Autómata de Pila

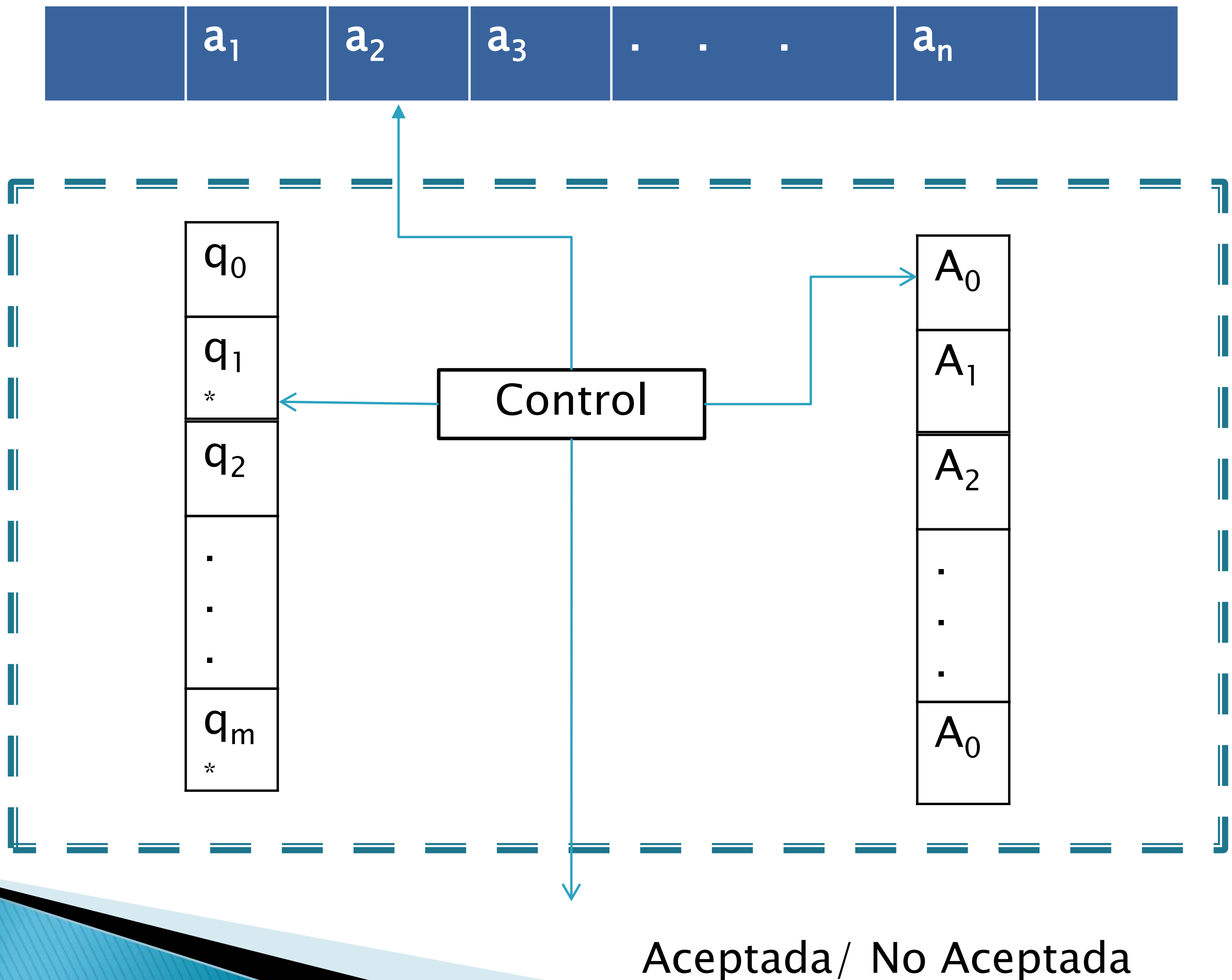
Autómata de Pila

- Los autómatas finitos de pila son una extensión de los autómatas finitos deterministas:
  - Mantienen un conjunto de estados y transiciones entre estados, considerando un alfabeto de entrada
  - Incorporan una pila, que les permite recordar que símbolos han procesado previamente, para tomar decisiones a futuro
  - Analizar palabras para reconocer si pertenecen o no a lenguaje de tipo 2.

