Autómatas finitos

Alfabeto Σ: conjunto (de *símbolos*) finito, no vacío.

Cadena (de elementos de Σ): sucesión finita de elementos de Σ. Se representa yuxtaponiendo sus elementos. A la cadena de 0 símbolos se le llama **cadena vacía** y la representaremos con el símbolo λ ($\lambda \notin \Sigma$, por supuesto).

$\Sigma^\star$: conjunto de (todas las) cadenas de elementos de Σ.

Lenguaje (sobre Σ): subconjunto de $\Sigma^\star$.

Autómata finito determinista: quintupleta $(S, \Sigma, \delta, \iota, F)$, en que:

- S es un conjunto finito de *estados*.
- Σ es un alfabeto ($S \cap \Sigma = \emptyset$).
- $\delta : S \times \Sigma \to S$ es la *función de transición*.
- $\iota \in S$ es el estado *inicial*.
- $F \subseteq S$ es el conjunto de *estados de aceptación*.

El autómata $(S, \Sigma, \delta, \iota, F)$ *acepta* la cadena $x_1 x_2 \cdots x_n$ si y solo si hay una sucesión de estados $s_0, s_1, \ldots, s_n$ tal que $s_0 = \iota$, $s_n \in F$ y $\delta(s_{j-1}, x_j) = s_j$ para cada $j = 1, \ldots, n$.

La función δ se puede representar mediante un diagrama de transiciones. Interpretando sobre éste, una cadena es aceptada si y solo si los símbolos de la cadena corresponden a un *camino* (sucesión de arcos etiquetados) que lleve del estado inicial a algún estado de aceptación.

Sea $M = (S, \Sigma, \delta, \iota, F)$ un autómata finito determinista. El **lenguaje aceptado** por M es el conjunto

$$L(M) = \{w \in \Sigma^\star : M \text{ acepta a } w\}.$$

A un lenguaje tal se le llama **lenguaje regular**.

Teorema. Para todo alfabeto Σ hay algún lenguaje (sobre Σ) que no es regular (e.d. no hay a.f.d. que reconozca a dicho lenguaje).

Demostración. Basta considerar autómatas M con alfabeto Σ, pues arcos etiquetados con símbolos no en Σ no son usados al procesar cadenas de $\Sigma^\star$. Estos autómatas constituyen un conjunto numerable, lo que se puede ver ordenándolos respecto al número de estados. Pero $\Sigma^\star$ es infinito, por lo que $|\mathcal{P}(\Sigma^\star)| > |\mathbb{N}|$. Entonces, como cada a.f.d. acepta un correspondiente l.r., hay lenguajes que no son aceptados por ningún a.f.d.

Teorema. Si un lenguaje regular tiene cadenas de la forma $x^n y^n$ para n arbitrariamente grande (es decir, para infinidad de n-es), entonces tiene cadenas de la forma $x^m y^n$, con $m \neq n$.

Demostración. Sea M un a.f.d. que acepta al lenguaje en cuestión. El lenguaje debe tener alguna cadena de la forma $x^n y^n$, con n mayor que el número de estados de M. Debido a ésto, el proceso de aceptación de $x^n y^n$ debe visitar algún estado más de una vez durante el reconocimiento de x^n, por lo que en esta parte del proceso debe seguir alguna trayectoria cíclica, digamos de j arcos ($j > 0$). Pero entonces M también acepta la cadena $x^{n+j} y^n$. Haciendo $m = n + j$, tenemos que $m \neq n$ y $x^m y^n \in L(M)$.

Corolario. $\{x^n y^n : n \in \mathbb{N}\}$ no es regular.

Autómata finito indeterminista: quintupleta $(S, \Sigma, \rho, \iota, F)$, en que:

- S es un conjunto finito de *estados*.
- Σ es un alfabeto.
- $\rho \subseteq S \times \Sigma \times S$ es la *relación de transición*.
- $\iota \in S$ es el estado *inicial*.
- $F \subseteq S$ es el conjunto de *estados de aceptación*.

Teorema. Para cada a.f.i. hay un a.f.d. que acepta exactamente el mismo lenguaje.

