

Algoritmo AFND-AFD

Objetivo

Convertir un Automata finito no deterministico (AFND) en un Automata finito deterministico (AFD).

Si en el AFND ibamos de un estado q con el token a a un conjunto de estados $\{r, s, t\}$ entonces la idea es que en el AFD desde q voy a parar al conjunto de los tres estados consumiendo la a .

En vez de ir por 3 caminos a la vez, vamos por un unico camino que despues de consumir toda la palabra me lleve a un conjunto de estados que sean todos los estados alcanzables desde el no deterministico etiquetados con esa palabra.

Definiciones

$M = \langle Q, \Sigma, \delta, q_0, F \rangle$

Sea M el lenguaje aceptado por el **AFND**. Definimos a M' tal que $M' = \langle Q', \Sigma, \delta', q_0', F' \rangle$ donde:

- Q' : Es exactamente uno de los posibles subconjuntos pertenecientes a $P(Q)$.
- δ' : Funcion de transicion que se constuye iterativamente con el algoritmo.
- q_0' : El conjunto que solamente tiene a q_0 . $q_0' = \{q_0\}$.
- F' : Estados finales que son a la vez estados del nuevo Q' .

Algoritmo

1. Definimos $q_0' = \{q_0\}$.
2. Inicializar Q' con $\{q_0\}$ y marco $\{q_0\}$ como *no visitado*.
3. Mientras exista un estado $t \in Q'$ que aun no fue visitado:
 - Marcar el estado t como *visitado*.
 - Para cada simbolo a del alfabeto Σ :
 - Para cada estado $r \in Q$ tal que $r = \delta(t, a)$:
 - Si $r \notin Q'$ entonces agregamos r a Q' .
 - Definir $\delta'(t, a) = r$.
4. Definir $F' = \{t \in Q' \mid t \cap F \neq \emptyset\}$

Este algoritmo corre sobre M . Estamos construyendo M' .

Queremos que F' sea el conjunto de estados nuevos que contenga algun estado final.