El Autómata con Pila: Transiciones

El Espacio de Configuraciones

Universidad de Cantabria

Esquema

Transiciones

Necesitamos ahora definir, paso por paso, como se comporta un autómata si queremos entender su funcionamiento. Por supuesto, esto debe ser una generalización para autómatas finitos.

Transiciones

Sea dado un autómata $A := (Q, V, \Sigma, Q_0, F, Z_0, \delta)$. El espacio de configuraciones es el producto $\mathcal{S}_A := Q \times \Sigma^* \times Z_0 \cdot V^*$. Dada una palabra $\omega \in \Sigma^*$, la configuración inicial vendrá dada por:

$$I_A(\omega) := (Q_0, \omega, Z_0),$$

esto es, escribimos ω en la cinta de trabajo, escribimos el estado inicial en la unidad de control y escribimos Z_0 en la pila como tareas de inicialización.

Recordemos

El autómata no esta obligado a leer la cinta de entrada, pero si esta obligado a leer la memoria, esto implica que, para que el autómata continúe la computación, tiene que estar por lo menos el fondo de pila Z_0. Además este símbolo esta en la pila o la pila esta vacía.

Transiciones Read/Push

$$(q, \omega, Z_0 Z) \to_A (q', \omega', Z_0 Z'),$$

donde $q \in Q$ y $\omega := w_1 \cdots w_r, \quad Z_0 Z = Z_0 Z_1 \cdots Z_n$.
Realizamos las siguientes operaciones:

- $\boxed{\text{Read}}$ Leemos la información (q, w_1, Z_n) (es decir, el estado, el primer símbolo de la palabra y el *top* de la pila). Supondremos $w_1 \neq \lambda$.
- $\boxed{\text{Transition}}$ Aplicamos la función de transición obteniendo $\delta(q, w_1, Z_n) = (q', \eta)$, con $\eta \in \Gamma^*$, $\eta \neq \lambda$.
- $\boxed{\text{Push and Move}}$ Entonces,
  - $\omega' = w_2 \cdots w_r$ ("borramos" un símbolo de la palabra a analizar (move).
  - $Z' := Z_0 Z_1 \cdots Z_{n-1}\eta$, es decir, $Z_0 Z' := push(pop(Z_0 Z), \eta)$.

Transiciones Read/Pop

$$(Q, \omega, Z_0 Z) \to_A (Q', \omega', Z_0 Z'),$$

donde $q \in Q$ y $\omega := w_1 \cdots w_r, \quad Z_0 Z = Z_0 Z_1 \cdots Z_n$.
Realizamos las siguientes operaciones:

- $\boxed{\text{Read}}$ Leemos la información (q, w_1, Z_n)
- $\boxed{\text{Transition}}$ Aplicamos la función de transición obteniendo $\delta(q, w_1, Z_n) = (q', \lambda)$. Obsérvese que, en este caso, se ha obtenido λ como símbolo de la pila, esto indica que debemos hacer pop sin hacer push.
- $\boxed{\text{Pop and Move}}$ Entonces,
  - $\omega' = w_2 \cdots w_r$ ("borramos" un símbolo de la palabra a analizar (move).
  - $Z_0 Z' := Z_0 Z_1 \cdots Z_{n-1} := pop(Z_0 Z) = push(pop(Z_0 Z), \lambda)$.

Transiciones Lambda/Push

$$(q, \omega, Z_0 Z) \rightarrow_A (q', \omega', Z_0 Z'),$$

donde $q \in Q$ y $\omega := w_1 \cdots w_r, \quad Z_0 Z = Z_0 Z_1 \cdots Z_n$. Realizamos las siguientes operaciones:

- $\boxed{\text{Read Lambda}}$ En este caso, no se lee la cinta aunque sí se lee la pila. Leeremos (q, λ, Z_n).
- $\boxed{\text{Transition}}$ Aplicamos la función de transición obteniendo $\delta(q, \lambda, Z_n) = (q', \eta)$, con $\eta \in \Gamma^*$, $\eta \neq \lambda$.
- $\boxed{\text{Push}}$ Entonces,
  - $q' = q$ (cambia el estado de la transición).
  - $\omega' = \omega$ ("No borramos" un símbolo de la palabra a analizar (make no move)).
  - $Z_0 Z' := Z_0 Z_1 \cdots Z_{n-1} \eta := push(pop(z), \eta)$.

