2. Autómatas Finitos.

Definición General.

Es una quíntupla $AF = (Q, \Sigma, \delta, q_0, F)$ donde:
- Q, conjunto finito de estados
- Σ, alfabeto de entrada
- $q_o \in Q$, estado inicial
- $F \subseteq Q$, conjunto de estados finales o de aceptación.
- δ, función de transición $(\delta : Q \times \Sigma \cup \{\lambda\} \rightarrow P(Q))$, donde $P(Q)$ denota (todos los subconjuntos de Q)

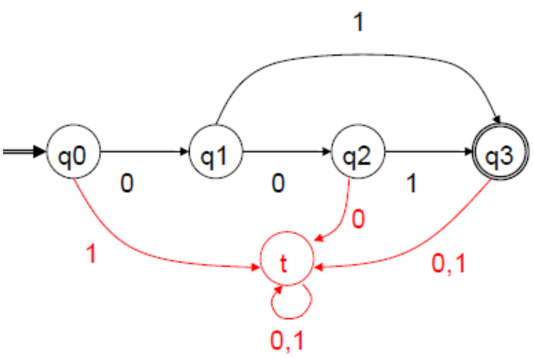
Función de transición. Representación mediante Tabla y Diagrama.

La función de transición de un AFD se puede representar de dos formas: mediante una ***tabla de transición*** o mediante un ***diagrama de transición***.

El ***diagrama de transición*** es una representación gráfica de un autómata finito (grafo) donde:

- Cada **círculo (nodo)** representa un **estado**.
- Cada **círculo doble** representa un **estado final**.
- Una **flecha** apuntando a un estado indica que es el **estado inicial**.
- Cada **arco dirigido** representa una parte de la función de **transición**.
- El símbolo o **etiqueta** sobre el arco indica **qué se consumió** de la entrada al efectuar esa transición.

Ejemplo: $M = (\{q_0, q_1, q_2, q_3, t\}, \{0,1\}, \delta, q_0, \{q_3\})$



$\delta(q_0,0) = \{q_1\}$	$\delta(q_0,1) = \{t\}$
$\delta(q_1,0) = \{q_2\}$	$\delta(q_1,1) = \{q_3\}$
$\delta(q_2,0) = \{t\}$	$\delta(q_2,1) = \{q_3\}$
$\delta(q_3,0) = \{t\}$	$\delta(q_3,1) = \{t\}$
$\delta(t,0) = \{t\}$	$\delta(t,1) = \{t\}$

La ***tabla de transición*** para el mismo autómata sería:

δ	***0***	***1***
q_0	$\{q_1\}$	$\{t\}$
q_1	$\{q_2\}$	$\{q_3\}$
q_2	$\{t\}$	$\{q_3\}$
$* q_3$	$\{t\}$	$\{t\}$
t	$\{t\}$	$\{t\}$

Autómatas Finitos Determinísticos (AFD).

Función de transición

En los AFD la función de transición se define $\delta : Q \times \Sigma \to Q$

A cada par (estado, símbolo) le corresponde uno y sólo un estado.

Función de transición extendida

La ***función de transición extendida*** $\hat{\delta}$ describe lo que ocurre cuando se parte de cualquier estado y se sigue una secuencia de entradas. Se construye por inducción a partir de δ

Definición por inducción sobre la longitud de la cadena de entrada.

BASE: $\hat{\delta}(q,\lambda) = q$

PASO INDUCTIVO: Siendo $\omega = \omega' a$, $\hat{\delta}(q,\omega) = \hat{\delta}(q,\omega' a) = \delta(\hat{\delta}(q,\omega'),a)$

Ejemplo: $M = (\{q_0,q_1,q_2,q_3,t\},\{0,1\},\delta,q_0,\{q_3\})$ del ejemplo anterior.

$$\hat{\delta}(q_0,\text{"001"}) = \delta(\hat{\delta}(q_0,\text{"00"}),1)$$
$$\hat{\delta}(q_0,\text{"00"}) = \delta(\hat{\delta}(q_0,\text{"0"}),0)$$
$$\hat{\delta}(q_0,\text{"0"}) = \delta(\hat{\delta}(q_0,\lambda),0)$$
$$\hat{\delta}(q_0,\lambda) = q_0$$

Por lo tanto, queda:

$$\hat{\delta}(q_0,\text{"0"}) = \delta(\hat{\delta}(q_0,\lambda),0) = \delta(q_0,0) = q_1$$
$$\hat{\delta}(q_0,\text{"00"}) = \delta(\hat{\delta}(q_0,\text{"0"}),0) = \delta(q_1,0) = q_2$$
$$\hat{\delta}(q_0,\text{"001"}) = \delta(\hat{\delta}(q_0,\text{"00"}),1) = \delta(q_2,1) = q_3$$

Es decir, $\hat{\delta}(q_0,\text{"001"}) = q_3$

Configuración instantánea

La ***configuración instantánea*** es una descripción del autómata finito en un momento dado.

Se denota con un par $[q,\omega]$, donde q es un estado y $\omega \in \Sigma^*$

Secuencia de configuraciones:

El símbolo $\mapsto$ indica el movimiento válido entre dos configuraciones.