Demostración. Sea $M = (S, \Sigma, \rho, \iota, F)$ un a.f.i. Definamos el a.f.d. $M' = (S', \Sigma, \delta, \iota', F')$, con $S' = \mathcal{P}(S)$, $\iota' = \{\iota\}$, F' la colección de subconjuntos de S con al menos un estado en F, y $\delta(s', x) = \{s \in S : (u, x, s) \in \rho, u \in s'\}$. El teorema se deriva del siguiente lema, aplicándolo al caso en que s_n es estado de aceptación:

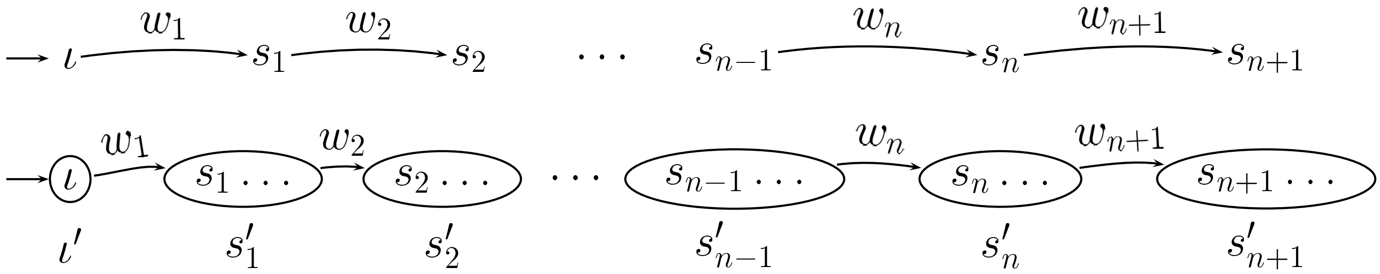
Lema. $w_1 w_2 \cdots w_n$ es camino de ι a s_n (en M) si y solo si hay $s'_n \in S'$ tal que $s_n \in s'_n$ y $w_1 w_2 \cdots w_n$ es camino de ι' a s'_n (en M').

Veamos una demostración del lema, por inducción sobre n:

> $n = 0$) El camino es de longitud 0 (corresponde a λ). Tenemos que $s_0 = \iota$, por lo que es cierta la afirmación "λ es camino de ι a ι". El lado derecho de la bicondicional también es cierto, pues se satisface con $s'_0 = \iota'$. Por lo tanto, la bicondicional resulta cierta.

$n > 0$)

$\Rightarrow$) Consideremos un camino $w_1 w_2 \cdots w_n w_{n+1}$ de ι a s_{n+1}. Sea s_n el estado precedente a s_{n+1}. Por la hipótesis de inducción, hay $s'_n \in S'$ tal que $s_n \in s'_n$ y $w_1 w_2 \cdots w_n$ es camino de ι' a s'_n. Como $(s_n, w_{n+1}, s_{n+1}) \in \rho$, hay algún arco en M', etiquetado por w_{n+1}, de s'_n a un estado que incluye a s_{n+1}. Sea s'_{n+1} este estado. Tenemos entonces que $w_1 w_2 \cdots w_n w_{n+1}$ es camino de ι' a s'_{n+1} en M'.



$\Leftarrow$) Supongamos que hay $s'_{n+1} \in S'$ tal que $s_{n+1} \in s'_{n+1}$ y $w_1 w_2 \ldots w_n w_{n+1}$ es camino de ι' a s'_{n+1}. Sea s'_n el estado precedente a s'_{n+1}, es decir, $\delta(s'_n, w_{n+1}) = s'_{n+1} = \{s \in S : (u, w_{n+1}, s) \in \rho, u \in s'_n\}$. Como $s_{n+1} \in s'_{n+1}$, hay $s_n \in s'_n$ tal que $(s_n, w_{n+1}, s_{n+1}) \in \rho$. Tenemos entonces que $s_n \in s'_n$ y $w_1 w_2 \cdots w_n$ es camino de ι' a s'_n. Aplicando la hipótesis de inducción, obtenemos: $w_1 w_2 \cdots w_n$ es camino de ι a s_n. Por lo tanto, como $(s_n, w_{n+1}, s_{n+1}) \in \rho$, $w_1 w_2 \cdots w_n w_{n+1}$ es camino de ι a s_{n+1}.