Autómata de Pila: Definición

- Un autómata de pila determinista se define como la séptupla $A = (Q, \Sigma, \Gamma, \delta, q0, \gamma 0, F)$, donde:
  - Q, conjunto finito de estados
  - Σ, alfabeto de entrada
  - Γ, alfabeto de pila
  - δ: $Q \times (\Sigma \cup \{\lambda\}) \times \Gamma \rightarrow P(Q \times \Gamma^*)$
  - $q0 \in Q$: estado inicial
  - $\gamma 0 \in \Gamma$: símbolo inicial de la pila
  - $F \subseteq Q$, conjunto de estados finales del autómata

Autómata de Pila: Arquitectura



Autómata de Pila: Movimientos

Tiene dos tipos de transiciones:

- Cuando lee un símbolo de la entrada $a \in \Sigma$, si está en el estado $q \in Q$, hay un símbolo en la cima de la pila $Z \in \Gamma$, y la función de transición para esta terna, es:

  $f(q,a,Z)=\{(q_1,z_1), (q_2,z_2),\ldots, (q_n,z_n)\}$, $q_i \in Q$, $z_i \in \Gamma^*$

- Cuando no se lee ningún símbolo de la entrada, si está en el estado $q \in Q$, hay un símbolo en la cima de la pila $Z \in \Gamma$, y la función de transición para la terna, es:

  $f(q,\lambda,Z)=\{(q_1,z_1), (q_2,z_2),\ldots, (q_n,z_n)\}$, $q_i \in Q$, $z_i \in \Gamma^*$

Autómata de Pila: Ejemplo

Ejemplo: El siguiente Autómata de Pila reconoce el lenguaje formado por n unos seguidos de n ceros, donde n>=1.

$AP=(\{0,1\}, \{A,1,0\}, \{q_o, q_1\}, A, q_0, f, \varnothing)$

Donde f se define como:

$F(q_0,1,A)=\{(q_0,1A)\}$

$F(q_0,1,1)=\{(q_0,11)\}$

$F(q_0,0,1)=\{(q_1,\lambda)\}$

$F(q_1,0,1)=\{(q_1,\lambda)\}$

$F(q_1,\lambda,A)=\{(q_1,\lambda)\}$

El funcionamiento intuitivo de este autómata es que cuando recibe un 1 en la entrada y está en el estado q_0, mete el 1 en la pila para memorizar cuantos ha leído. Cuando lee un 0, transita al estado q_1 que se dedica a eliminar un 1 de la pila por cada 0 que lee. Asi, si la pila se queda vacía (ha reconocido la palabra, como se estudiará más adelante) es porque ha leído el mismo número de unos que de ceros

Autómata de Pila: Ejemplo

Ejemplo: Para el siguiente Autómata de Pila
$AP=(\{0,1\}, \{A,1,0\}, \{q_o, q_1\}, A, q_0, f, \varnothing)$

Donde f se define como:

$F(q_0,1,A)=\{(q_0,1A)\}$
$F(q_0,1,1)=\{(q_0,11)\}$
$F(q_0,0,1)=\{(q_1,\lambda)\}$
$F(q_1,0,1)=\{(q_1,\lambda)\}$
$F(q_1,\lambda,A)=\{(q_1,\lambda)\}$

Se puede realizar los siguientes movimiento dada la palabra de entrada 1100

Autómata de Pila: Descripción Instantánea

La descripción instantánea permite describir sencillamente la configuración del autómata en cada momento.