Es decir: $[q,a\omega] \mapsto [p,\omega] \Leftrightarrow \delta(q,a) = p$

Definición por inducción de secuencia de configuraciones.

BASE: Si I es una configuración instantánea, $I \mapsto^* I$ es una secuencia válida.

PASO INDUCTIVO: $I \mapsto^* J$ es una secuencia válida si existe alguna configuración K tal que $I \mapsto K$ y $K \mapsto^* J$

Ejemplo: $M = (\{q_0,q_1,q_2,q_3,t\},\{0,1\},\delta,q_0,\{q_3\})$ del ejemplo anterior.

$$[q_0,\text{"001"}] \mapsto [q_1,\text{"01"}] \mapsto [q_2,\text{"1"}] \mapsto [q_3,\lambda]$$

Es decir, $[q_0,"001"] \mapsto *[q_3,\lambda]$

Lenguaje aceptado por un Autómata Finito Determinista.

El lenguaje de un AFD es: $L = \{\omega \in \Sigma * / \hat{\delta}(q_0,\omega) \in F\}$

El lenguaje de un AFD es: $L = \{\omega \in \Sigma * / [q_0,\omega] \mapsto *[p,\lambda], p \in F\}$

Determinismo de un autómata finito determinístico.

Dados una palabra $\omega \in \Sigma^*$ y **un** autómata finito definido como se indicó antes, sólo hay **una** secuencia de configuraciones $[q_0,\omega] \mapsto \ldots \mapsto [q_{final},\lambda]$ *(Se demuestra por inducción en longitud de ω y contradicción)*

Equivalencia de autómatas.

Dos AFD M y M' son equivalentes si y sólo si L(M)=L(M')

Es decir, dos autómatas M y M' son equivalentes si y sólo si reconocen el mismo lenguaje.

AFD Mínimo.

Minimización de un AFD.

Estados accesibles.

Un estado $q_i \in Q$ es ***accesible*** si: $\exists \alpha \in \Sigma * / \hat{\delta}(q_0,\alpha) = q_i \in Q$

Un estado $q_i \in Q$ es ***accesible*** si: $\exists \alpha \in \Sigma * / [q_0,\alpha] \xrightarrow{*} [q_i,\lambda]$

Construcción inductiva del conjunto de estados accesibles.

Dado el conjunto Q de estados del autómata finito determinístico M, se construye Q' de estados accesibles:

BASE: El conjunto de un solo elemento $Q' = \{q_0\}$ es accesible ya que

$\exists \lambda \in \Sigma * / \hat{\delta}(q_0,\lambda) = q_0 \in Q$

PASO INDUCTIVO: Si $S \subset Q$ es un conjunto de estados accesibles, para cada $a \in \Sigma \wedge q_i \in S$, el conjunto $S' = \{q_j \in Q / \delta(q_i,a) = q_j\}$ es de estados accesibles.

Estados equivalentes o indistinguibles.

Los estados p y q son ***indistinguibles*** si para toda cadena de entrada ω, $\hat{\delta}(p,\omega)$ es un estado de aceptación si y sólo si $\hat{\delta}(q,\omega)$ es un estado de aceptación.

Simbólicamente, p **indistinguible** de q si:

$$\forall \omega \in \Sigma^* : \hat{\delta}(p,\omega) \in F \Leftrightarrow \hat{\delta}(q,\omega) \in F$$

Por lo tanto:

Los estados p y q son ***distinguibles*** si existe por lo menos una cadena ω tal que $\hat{\delta}(p,\omega)$ es un estado de aceptación y $\hat{\delta}(q,\omega)$ no lo es, o viceversa.

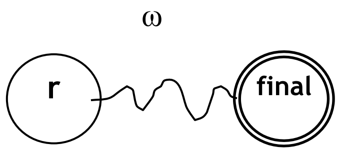
Para detectar estados equivalentes, hay que buscar estados **distinguibles**. Eso se puede hacer con el siguiente algoritmo:

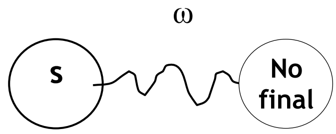
BASE: Si p es un estado de aceptación, y q no lo es, el par {p,q} es distinguible.

- Existe la cadena λ, tal que $\hat{\delta}(p,\lambda)$ es de aceptación y $\hat{\delta}(q,\lambda)$ no lo es.

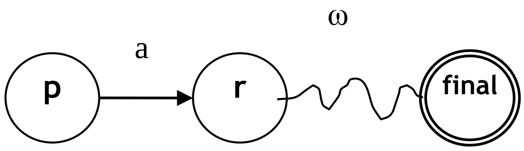
PASO INDUCTIVO: Si p y q son estados tales que, para alguna entrada a, $\delta(p,a)=r$ y $\delta(q,a)=s$, y {r,s} son distinguibles, entonces {p,q} son distinguibles.