<u>Teorema</u>. Para cada alfabeto Σ,

$$\{L(M) : M \text{ a.f.d. de alfabeto } \Sigma\} = \{L(M) : M \text{ a.f.i. de alfabeto } \Sigma\}.$$

Ejemplo de gramática (del lenguaje de identificadores con minúsculas):

```
<identificador> → <letra> <letras o dígitos>
<letras o dígitos> → λ
<letras o dígitos> → <letra>
<letras o dígitos> → <dígito>
<letras o dígitos> → <letra> <letras o dígitos>
<letras o dígitos> → <dígito> <letras o dígitos>
<letra> → a
<letra> → b
 ...
<letra> → z
<dígito> → 0
<dígito> → 1
 ...
<dígito> → 9
```

I → LR	L → b
R → λ	...
R → L	L → z
R → D	D → 0
R → LR	D → 1
R → DR	...
L → a	D → 9

I → aR	R → a	R → aR
I → bR	R → b	R → bR
...	...	...
I → zR	R → z	R → zR
R → λ	R → 0	R → 0R
	R → 1	R → 1R
	...	...
	R → 9	R → 9R

Gramática: cuarteta $G = (V, T, S, R)$, en que:

- V es un alfabeto de *no terminales*.
- T es un alfabeto de *terminales* ($T \cap V = \emptyset$).
- $S \in V$ es el *símbolo inicial*.
- R es un conjunto finito de *reglas de reescritura* o *producciones*.

Derivación de `ba2`:

$$\mathtt{I} \underset{G}{\Rightarrow} \mathtt{LR} \underset{G}{\Rightarrow} \mathtt{bR} \underset{G}{\Rightarrow} \mathtt{bLR} \underset{G}{\Rightarrow} \mathtt{baR} \underset{G}{\Rightarrow} \mathtt{baD} \underset{G}{\Rightarrow} \mathtt{ba2}$$

$\underset{G}{\Rightarrow}$ es la relación en $(V \cup T)^\star$ siguiente. Con $\alpha, \beta, \gamma \in (V \cup T)^\star$, $\alpha A \gamma \underset{G}{\Rightarrow} \alpha\beta\gamma$ si y solo si $A \to \beta$ es producción de G.

$\overset{*}{\underset{G}{\Rightarrow}}$ es la cerradura reflexiva y transitiva de $\underset{G}{\Rightarrow}$. Así, de la derivación antes presentada obtenemos:

$$\mathtt{I} \overset{*}{\underset{G}{\Rightarrow}} \mathtt{ba2}$$

Una gramática *genera* una cadena $\nu \in T^\star$ si y solo si $S \overset{*}{\underset{G}{\Rightarrow}} \nu$ (es decir, si dicha cadena se puede producir a partir del símbolo inicial, reemplazando sucesivamente patrones que aparezcan como lado izquierdo de alguna producción, por el respectivo lado derecho, hasta obtener ν).

Si $T \subseteq \Sigma$, decimos G es una gramática sobre Σ y llamamos **lenguaje generado** por G al conjunto

$$L(G) = \{\omega \in \Sigma^\star : G \text{ genera a } \omega\}.$$

Ejemplo. Con $G = (V, T, S, R)$, $V = \{S, X, Y, Z\}$, $T = \{x, y, z\} \subseteq \Sigma$ y $R = \{S \to XSZ,\ S \to Y,\ Y \to yY,\ Y \to \lambda,\ X \to x,\ Z \to z\}$, tenemos que:

$$L(G) = \{x^m y^n z^m : m, n \in \mathbb{N}\}.$$

Decimos que una gramática es **regular** si para cada producción el lado izquierdo es un símbolo no terminal y el lado derecho es λ o un terminal o un terminal seguido de un no terminal.

Ejemplo. En una gramática regular Se permiten producciones como las siguientes:

`Z` $\to$ `yX`   `Z` $\to$ `x`   `W` $\to$ λ

pero no como las siguientes:

`yW` $\to$ `X`   `X` $\to$ `xZy`   `YX` $\to$ `WvZ`

Cada producción de la forma $N \to x$ puede sustituirse por el par de reglas $N \to xX$ y $X \to \lambda$, en que X es un nuevo símbolo no terminal (que no se use en otras reglas), sin que se altere el lenguaje generado por la gramática original.

$$\begin{aligned} &S \to xX \\ &S \to yY \\ &X \to yY \\ &X \to \lambda \\ &Y \to xX \\ &Y \to \lambda \end{aligned}$$

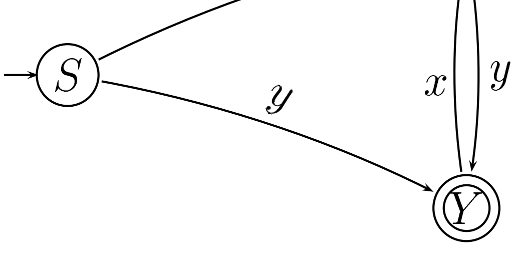


Figura 1.25: Gramática regular G y diagrama de a.f. M, con $L(G) = L(M)$.

