una palabra de entrada depende del contenido de la pila, el cual puede cambiar en cada paso computacional.

Ejemplo Diseñar un AFPD que acepte el lenguaje $\{a^i b^i : i \geq 1\}$, sobre el alfabeto $\Sigma = \{a, b\}$.

IDEA: Copiar las a'es en la pila y borrar una a por cada b que sea leída sobre la cinta. Una palabra será aceptada si es procesada completamente y en la pila sólo queda el símbolo s_0.

$M = (\Sigma, Q, q_0, F, \Gamma, s_0, \Delta)$, donde

$$\begin{aligned}
\Sigma &= \{a, b\} \\
\Gamma &= \{s_0, A, B\} \\
Q &= \{q_0, q_1, q_2\} \\
F &= \{q_2\}
\end{aligned}$$

y la función de transición está dada por:

$$
\begin{array}{lcl}
\Delta(q_0, a, s_0) & = & (q_0, As_0) \\
\Delta(q_0, a, A) & = & (q_0, AA) \\
\Delta(q_0, b, A) & = & (q_1, \lambda) \\
\Delta(q_1, b, A) & = & (q_1, \lambda) \\
\Delta(q_1, \lambda, s_0) & = & (q_2, s_0)
\end{array}
$$

Podemos ilustrar el procesamiento de varias palabras de entrada. Sea, inicialmente, $w = aaabbb$.

$$
\begin{array}{rcl}
(q_0, aaabbb, s_0) & \vdash & (q_0, aabbb, As_0) \vdash (q_0, abbb, AAs_0) \vdash (q_0, bbb, AAAs_0) \\
 & \vdash & (q_1, bb, AAs_0) \vdash (q_1, b, As_0) \vdash (q_1, \lambda, s_0) \vdash (q_2, \lambda, s_0)
\end{array}
$$

La última es una configuración de aceptación; por lo tanto $aaabbb$ es aceptada.

Para la palabra de entrada $w = aabbb$, se obtiene el siguiente procesamiento:

$$
\begin{array}{rcl}
(q_0, aabbb, s_0) & \vdash & (q_0, abbb, As_0) \vdash (q_0, bbb, AAs_0) \vdash (q_1, bb, As_0) \\
 & \vdash & (q_1, b, s_0) \quad \text{[cómputo abortado]}
\end{array}
$$

Para la palabra de entrada $w = aaabb$, se tiene:

$$
\begin{array}{rcl}
(q_0, aaabb, s_0) & \vdash & (q_0, aabb, As_0) \vdash (q_0, abb, AAs_0) \vdash (q_0, bb, AAAs_0) \\
 & \vdash & (q_1, b, AAs_0) \vdash (q_1, \lambda, As_0)
\end{array}
$$

A pesar de que se ha procesado completamente la palabra de entrada, la configuración (q_0, λ, As_0) no es de aceptación.

Ejemplo Diseñar un AFPD que acepte el lenguaje de todas las palabras sobre el alfabeto $\Sigma = \{a, b\}$ (diferentes de λ) que tienen igual número de a's que de b's.

IDEA: Acumular las a's o b's consecutivas en la pila. Si en el tope de la pila hay una A y el autómata lee una b, borrar la A; similarmente, si en el tope de la pila hay una B y el autómata lee una a, borrar la B. La palabra de entrada será aceptada si es procesada completamente y en la pila sólo queda el símbolo s_0.

$M = (\Sigma, Q, q_0, F, \Gamma, s_0, \Delta)$, donde

$$
\begin{array}{l}
\Sigma = \{a, b\} \\
\Gamma = \{s_0, A, B\} \\
Q = \{q_0, q_1, q_2\} \\
F = \{q_2\}
\end{array}
$$

y la función de transición está dada por:

$$
\begin{array}{lcl}
\Delta(q_0, a, s_0) & = & (q_1, As_0) \\
\Delta(q_0, b, s_0) & = & (q_1, Bs_0) \\
\Delta(q_1, a, A) & = & (q_1, AA) \\
\Delta(q_1, b, B) & = & (q_1, BB) \\
\Delta(q_1, a, B) & = & (q_1, \lambda) \\
\Delta(q_1, b, A) & = & (q_1, \lambda) \\
\Delta(q_1, \lambda, s_0) & = & (q_2, s_0)
\end{array}
$$

Procesamiento de algunas palabras de entrada.

Palabra de entrada: *aabababb*

$$
\begin{array}{lcl}
(q_0, aabababb, s_0) & \vdash & (q_1, abababb, As_0) \vdash (q_1, bababb, AAs_0) \vdash (q_1, ababb, As_0) \\
& \vdash & (q_1, babb, AAs_0) \vdash (q_1, abb, As_0) \vdash (q_1, bb, AAs_0) \\
& \vdash & (q_1, b, As_0) \vdash (q_1, \lambda, s_0) \vdash (q_2, \lambda, s_0)
\end{array}
$$

Palabra de entrada: *bbbaba*

$$
\begin{array}{lcl}
(q_0, bbbaba, s_0) & \vdash & (q_1, bbaba, Bs_0) \vdash (q_1, baba, BBs_0) \vdash (q_1, aba, BBBs_0) \\
& \vdash & (q_1, ba, BBs_0) \vdash (q_1, a, BBBs_0) \vdash (q_1, \lambda, BBs_0)
\end{array}
$$

En este último caso, la palabra de entrada *bbbaba* es procesada completamente pero la configuración final no es una configuración de aceptación.

Ejemplo Diseñar un AFPD que acepte el lenguaje $L = \{wcw^{-1} : w \in \{a, b\}^*\}$. Aquí el alfabeto es $\Sigma = \{a, b, c\}$.

IDEA. Acumular los símbolos en la pila hasta que aparezca la c. Luego comparar los símbolos leídos con los almacenados en la pila, borrando en cada paso el tope de la pila. La palabra de entrada será aceptada si es procesada completamente y en la pila sólo queda el símbolo s_0.

$M = (\Sigma, Q, q_0, F, \Gamma, s_0, \Delta)$, donde

$$
\begin{array}{l}
\Sigma = \{a, b\} \\
\Gamma = \{s_0, A, B\} \\
Q = \{q_0, q_1, q_2\} \\
F = \{q_2\}
\end{array}
$$

y la función de transición está dada por:

$$
\begin{array}{lcll}
\Delta(q_0, a, s_0) & = & (q_0, As_0) & \\
\Delta(q_0, b, s_0) & = & (q_0, Bs_0) & \\
\Delta(q_0, c, s_0) & = & (q_2, s_0) & \text{(para aceptar la palabra } c\text{)} \\
\Delta(q_0, a, A) & = & (q_0, AA) & \\
\Delta(q_0, a, B) & = & (q_0, AB) & \\
\Delta(q_0, b, A) & = & (q_0, BA) & \\
\Delta(q_0, b, B) & = & (q_0, BB) & \\
\Delta(q_0, c, A) & = & (q_1, A) & \\
\Delta(q_0, c, B) & = & (q_1, B) & \\
\Delta(q_1, a, A) & = & (q_1, \lambda) & \\
\Delta(q_1, b, B) & = & (q_1, \lambda) & \\
\Delta(q_1, \lambda, s_0) & = & (q_2, s_0) &
\end{array}
$$

Ejercicios Diseñar AFPD's que acepten los siguientes lenguajes sobre $\Sigma = \{a, b\}$

1. $L = \{a^{2i}b^i : i \geq 1\}$.
2. $L = \{a^i b^{2i} : i \geq 1\}$.