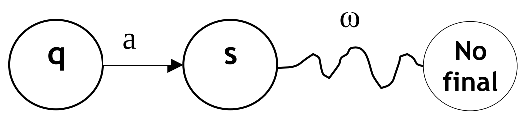
- Como r y s son distinguibles, existe alguna ω/ $\hat{\delta}(r,\omega)$ es de aceptación y $\hat{\delta}(s,\omega)$ no lo es (o viceversa).





- Por lo tanto, si se evalúa $\hat{\delta}(p,\omega')$ y $\hat{\delta}(q,\omega')$ para la cadena ω'= aω resulta:





La ***indistinguibilidad*** es una **relación de equivalencia**.

Propiedad Reflexiva: p es indistinguible de p.

$$\forall \omega \in \Sigma^* : \hat{\delta}(p,\omega) \in F \Leftrightarrow \hat{\delta}(p,\omega) \in F$$

Propiedad Simétrica: si p es indistinguible de q, entonces q es indistinguible de p.

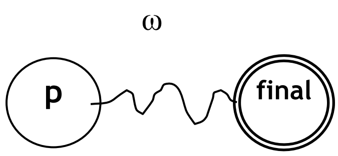
Que p sea indistinguible de q significa que $\forall \omega \in \Sigma^* : \hat{\delta}(p,\omega) \in F \Leftrightarrow \hat{\delta}(q,\omega) \in F$

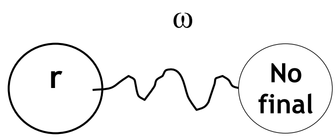
Que q sea indistinguible de p significa que $\forall \omega \in \Sigma^* : \hat{\delta}(q,\omega) \in F \Leftrightarrow \hat{\delta}(p,\omega) \in F$

Propiedad Transitiva: si los pares {p,q} y {q,r} son indistinguibles, entonces {p,r} también lo son.

Demostración. Supongamos que {p,r} no fueran indistinguibles. Entonces, existe alguna ω/ $\hat{\delta}(p,\omega)$ es de aceptación y $\hat{\delta}(r,\omega)$ no lo es (o viceversa).

Suponiendo que_$\hat{\delta}(p,\omega)$ sea el de aceptación.





Veamos si $\hat{\delta}(q,\omega)$ es de aceptación o no.

Si fuera de aceptación, entonces el par {q,r} sería distinguible porque $\hat{\delta}(q,\omega)$ es de aceptación y $\hat{\delta}(r,\omega)$ no lo es.

Si no fuera de aceptación, entonces el par {p,q} sería distinguible porque $\hat{\delta}(p,\omega)$ es de aceptación y $\hat{\delta}(q,\omega)$ no lo es.

Absurdo que contradice la hipótesis y provino de suponer que {p,r} eran distinguibles.

La ***indistinguibilidad de orden k*** considera palabras de una misma longitud k. Es una **relación de equivalencia** y determina una partición del conjunto de estados en clases de equivalencia.

Sea E_k la relación de indistinguibilidad de orden k.

Lema: Dado un autómata, se cumple que $\frac{Q}{E_0} = \{F, Q-F\}$

Si consideramos la relación de indistinguibilidad para palabras de longitud 0,
$\forall \omega \in \Sigma^* \wedge |\omega| = 0 : \hat{\delta}(p,\omega) \in F \Leftrightarrow \hat{\delta}(q,\omega) \in F$

O sea, $\hat{\delta}(p,\lambda) \in F \Leftrightarrow \hat{\delta}(q,\lambda) \in F$ .Por definición de la función de transición extendida, $p \in F$ y $q \in F$

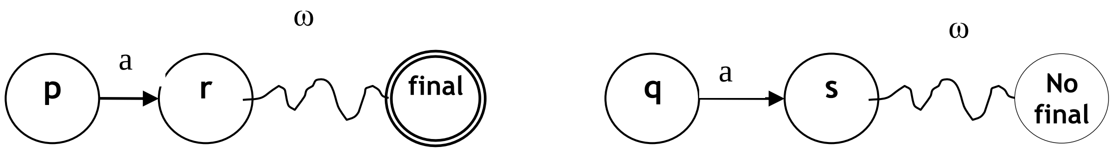
Lema: Dado un autómata y dos estados p y $q \in Q$, $\forall a \in \Sigma : \delta(p,a) E_n \delta(q,a) \Leftrightarrow pE_{n+1}q$

Si $\forall a \in \Sigma : \delta(p,a) E_n \delta(q,a)$, eso significa que
$\forall \omega \in \Sigma^* \wedge |\omega| = n : \hat{\delta}(r,\omega) \in F \Leftrightarrow \hat{\delta}(s,\omega) \in F$, siendo $\delta(p,a) = r \wedge \delta(q,a) = s$.

Por lo tanto, $\forall \omega' \in \Sigma^* \wedge \omega' = a\omega, |\omega'| = n+1 : \hat{\delta}(p,\omega') \in F \Leftrightarrow \hat{\delta}(q,\omega') \in F$

Es decir, $pE_{n+1}q$.

Gráficamente:



Algoritmo de construcción del conjunto cociente.