<u>Teorema</u>. Para cada alfabeto Σ,

$$\{L(G) : G \,\text{gramática regular sobre}\, \Sigma\} = \{L(M) : M \,\text{a.f. sobre}\, \Sigma\}.$$

Demostración.

$\subseteq$) Sea $G = (V, T, S, R)$ gramática regular sobre Σ. Podemos suponer que G no tiene reglas de la forma $N \to x$. Definimos el a.f.i. $M = (\mathcal{S}, \Sigma, \rho, , \iota, F)$ de tal manera que $\mathcal{S} = V$, $\iota = S$, $F = \{Q \in V : Q \to \lambda \in R\}$ y $(P, x, Q) \in \rho$ si $P \to xQ$ en R. Una derivación de $x_1 x_2 \cdots x_n$ tiene que ser de la forma:

$$S \Rightarrow x_1 X_1 \Rightarrow x_1 x_2 X_2 \Rightarrow \cdots \Rightarrow x_1 x_2 \ldots x_n X_n \Rightarrow x_1 x_2 \ldots x_n$$

obtenida mediante la aplicación sucesiva de las reglas

$$S \to x_1 X_1,\ X_1 \to x_2 X_2, \ldots, X_{n-1} \to x_n X_n,\ X_n \to \lambda.$$

De aquí que, por la definición de M,

$$(S, x_1, X_1), (X_1, x_2, X_2), \ldots, (X_{n-1}, x_n, X_n) \in \rho \quad \text{y} \quad X_n \in F,$$

o sea que $x_1 x_2 \cdots x_n \in L(M)$.

$\supseteq$) Sea $M = (\mathcal{S}, \Sigma, \rho, \iota, F)$ un a.f.i. Definimos la gramática regular (sobre Σ) $G = (V, T, S, R)$, con $V = \mathcal{S}$, $S = \iota$, $P \to xQ$ en R si $(P, x, Q) \in \rho$, y $Q \to \lambda$ en R si $Q \in F$. De esta menera queda garantizado que si $x_1 x_2 \cdots x_n$ es aceptada por M, siguiendo el camino

$$(S, x_1, X_1), (X_1, x_2, X_2), \ldots, (X_{n-1}, x_n, X_n),$$

entonces $x_1 x_2 \cdots x_n$ es generada por G (mediante la derivación presentada arriba).

Sea Σ un alfabeto y $L_1, L_2 \subseteq \Sigma^\star$. Al lenguaje $L_1 \cup L_2$ también se le representa mediante $L_1 + L_2$.

La *concatenación* de L_1 y L_2 es el lenguaje

$$L_1 \circ L_2 = \{w_1 w_2 : w_1 \in L_1, w_2 \in L_2\},$$

también se representa mediante $L_1 L_2$.

Sea $L \subseteq \Sigma$. La *cerradura de Kleene* de L es el lenguaje

$$L^* = \bigcup_{n \in \mathbb{N}} L^n$$

en que $L^0 = \{\lambda\}$ y $L^{n+1} = L^n L$ $(n \in \mathbb{N})$.

Expresiones regulares (**e.r.**) sobre Σ:

- $\emptyset$ es e.r.
- λ es e.r.
- Cada $x \in \Sigma$ es e.r.
- Si p y q son e.r., entonces $(p \cup q)$ y $(p \circ q)$ son e.r.
- Si p es e.r., entonces p^* es e.r.

Cada e.r. p denota un lenguaje $L(p)$, definido a continuación:

$L(\emptyset) = \emptyset$, $L(\lambda) = \{\lambda\}$, $L(x) = \{x\}$, $L(p \cup q) = L(p) \cup L(q)$, $L(p \circ q) = L(p) \circ L(q)$ y $L(p^*) = L(p)^*$.