- Representación: es una terna (q,x,z), $q \in Q$, $x \in \Sigma^*$, $z \in \Gamma^*$, contiene el estado actual, el resto de la palabra de entrada que queda por leer, y la situación de la pila en ese instante.

  Ejemplo: El momento inicial se puede describir como (q_0, 1100, A). El siguiente momento se representaría como (q_0, 100, 1A). El momento final se representaría como (q1,λ,λ)

- Movimientos: si $(p,x) \in f(q,a,Z)$, con $p, q \in Q$, $a \in \Sigma \cup \{\lambda\}$, $z \in \Gamma$, $x \in \Gamma^*$. La descripción instantánea se puede representar como: $(q,ay, ZX) \vdash (p,y,xX)$

  Ejemplo: En el ejemplo anterior se producirían los siguientes movimientos:

  $(q_0, 1100, A) \vdash (q_0, 100, 1A) \vdash (q_0, 00, 11A) \vdash (q_1, 0, 1A) \vdash (q_1, \lambda, A) \vdash (q_1, \lambda, \lambda)$

Autómata a Pila Determinista

- Se dice que un autómata a pila es determinista cuando:
  - $\forall\ q \in Q,\ A \in \Gamma$, si $|f(q,\lambda,A)|>0$ entonces $\forall\ a \in \Sigma$, $f(q,a,A)=\emptyset$; y
  - $\forall\ q \in Q,\ A \in \Gamma,\ a \in \Sigma \cup \{\lambda\}$, $|f(q,a,A)|< 2$

Es decir si hay alguna transición λ dados un estado q y un símbolo de pila A, entonces no puede haber transición con ningún símbolo de entrada, y además, no puede haber más de una transición dados el mismo estado q y símbolo de la pila en la cima de la pila A, incluyendo las transiciones λ

Autómata de Pila Determinista

- Ejemplo de APND:
- $AP=(\{0,1\}, \{A,1,0\}, \{q_o, q_1\}, A, q_0, f, \varnothing)$

Donde f se define como:

$f(q_0,0,A)=\{(q_0,0A)\}$
$f(q_0,1,A)=\{(q_0,1A)\}$
$f(q_0,0,0)=\{(q_0,00), (q_1,\lambda)\}$
$f(q_0,0,1)=\{(q_0,01)\}$
$f(q_0,1,0)=\{(q_0,10)\}$
$f(q_0,1,1)=\{(q_0, 11), (q_1,1), (q_1,\lambda)\}$
$f(q_1,0,0)=\{(q_1,\lambda)\}$
$f(q_1,1,1)=\{(q_1,\lambda)\}$
$f(q_1,\lambda,A)=\{(q_1,\lambda)\}$

Es no determinista debido a que , para las transici
$f(q_0,0,0)$ y $f(q_0,1,1)$, hay más de una posible transic

Lenguaje aceptado por un AP

Hay dos formas de caracterizar el lenguaje aceptado por un autómata a Pila:

- Por vaciado de Pila: el lenguaje aceptado es el conjunto de palabras que permiten desde el estado inicial hasta una descripción instantánea en la que tanto la entrada como la pila estén vacías.

  $LV_{AP} = \{x \mid (q_0, x, A) \vdash^{*} (p, \lambda, \lambda) \text{ y } p \in Q,\ x \in \Sigma^*\}$

- Por estado final el lenguaje por estado final es el equivalente al de los AF: todas las palabras que permiten transitar desde el estado inicial a uno final, independientemente del contenido de la pila.