Entrada: Q

Salida: Conjunto cociente de Q por la relación de indistinguibilidad $\frac{Q}{E}$

1. $\frac{Q}{E_0} = \{F, Q-F\}$
2. Generar $\frac{Q}{E_{i+1}}$ a partir de $\frac{Q}{E_i}$ de la siguiente manera:

p y q pertenecen a la misma clase en $\frac{Q}{E_{i+1}}$ $\Leftrightarrow$

- p y q pertenecen a la misma clase en $\frac{Q}{E_i}$ y
- $\forall a \in \Sigma, \delta(p,a)$ y $\delta(q,a)$ pertenecen a la misma clase en $\frac{Q}{E_i}$

3. Si $\frac{Q}{E_{i+1}} = \frac{Q}{E_i}$, entonces $\frac{Q}{E} = \frac{Q}{E_i}$. Sino, volver al paso 2.

Ejemplo: $A = (\{p, r, q, s, t\}, \{0,1\}, \delta, p, \{s, t\})$

δ	***0***	***1***
p	r	q
r	q	*s
q	q	*t
$*s$	*s	*s
****t***	*t	*t

$\frac{Q}{E_0} = \{F, Q - F\} = \{\{s, t\}, \{p, r, q\}\}$

$\delta(p,0) = \{r\} \notin F \qquad \delta(p,1) = \{q\} \notin F$
$\delta(r,0) = \{q\} \notin F \qquad \delta(r,1) = \{s\} \in F$
$\delta(q,0) = \{q\} \notin F \qquad \delta(q,1) = \{t\} \in F$

$\delta(s,0) = \{s\} \in F \qquad \delta(s,1) = \{s\} \in F$
$\delta(t,0) = \{t\} \in F \qquad \delta(t,1) = \{t\} \in F$

Por lo tanto:

$\frac{Q}{E_1} = \{\{s, t\}, \{p\}, \{r, q\}\}$

Y luego:

$\frac{Q}{E_2} = \{\{s, t\}, \{p\}, \{r, q\}\}$

Al ser $\frac{Q}{E_2} = \frac{Q}{E_1}$ termina el algoritmo y el conjunto cociente es $\frac{Q}{E} = \{\{s, t\}, \{p\}, \{r, q\}\}$

Algoritmo de construcción de AFD mínimo.

Entrada: $A = (Q, \Sigma, \delta, q_0, F)$

Salida: $A' = (Q', \Sigma, \delta', q_0', F')$ equivalente a A con mínima cantidad de estados.

1. Eliminar estados inaccesibles desde q_0.
2. Construir el conjunto cociente $\frac{Q}{E}$.
3. $A' = (Q', \Sigma, \delta', q_0', F')$ donde:

- $Q' = \frac{Q}{E}$
- q_0' es el elemento de $\frac{Q}{E}$ tal que $q_0 \in q_0'$ (la clase donde está q_0)
- $F' = \left\{ s / s \in \frac{Q}{E} \wedge \exists p \in S \text{ tal que } p \in F \right\}$ (las clases donde están estados que pertenecían a F).
- $\delta'(s_i, a) = s_j \Leftrightarrow \exists p \in s_i \wedge \exists q \in s_j / \delta(p, a) = q$

Para el ejemplo anterior, el AFD mínimo es:

$A' = (Q', \Sigma, \delta', q_0', F')$ donde:

- $Q' = \{S, P, Q\}$ (Siendo $S = \{s,t\}, P = \{p\}, Q = \{r, q\}$
- $q_0' = P$
- $F' = \{S\}$
- δ'

δ	0	1
P	Q	Q
Q	Q	$*S$
$*S$	$*S$	$*S$

Teorema: El autómata A' que se obtiene con el algoritmo anterior es equivalente al autómata A y es mínimo (el número de estados de A' es menor o igual que el de cualquier otro AFD equivalente a A).

Demostración:

El autómata A' tiene estados que corresponde a las clases de equivalencias del autómata A. Por lo tanto cualquier transición $\forall p \in Q, \forall x \in \Sigma : \delta(p, x) = q$ se puede corresponder con la equivalente $\delta'(P, x) = Q$ para $P / p \in P$ y $Q / q \in Q$. Por lo tanto, para cualquier palabra $\omega \in \Sigma^*$, si A consume ω en un estado p, A' consume a ω en el estado que contiene a p y estos dos estados son ambos finales o ambos no finales. Eso asegura equivalencia.