Teorema. Los lenguajes regulares sobre Σ son exactamente los representables por expresiones regulares sobre Σ, es decir,

$$\{L(p) : p \text{ e.r. sobre } \Sigma\} = \{L(M) : M \text{ a.f. sobre } \Sigma\}.$$

Propiedades (generales).

$$\{\lambda\} \circ L = L \circ \{\lambda\} = L$$

$$\emptyset \circ L = L \circ \emptyset = \emptyset$$

$$L \circ (L_1 \cup L_2) = (L \circ L_1) \cup (L \circ L_2)$$

$$L \circ \bigcup_{n \in \mathbb{N}} L_n = \bigcup_{n \in \mathbb{N}} (L \circ L_n)$$

Propiedades de exps. regulares, con $p = q \overset{\text{def}}{\Longleftrightarrow} L(p) = L(q)$.

$$p + q = q + p$$

$$(p+q)+r = p+(q+r)$$

$$p + p = p$$

$$p + \emptyset = p$$

$$(pq)r = p(qr)$$

$$\lambda p = p\lambda = p$$

$$\emptyset p = p\emptyset = \emptyset$$

$$p(q+r) = pq + pr$$

$$(p+q)r = pr + qr$$

$$\lambda^\star = \lambda$$

$$\emptyset^\star = \lambda$$

$$pp^\star = p^\star p$$

$$p^\star p^\star = p^\star$$

$$(p^\star)^\star = p^\star$$

$$(pq^\star)^\star = \lambda + p(p+q)^\star$$

$$(p+q)^\star = (p^\star q^\star)^\star = (p^\star + q^\star)^\star$$

$$p^\star = \lambda + pp^\star$$

$$(pq)^\star p = p(qp)^\star$$

$$p^\star = \lambda + p + \cdots + p^{k-1} + p^k p^\star \ , \quad (k > 0)$$

$$p^\star = (p^k)^\star(\lambda + p + \cdots + p^{k-1}) \ , \quad (k > 0)$$

$$p = q^\star r \quad \Longrightarrow \quad p = qp + r$$

$$\lambda \notin L(q) \quad \wedge \quad p = qp + r \quad \Longrightarrow \quad p = q^\star r \qquad \text{(regla de Arden)}$$

Se recomienda consultar: Abraham Ginzburg, **Algebraic Theory of Automata**, ACM Monograph Series, 1968. [QA-266-G53, en biblioteca del IIMAS]