  $LF_{AP} = \{x \mid (q_0, x, A_0) \vdash^{*} (p, \lambda, X) \text{ y } p \in F,\ x \in \Sigma^*,\ X \in \Gamma^*\}$

Forma Normal de Greibach

- En este caso la gramática tiene la forma:

$P=\{(A::=aX)$ ó $(S::=\lambda) \mid A \in \Sigma_N, X \in \Sigma^*_N, a \in \Sigma_T\}$

- Para pasar a FNG, se realizan los siguientes pasos:

1. Se convierte la gramática a su equivalente gramática limpia.
2. Elimina la recursividad por la izquierda.

   Ejemplo B::=BC seria recursiva por la izquierda, con lo que se borrarían las reglas del símbolo B y se añadirían:

   B::=1B', B'::=CB', y B'::= λ

   O bien, para no tener reglas no generativas:

   B::=1B', B::=1, B'::=CB', y B'::=C

   Eliminando las reglas de redenominación B'::=C, se añade la regla B'::=2

Forma Normal de Greibach

3. Como no puede haber reglas en las que sus partes derechas comiencen por un símbolo no terminal, se deben sustituir esas reglas por otras. Para ello, se establece un determinado orden en los símbolos no terminales, $A_1, A_2, \ldots, A_n$. Y se dividen las reglas en tres grupos:

- Grupo1. $A_i ::= ax (a \in \Sigma_T,\ x \in \Sigma^*)$
- Grupo2. $A_i ::= A_j x,\ (A_i,\ A_j \in \Sigma_N, i < j,\ x \in \Sigma^*)$
- Grupo3. $A_i ::= A_j x,\ (A_i,\ A_j \in \Sigma_N, i > j,\ x \in \Sigma^*)$

El objetivo es conseguir que todas las reglas sean del grupo1. Para ello se seleccionan primero las reglas del grupo3 cuyo i sea mínimo en la ordenación, y se sustituyen estas reglas por las obtenidas al sustituir Aj por todas las partes derechas de sus producciones. Este proceso se aplica iterativamente en orden creciente hasta alcanzar el objetivo.

Forma Normal de Greibach

- Ejemplo: se establece el orden A,B,B’,C en los símbolos no terminales. Con este orden, no hay ninguna regla del Grupo3, por lo que se tratan las reglas del Grupo2. Se sustituyen las apariciones como primer símbolo de la parte derecha de C en las producciones de B, B’ y A, y las de B en A. Asi, la regla A::=CB2 generaría la regla A::=2B2 y se borra la regla A::=CB2. La regla B’::=CB’ se elimina y se sustituye por B’::=2B’.

- Por tanto tras esta primera fase, las producciones serian:

  P’={(A::=2B2), (A::=1B), (A::=λ), (B::=1B’), (B::=1)
  (B’::=2B’), (B’::=2), (C::=2)}

Forma Normal del Greibach

4. En este momento, todas las partes derechas de las reglas empiezan por un símbolo terminal (a excepción de la regla $A::=\lambda$). Si todas las reglas son de alguno de los tres tipos permitidos en FNG, no se hace nada. Para cada regla del tipo:

   $$A::=aX,\ A \in \Sigma_N,\ X \in \Sigma^*,\ a \in \Sigma_T$$

   Donde existe simbolos terminales b en X, entonces por cada símbolo b, se sustituye su aparición en esa producción por un símbolo no terminal N (nuevo si hace falta) que sólo tenga una producción que sea N::=b

Forma Normal de Greibach

Ejemplo: de acuerdo a P'={(A::=2B2), (A::=1B), (A::=λ), (B::=1B'), (B::=1)

(B'::=2B'), (B'::=2), (C::=2)}

La única regla que no esta en FNG es A::=2B2. Se sustituye C::=2 y nos queda A::=2BC.

G=({0,1,2}, {A,B,B',C}, A, P')

P'={(A::=2BC), (A::=1B), (A::=λ), (B::=1B'), (B::=1), (B'::=2B'), (B'::=2), (C::=2)}

Autómata de Pila asociado a una gramática

- Por cada gramática G de tipo 2 existe un Autómata a pila AP que reconoce el lenguaje generado por ésta
- Se estudiaran 2 formas de construirlo:
  a) Definiendo un AP que reconozca por vaciado de Pila
  b) Definiendo un AP que reconozca por estado final.