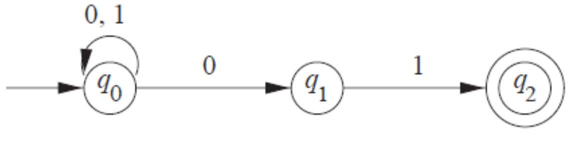
Sea $A = \langle Q, \Sigma, q_0, F, \delta \rangle$ con $|Q| = n$. Supongamos que existiese un autómata $A' = \langle Q', \Sigma, q'_0, F', \delta' \rangle$ equivalente con menos estados $|Q| < n$. Sean α y β tales que $\hat{\delta}(q_0, \alpha) = p$ y $\hat{\delta}(q_0, \beta) = q$ en A, pero $\hat{\delta}'(q'_0, \alpha) = \hat{\delta}'(q'_0, \beta) = r$ en A'. Como A fue obtenido por el algoritmo sabemos que los estados p y q son distinguibles, y que por lo tanto $\exists \omega \in \Sigma^* / \hat{\delta}(p, \omega) \in F \wedge \hat{\delta}(q, \omega) \notin F$ (o viceversa). Por lo tanto, el autómata A aceptaría $\alpha\omega$ pero rechazaría $\beta\omega$. En cambio, en A', las palabras α y β llevan al mismo estado y, dado que $\forall \omega \in \Sigma^* / \hat{\delta}'(r, \omega) \in F \Leftrightarrow \hat{\delta}'(r, \omega) \in F$, $\alpha\omega$ sería aceptada si y sólo si $\beta\omega$ también es aceptada. Por lo tanto A' puede tener menos estados pero no sería equivalente al autómata A.

El AFD mínimo es único, excepto por un posible cambio de nombre de los estados.

Autómatas Finitos No Determinísticos (AFND)

Función de transición

En los AFND la función de transición se define $\delta : Q \times \Sigma \to P(Q)$ donde $P(Q)$ denota (todos los subconjuntos de Q)



$M = (\{q_0, q_1, q_2\}, \{0,1\}, \delta, q_0, \{q_2\})$

Función de transición extendida para un AFND.

La ***función de transición extendida*** $\hat{\delta}$ se redefine para los AFND.

Definición por inducción sobre la longitud de la cadena de entrada.

BASE: $\hat{\delta}(q,\lambda)=\{q\}$

PASO INDUCTIVO: Dada $\omega=\omega' a$, $\hat{\delta}(q,\omega')=\{p_1,p_2,\ldots,p_k\}$, y $\bigcup_{i=1}^{k}\delta(p_i,a)=\{r_1,r_2,\ldots,r_m\}$,

entonces $\hat{\delta}(q,\omega)=\hat{\delta}(q,\omega' a)=\{r_1,r_2,\ldots,r_m\}$

Ejemplo: Sea $M=(\{q_0,q_1,q_2\},\{0,1\},\delta,q_0,\{q_2\})$ del ejemplo anterior.

Entonces, $\hat{\delta}(q_0,00101)=\{q_0,q_2\}$

Secuencia de configuraciones para AFND.

Se redefine: $[q,a\omega]\mapsto[p,\omega]\Leftrightarrow p\in\delta(q,a)$

Lenguaje aceptado por un Autómata Finito No Determinista.

El lenguaje de un AFND es: $L=\{\omega\in\Sigma^*/\hat{\delta}(q_0,\omega)\cap F\neq\varnothing\}$

El lenguaje de un AFND es: $L=\{\omega\in\Sigma^*/[q_0,\omega]\mapsto^*[p,\lambda],\ p\in F\}$

Ejemplo: Sea $M=(\{q_0,q_1,q_2\},\{0,1\},\delta,q_0,\{q_2\})$ del ejemplo anterior.

Como $\hat{\delta}(q_0,00101)=\{q_0,q_2\}\cap F=\{q_2\}$, entonces $00101\in L(M)$

Equivalencia entre AFD y AFND.

Todo lenguaje que puede ser reconocido por un AFND puede ser reconocido por un AFD y viceversa.

Construcción de subconjuntos.

Sea $N=(Q_N,\Sigma,\delta_N,q_0,F_N)$ un AFN, y sea $D=(Q_D,\Sigma,\delta_D,\{q_0\},F_D)$, donde además:

- $Q_D=P(Q_N)$
- $F_D=\{S\in Q_D|S\cap F_N\neq\emptyset\}$
- $\forall a\in\Sigma,\forall S\subseteq Q_N:\delta_D(S,a)=\bigcup_{p\in S}\delta_N(p,a)$

Ejemplo: Sea $M=(\{q_0,q_1,q_2\},\{0,1\},\delta,q_0,\{q_2\})$ del ejemplo anterior.

El autómata $D=(Q_D,\Sigma,\delta_D,\{q_0\},F_D)=\langle P(Q_M),\{0,1\},\delta_D,\{q0\},F_D\rangle$

Entonces:

$Q_D=P(Q_M)=\{\varnothing,\{q_0\},\{q_1\},\{q_2\},\{q_0,q_1\},\{q_0,q_2\},\{q_1,q_2\},\{q_0,q_1,q_2\}\}$

$F_D=\{\{q_2\},\{q_0,q_2\},\{q_1,q_2\},\{q_0,q_1,q_2\}\}$

Y la tabla que describe δ_D resulta:

δ_D	*0*	*1*
$\varnothing$	$\varnothing$	$\varnothing$
$\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_1\}$	$\varnothing$	$\{q_2\}$
$*\{q_2\}$	$\varnothing$	$\varnothing$
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$
$*\{q_1,q_2\}$	$\varnothing$	$\{q_2\}$
$*\{q_0,q_1,q_2\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$

Algunos estados son inaccesibles desde el estado inicial, por lo que podrían eliminarse, quedando finalmente:

$D=(\{\{q_0\},\{q_2\},\{q_0,q_1\},\{q_0,q_2\}\},\{0,1\},\delta_D,\{q_0\},\{\{q_2\},\{q_0,q_2\}\})$

δ_D	*0*	*1*
$\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$

Se puede hacer una Construcción con evaluación "perezosa (lazy)" de la siguiente manera.