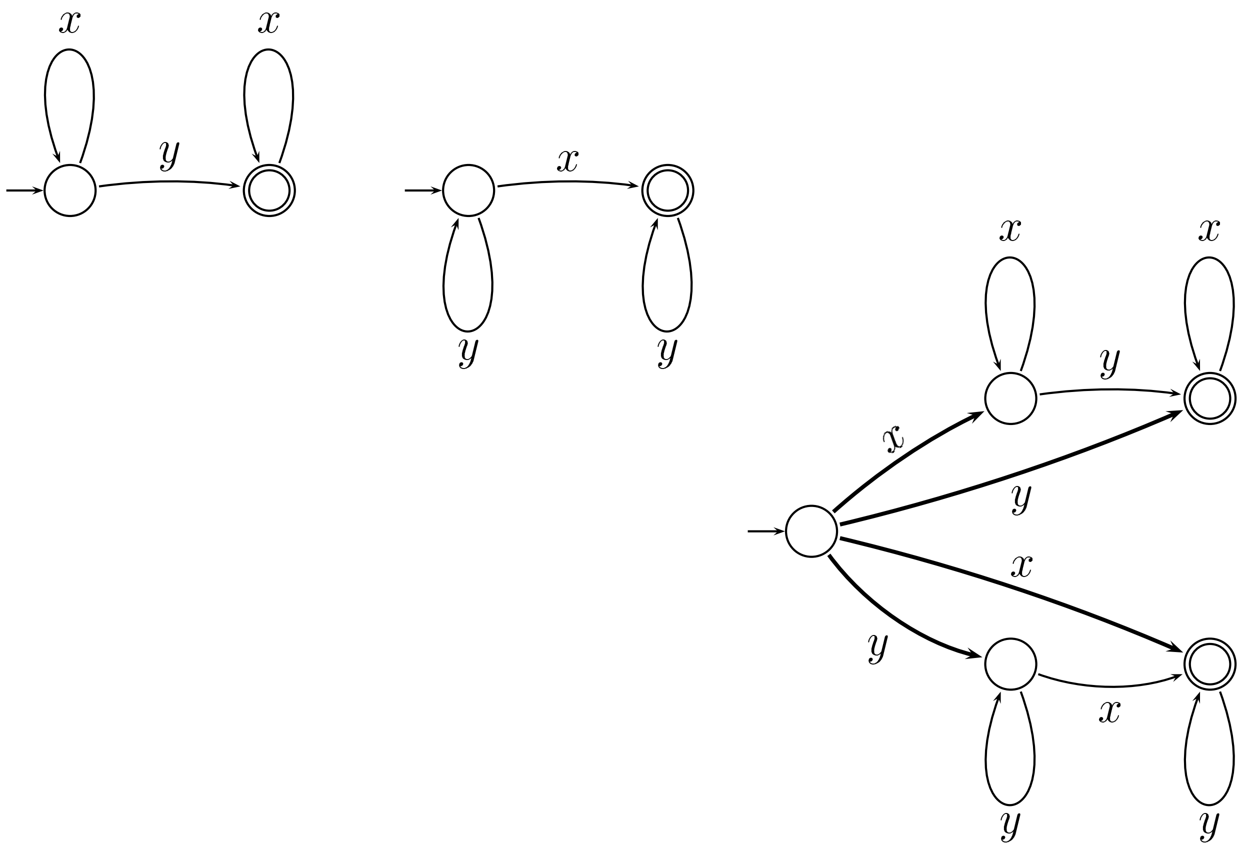


Figura 1.27: Aceptan a L_1, L_2 y $L_1 \cup L_2$, respectivamente.

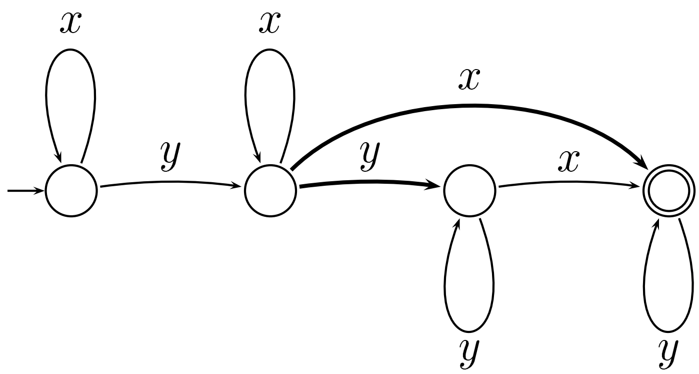


Figura 1.28: Acepta a L_1L_2.

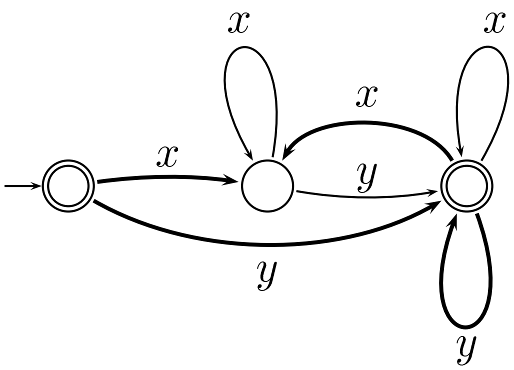


Figura 1.29: Acepta a $L_1^\star$.

Sean $M_1\!=\!(S_1,\Sigma,\rho_1,\iota_1,F_1)$, $M_2\!=\!(S_2,\Sigma,\rho_2,\iota_2,F_2)$, $S_1\!\cap\! S_2\!=\!\emptyset$.

<u>Teorema</u>. La unión de lenguajes regulares es lenguaje regular.

Demostración. Sea $M = (S,\Sigma,\rho,\iota,F)$, con

$$
\begin{aligned}
&S = S_1 \cup S_2 \cup \{\iota\}\\
&\rho = \rho_1 \cup \rho_2 \cup \{(\iota,x,s) : (\iota_1,x,s)\in\rho_1 \vee\ (\iota_2,x,s)\in\rho_2\}\\
&\iota \notin S_1 \cup S_2\\
&F = \begin{cases} F_1 \cup F_2 \cup \{\iota\} & \text{si } \ \iota_1\in F_1 \vee\ \iota_2\in F_2;\\ F_1 \cup F_2 & \text{en caso contrario.}\end{cases}
\end{aligned}
$$

Entonces $L(M) = L(M_1) \cup L(M_2)$.

<u>Teorema</u>. La concatenación de lenguajes regulares es l.r.