Énfasis

Es importante señalar que la diferencia entre "instrucciones" /**Push** y /**Pop** es "artificial". Nótese que

$$pop(z) = push(pop(z), \lambda)$$

y

$$push(z) = push(pop(z), \omega),$$

con $\omega \in \Gamma^*$, no tienen diferencias semánticas significativas porque, obviamente, $\lambda \in \Gamma^*$. Las distinguimos para que se pueda ver la operación "borrar el último dígito de la pila" como una operación distinguida.

Énfasis

- No añadimos ninguna novedad si la pila nos dejara leer los k primeros símbolos, en vez del primero solo.
- Si la cinta lee más de una vez el mismo símbolo y solo lee el siguiente dependiendo de la transición tampoco añade más expresividad al autómata.

Énfasis

También es importante ver que los autómatas finitos se pueden ver como autómatas con pila del modo siguiente: Suponemos que la función de transición δ verifica que

$$\delta(q, x, Z) = (q', Z), \forall (q, x, Z) \in Q \times (\Sigma \cup \{\lambda\}) \times \Gamma^*.$$

En este caso, todas las instrucciones pasan por hacer $push(pop(Z_0), \lambda)$ que no cambia el contenido de la pila desde la configuración inicial.

Énfasis

La configuración del autómata es una instantánea que muestra un paso concreto en la computación sobre una palabra. Los autómatas más interesantes son aquellos que son deterministas porque permiten ser simulados en la práctica, ya que realizan una única secuencia de computaciones bien definida.

Énfasis

Teorema

Si A es un autómata con pila determinista, su sistema de transición $(S_A, \rightarrow_A)$ *es determinista. Es decir, dada una configuración* $c \in S_A$*, existirá a lo sumo una única configuración* $c' \in S_A$ *tal que* $c \rightarrow_A c'$.

Demostración

Siguiendo las dos hipótesis de la Definición de Autómata Determinista: Dada una configuración, uno pude hacer una transición **Read/...** o una **Lambda/....** Si cabe la posibilidad de hacer una transición **Lambda/...**, no habrá ninguna transición que permita hacer lectura (por la segunda de las condiciones impuestas). Si, por el contrario, no hay ninguna transición **Lambda/...**, entonces es forzoso hacer una transición de lectura y ésta es, a lo sumo, única.

Representación de un Autómata

Nuestro problema ahora es como representar los autómatas con pila. Solo nos centraremos en los autómatas deterministas, ya que son los únicos que permiten una implementación realista.

Representación de un Autómata

Con los autómatas finitos, utilizábamos una matriz donde se colocaban por filas y columnas, estados y símbolos respectivamente.

Codificación de un Autómata

Vamos a representar la función de transición como una tabla. Para ello, podemos usar dos entradas. De una parte, el producto cartesiano $E := Q \times (\Sigma \cup \{\lambda\}) \times (\Gamma \cup \{Z_0\})$ que será el conjunto de las entradas de la función de transición. De otro lado tenemos un conjunto infinito de las "salidas" $O := Q \times \Gamma^*$, pero nuestra hipótesis nos dice que sólo un número finito de elementos de $Q \times \Gamma^*$ van a entrar en la relación.

Representación de un Autómata

¿Ya están resueltos todos los problemas?
¿Es posible que se den configuraciones que no permitan acabar las computaciones? ¿Por qué esto no ocurre con los autómatas finitos?