Algoritmo:

Entrada: $P(Q_N)$ y $Q_D=\varnothing$

Salida: Q_D con sólo estados accesibles.

1. Agregar $\{q_0\}$ a Q_D. Entonces $Q_D=\{\{q_0\}\}$
2. $\forall a\in\Sigma, \forall S\in Q_D$, si $\delta_D(S,a)\notin Q_D$, agregar $\delta_D(S,a)$ a Q_D

Teorema: Si $D=(Q_D,\Sigma,\delta_D,\{q_0\},F_D)$ es el AFD construido a partir del AFND $N=(Q_N,\Sigma,\delta_N,q_0,F_N)$ mediante la construcción de subconjuntos, entonces $L(D)=L(N)$

Demostración:

En primer lugar se demuestra $\forall\omega\in\Sigma^*:\hat{\delta}_D(\{q_0\},\omega)=\hat{\delta}_N(q_0,\omega)$ por inducción en la longitud de $|\omega|$

BASE: $|\omega|=0\Rightarrow\omega=\lambda$.

Por definición de $\hat{\delta}_D$ para un AFD, $\hat{\delta}_D(\{q_0\},\lambda)=\{q_0\}$.

Por definición de $\hat{\delta}_N$ para un AFND, $\hat{\delta}_N(\{q_0\},\lambda)=\{q_0\}$.

Entonces $\hat{\delta}_D(\{q_0\},\lambda)=\hat{\delta}_N(\{q_0\},\lambda)$

PASO INDUCTIVO:

Sea $|\omega| = h+1 \Rightarrow \omega = \alpha t$, donde t es el símbolo final de ω . Por hipótesis inductiva, $\hat{\delta}_D(\{q_0\},\alpha) = \hat{\delta}_N(q_0,\alpha) = \{p_1, p_2,..., p_k\}$

Por definición de $\hat{\delta}_N$ para un AFND, $\hat{\delta}_N(\{q_0\},\alpha t) = .\bigcup_{i=1}^{k}\delta_N(p_i,t)$

Por definición de $\hat{\delta}_D$ para un AFD,
$\hat{\delta}_D(\{q_0\},\alpha t) = \delta_D(\hat{\delta}_D(\{q_0\},\alpha),t) = \delta_D(\{p_1,p_2,...,p_k\},t)$.

Que por la construcción de subconjuntos, resulta ser:
$\delta_D(\{p_1,p_2,...,p_k\},t) = \bigcup_{i=1}^{k}\delta_N(p_i,t)$

Por lo tanto, queda demostrado que $\forall\omega \in \Sigma^* : \hat{\delta}_D(\{q_0\},\omega) = \hat{\delta}_N(q_0,\omega)$

Ahora demostraremos que $\omega \in L(N) \Leftrightarrow \omega \in L(D)$

$\omega \in L(N) \Leftrightarrow \hat{\delta}_N(q_0,\omega) \cap F_N \neq \emptyset$, es decir, $\hat{\delta}_N(q_0,\omega)$ contiene algún estado final.

Por lo demostrado antes, $\hat{\delta}_N(q_0,\omega) = \hat{\delta}_D(\{q_0\},\omega)$, entonces $\hat{\delta}_D(\{q_0\},\omega) \cap F_N \neq \emptyset$ y como por construcción, $F_D = \{S \in Q_D / S \cap F_N \neq \emptyset\}$ resulta $\hat{\delta}_D(\{q_0\},\omega) \in F_D$

Y por definición, $\hat{\delta}_D(q_0,\omega) \in F_D \Leftrightarrow \omega \in L(D)$.

Así se completa la demostración de que $L(N) = L(D)$

Autómatas Finitos No Determinísticos con transiciones lambda(AFND-lambda).

Función de transición para AFND-λ.

En los AFND-lambda la función de transición se define $\delta : Q \times \Sigma \cup \{\lambda\} \to P(Q)$ donde $P(Q)$ denota (todos los subconjuntos de Q) y se pueden tener transiciones en donde no se consume ningún símbolo de la entrada.

Función de transición extendida para un AFND-λ.

La ***función de transición extendida*** $\hat{\delta}$ se redefine para los AFND-λ.

Clausuras respecto de λ de un estado.

De manera informal, la clausura respecto de λ de un estado q es el conjunto de todos los estados a los que se puede llegar desde q siguiendo cualquier camino cuyos arcos estén etiquetados con λ.