AP que reconoce por vaciado de pila generado por una gramática de tipo 2

- Si la $G=(\Sigma_T, \Sigma_N, S, P)$ está en forma normal de Greibach, se convierte en un Autómata a Pila No Determinista:

  $$AP=(\Sigma_T, \Sigma_N, \{q\}, S, q, f, \varnothing)$$

  Donde f se define:

  1. Si $A::=aZ \in P$, ($a \in \Sigma_T$, $A \in \Sigma_N$, $Z \in \Sigma^*_N$), entonces $(q,Z) \in f(q,a,A)$
  2. Si $S::=\lambda \in P$, entonces $(q, \lambda) \in f(q, \lambda, S)$

- Si la gramática puede generar la forma sentencial xX, el autómata va a poder pasar desde la situación inicial a una en la que X esté en la pila y haya leído la palabra x. Si X=λ, el autómata habrá reconocido x como palabra válida del lenguaje por vaciado de pila

Ejemplo:

- Se tiene la gramática:

G=({0,1,2}, {A,B,B',C},A,P)

P={(A::=2BC), (A::=1B), (A::=λ),(B::=1B'),
(B::=1C), (B::=1), (B'::=2B'), (B'::=2C), (C::=2)}

El autómata equivalente sería:

AP=({0,1,2}, {A,B,B',C},{q}, A, q, f, ∅)

Donde:

f(q,2,A)={(q,BC)}
f(q,1,A)={(q,B)}
f(q,λ,A)={(q, λ)}
f(q,1,B)={(q,B'), (q,C), (q, λ)}
f(q,2,B')={(q,B'),(q,C)}
f(q,2,C)={(q, λ)}

AP que reconoce por estado final el lenguaje generado por una gramática

- Si la $G=(\Sigma_T, \Sigma_N, S, P)$, se construye un Autómata a Pila No Determinista:

$$AP=(\Sigma_T, \Gamma, Q, A_0, q_0, f, q_2)$$

Donde f se define:

- $\Gamma=\Sigma_T \cup \Sigma_N \cup \{A_0\}$, de forma que $A_0 \notin \Sigma_T \cup \Sigma_N$
- $Q=\{q_0,q_1,q_2\}$ donde q_0 es el estado inicial, q_1 es el estado por el que se realizan las transiciones, y q_2 es el estado final
- La función de transición f se define como sigue:
  - $f(q_0,\lambda,A_0)=\{q_1,SA_0\}$ donde S es el axioma de la gramática
  - Para cada $A \in \Sigma_N$, si $A::=\alpha \in P$, $(\alpha \in \Sigma^*)$, entonces $(q_1, \alpha) \in f(q_1,\lambda,A)$;
  - Para cada $a \in \Sigma_T$, $(q_1, \lambda) \in f(q_1,a,a)$;
  - $f(q_1,\lambda,A_0)=\{q_2,A_0\}$

Ejemplo:

- El autómata de Pila que reconoce por estado final será:
- **AP=({0,1,2}, {0,1,2,A,B,B',C,D},{q_0,q_1,q_2}, D, q_0, f, {q_2})**

Donde f:

$f(q_0,\lambda,D)=\{(q_1, \lambda)\}$

$f(q_1,\lambda,A)=\{(q_1,2BC), (q_1,1B), (q_1, \lambda)\}$

$f(q_1,\lambda,B)=\{(q_1,1B'), (q_1,1C), (q_1,1)\}$

$f(q_1,\lambda,B)=\{(q_2,2B'), (q_1,2C)\}$

$f(q_1,\lambda,C)=\{(q_1,2)\}$

$f(q_1,0,0)=\{(q_1, \lambda)\}$

$f(q_1,1,1)=\{(q_1, \lambda)\}$

$f(q_1,2,2)=\{(q_1, \lambda)\}$

$f(q1, \lambda ,D)=\{(q_2,D)\}$