Formalmente, $claus_\lambda(q)$ se define recursivamente de la forma siguiente:

BASE: $q \in claus_\lambda(q)$

PASO INDUCTIVO: Si $p \in claus_\lambda(q) \wedge r \in \delta(p,\lambda) \Rightarrow r \in claus_\lambda(q)$

Ejemplo: Sea $M = (\{p,q,r,s\},\{a,b\},\delta,p,\{p,s\})$ cuya función de transición se ve en la tabla:

δ	***a***	***b***	***λ***
$*p$	$\{q\}$		
q	$\{q,r,s\}$	$\{p,r\}$	$\{s\}$
r		$\{p,s\}$	$\{r,s\}$
$*s$			$\{r\}$

$claus_\lambda(p) = \{p\}$
$claus_\lambda(q) = \{q,s,r\}$
$claus_\lambda(r) = \{r,s\}$
$claus_\lambda(s) = \{s,r\}$

Definición de $\hat{\delta}$ por inducción sobre la longitud de la cadena de entrada.

BASE: $\hat{\delta}(q,\lambda) = claus_\lambda(q)$

PASO INDUCTIVO: Dada $\omega = \omega' a$, $\hat{\delta}(q,\omega') = \{p_1, p_2, \ldots, p_k\}$ y $\bigcup_{i=1}^{k} \delta(p_i, a) = \{r_1, r_2, \ldots, r_m\}$,

entonces $\hat{\delta}(q,\omega) = \bigcup_{j=1}^{m} claus_\lambda(r_j)$

Ejemplo: Sea $M = (\{p,q,r,s\},\{a,b\},\delta,p,\{p,s\})$ del ejemplo anterior.

Entonces, $\hat{\delta}(p,'aab') = \{p,r,s\}$

Secuencia de configuraciones para AFND-λ.

Se redefine: $[q,a\omega] \mapsto [p,\omega] \Leftrightarrow p \in \delta(q,a)$

Lenguaje aceptado por un Autómata Finito No Determinista con transiciones λ

El lenguaje de un AFND- λ es: $L = \{\omega \in \Sigma^* / \hat{\delta}(q_0,\omega) \cap F \neq \varnothing\}$

El lenguaje de un AFND es: $L = \{\omega \in \Sigma^* / [q_0,\omega] \mapsto^* [p,\lambda], p \in F\}$

Ejemplo: Sea $M = (\{p,q,r,s\},\{a,b\},\delta,p,\{p,s\})$ del ejemplo anterior.

Como $\hat{\delta}(p,'aab') = \{p,r,s\} \cap F = \{p,s\}$, entonces $aab \in L(M)$

Equivalencia entre AFD y AFND-λ.

Eliminación de las transiciones λ.

Todo lenguaje que puede ser reconocido por un AFD puede ser reconocido por un AFND-λ y viceversa.

Construcción.

Sea $E = (Q_E, \Sigma, \delta_E, q_0, F_E)$ un AFND-λ. El AFD equivalente $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$ se define así:

- Q_D es el conjunto de subconjuntos de Q_E, $S \subseteq Q_E$ tales que $S = claus_\lambda(S)$
- $q_D = claus_\lambda(q_0)$
- $F_D = \{S \in Q_D | S \cap F_E \neq \emptyset\}$ Es decir, todos aquellos conjuntos de estados que contienen al menos un estado de aceptación de E.
- Se calcula $\delta_D(S, a)$ para todo $a \in \Sigma$ y todos los conjuntos S pertenecientes a Q_D como sigue:

  a) Sea $S = \{p_1, p_2, ..., p_k\}$

  b) Se calcula $\bigcup_{i=1}^{k} \delta(p_i, a) = \{r_1, r_2, ..., r_m\}$

  c) Luego $\delta_D(S, a) = \bigcup_{j=1}^{m} claus_\lambda(r_j)$

Ejemplo: Sea $M = (\{p, q, r, s\}, \{a, b\}, \delta, p, \{p, s\})$ del ejemplo anterior.

Para el autómata $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$:

$q_D = claus_\lambda(p) = \{p\}$

Se agrega $\{p\}$a Q_D. Por ahora $Q_D = \{\{p\}\}$

Se calcula $\delta_D(\{p\}, x) \forall x \in \Sigma$

En este caso sólo se busca $\delta_E(\{p\}, a) = \{q\}$ y $\delta_E(\{p\}, b) = \{\ \}$ y por lo tanto
$\delta_D(\{p\}, a) = claus\lambda(q) = \{q, r, s\}$ y $\delta_D(\{p\}, b) = claus\lambda(\emptyset) = \emptyset$

Se agregan $\{q, r, s\}$ y $\emptyset$ a Q_D. Ahora $Q_D = \{\{p\}, \{q, r, s\}, \emptyset\}$

Se calcula $\delta_D(\{q, r, s\}, x) \forall x \in \Sigma$

Se busca $\delta_E(\{q\}, a) \cup \delta_E(\{r\}, a) \cup \delta_E(\{s\}, a) = \{q, r, s\}$ y,
$\delta_E(\{q\}, b) \cup \delta_E(\{r\}, b) \cup \delta_E(\{s\}, b) = \{p, r, s\}$ y por lo tanto
$\delta_D(\{q, r, s\}, a) = claus_\lambda(q) \cup claus_\lambda(r) \cup claus_\lambda(s) = \{q, r, s\}$ y
$\delta_D(\{q, r, s\}, b) = claus_\lambda(p) \cup claus_\lambda(r) \cup claus_\lambda(s) = \{p, r, s\}$

Se calcula $\delta_D(\emptyset, x) \forall x \in \Sigma$

Se busca $\delta_E(\emptyset, a) = \emptyset$ y, $\delta_E(\emptyset, b) = \emptyset$ y por lo tanto
$\delta_D(\emptyset, a) = claus_\lambda(\emptyset) = \emptyset$ y $\delta_D(\emptyset, b) = claus_\lambda(\emptyset) = \emptyset$

Se agrega $\{p, r, s\}$ a Q_D. Ahora $Q_D = \{\{p\}, \{q, r, s\}, \emptyset, \{p, r, s\}\}$

Se calcula $\delta_D(\{p, r, s\}, x) \forall x \in \Sigma$

Se busca $\delta_E(\{p\}, a) \cup \delta_E(\{r\}, a) \cup \delta_E(\{s\}, a) = \{q\}$ y,
$\delta_E(\{p\}, b) \cup \delta_E(\{r\}, b) \cup \delta_E(\{s\}, b) = \{p, s\}$ y por lo tanto
$\delta_D(\{p, r, s\}, a) = claus_\lambda(q) = \{q, r, s\}$ y
$\delta_D(\{p, r, s\}, b) = claus_\lambda(p) \cup claus_\lambda(s) = \{p, r, s\}$

Queda entonces:

δ	a	b
$\rightarrow \{p\}$	$\{q,r,s\}$	$\varnothing$
$\{q,r,s\}$	$\{q,r,s\}$	$\{p,r,s\}$
$\varnothing$	$\varnothing$	$\varnothing$
$\{p,r,s\}$	$\{q,r,s\}$	$\{p,r,s\}$

Y $Q_D = \{\{p\},\{q,r,s\},\varnothing,\{p,r,s\}\}$

Con $F_D = \{\{p\},\{q,r,s\},\{p,r,s\}\}$

Equivalencia entre AFD y AFND-λ.

Teorema: Si L es aceptado por algún AFND-λ si y sólo si L es aceptado por algún AFD.

1ª Demostración. Si L es aceptado por algún AFD, entonces es aceptado por un AFN-λ.

Transformamos D en un AFND-λ añadiendo transiciones $\forall q \in Q_D : \delta_E(q,\lambda) = \varnothing$ y $\forall q \in Q_D \forall a \in \Sigma : \delta_D(q,a) = p \Rightarrow \delta_E(q,a) = \{p\}$

Queda: $E = (Q_E, \Sigma, \delta_E, \{q_D\}, F_E)$ con $Q_E = \{\{q\} / q \in Q_D\} \cup \varnothing$ y $F_E = \{\{q\} / q \in F_D\}$

Como las transiciones no han cambiado, toda palabra aceptada en el AFD es aceptada en el AFND-λ

2ª Demostración. Si L es aceptado por algún AFND- λ, entonces aceptado por un AFD.

Sea $E = (Q_E, \Sigma, \delta_E, q_0, F_E)$ un AFND- λ. Aplicamos la construcción anterior para generar el AFD, obteniendo así $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$

Demostraremos por inducción en la longitud de ω que $\hat{\delta}_E(q_0,\omega) = \hat{\delta}_D(q_D,\omega)$

BASE: $|\omega| = 0 \Rightarrow \omega = \lambda$.

Por definición de $\hat{\delta}_E$ para un AFND-λ, $\hat{\delta}_E(\{q_0\},\lambda) = claus_\lambda(q_0)$.

Por otra parte, por la construcción, se tiene que $q_D = claus_\lambda(q_0)$, y por definición de $\hat{\delta}_D$ para un AFD, $\hat{\delta}_D(q_D,\lambda) = q_D = claus_\lambda(q_0)$.

Por lo tanto $\hat{\delta}_E(q_0,\lambda) = \hat{\delta}_D(q_D,\lambda)$

PASO INDUCTIVO: $|\omega| = n+1 \Rightarrow \omega = \alpha t$, donde t es el último símbolo de ω y por hipótesis inductiva se cumple que $\hat{\delta}_E(q_0,\alpha) = \hat{\delta}_D(q_D,\alpha) = \{p_1, p_2,..., p_k\}$

Se calcula $\hat{\delta}_E(q_0,\omega) = \bigcup_{j=1}^{m} claus_\lambda(r_j)$ donde $\{r_1, r_2,..., r_m\} = \bigcup_{i=1}^{k} \delta_E(p_i, t)$.

Por otro lado, se calcula $\hat{\delta}_D(q_D,\omega) = \delta_D(\hat{\delta}_D(q_D,\alpha),t) = \delta_D(\{p_1, p_2,..., p_k\},t)$ que por la construcción es $\delta_D(\{p_1, p_2,..., p_k\},t) = \bigcup_{j=1}^{m} claus_\lambda(r_j)$, donde

$\{r_1, r_2, ..., r_m\} = \bigcup_{i=1}^{k} \delta_E(p_i, t)$. Por lo tanto, queda demostrado que

$\hat{\delta}_E(q_0, \omega) = \hat{\delta}_D(q_D, \omega)$