Demostración. Sea $M = (S,\Sigma,\rho,\iota,F)$, con

$$
\begin{aligned}
&S = S_1 \cup S_2\\
&\rho = \rho_1 \cup \rho_2 \cup \{(t,x,s) : t\in F_1 \wedge\ (\iota_2,x,s)\in\rho_2\}\\
&\iota = \iota_1\\
&F = \begin{cases} F_1 \cup F_2 & \text{si } \ \iota_2\in F_2;\\ F_2 & \text{en caso contrario.}\end{cases}
\end{aligned}
$$

Entonces $L(M) = L(M_1)L(M_2)$.

<u>Teorema</u>. La cerradura de Kleene de un lenguaje regular es l.r.

Demostración. Sea $M = (S,\Sigma,\rho,\iota,F)$, con

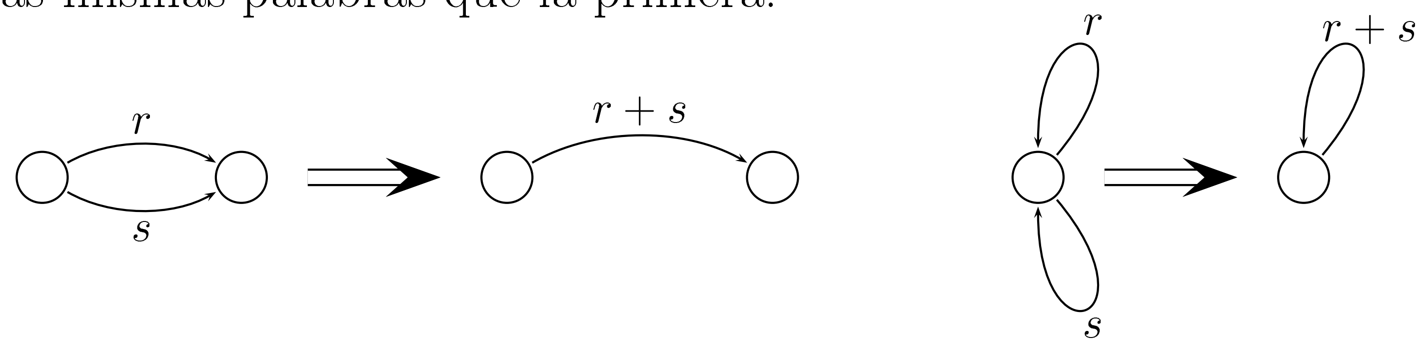
$$
\begin{aligned}
&S = S_1 \cup \{\iota\}\\
&\rho = \rho_1 \cup \{(\iota,x,s) : (\iota_1,x,s)\in\rho_1\} \cup\\
&\qquad\qquad \{(t,x,s) : t\in F_1 \wedge\ (\iota_1,x,s)\in\rho_1\}\\
&\iota \notin S_1\\
&F = F_1 \cup \{\iota\}
\end{aligned}
$$

Entonces $L(M) = L(M_1)^\star$.

Una *gráfica de transiciones* sobre Σ es una gráfica dirigida en que cada arco tiene como etiqueta una palabra $w \in \Sigma^\star$, con al menos un vértice *inicial*. Una palabra w es *aceptada* por la gráfica si hay un camino $w_1, \ldots, w_n$ del vértice inicial a un vértice final, tal que $w = w_1 \cdots w_n$.

Una *gráfica de transiciones generalizada* sobre Σ es una gráfica dirigida en que cada arco tiene como etiqueta una expresión regular (sobre Σ). Una palabra w es *aceptada* por la gráfica si hay un camino $r_1, \ldots, r_n$ del vértice inicial a un vértice final, tal que $w = w_1 \cdots w_n$, con $w_1 \in L(r_1), \ldots, w_n \in L(r_n)$ [1].

Si una gráfica de transiciones generalizada tiene dos arcos r y s, con los mismos vértices de salida e incidencia, los arcos se pueden sustituir por uno solo, $r + s$, obteniéndose una gráfica que acepta las mismas palabras que la primera.



Teorema. $\{L(p) : p \text{ e.r. sobre } \Sigma\} = \{L(M) : M \text{ a.f. sobre } \Sigma\}$.

Demostración. $\subseteq$) Los lenguajes denotados por $\emptyset$, λ y x ($x \in \Sigma$) son regulares porque son finitos. De aquí que, por el teorema anterior, las expresiones regulares de la forma $p+q$, pq ó $p^\star$ también deben representar conjuntos regulares.

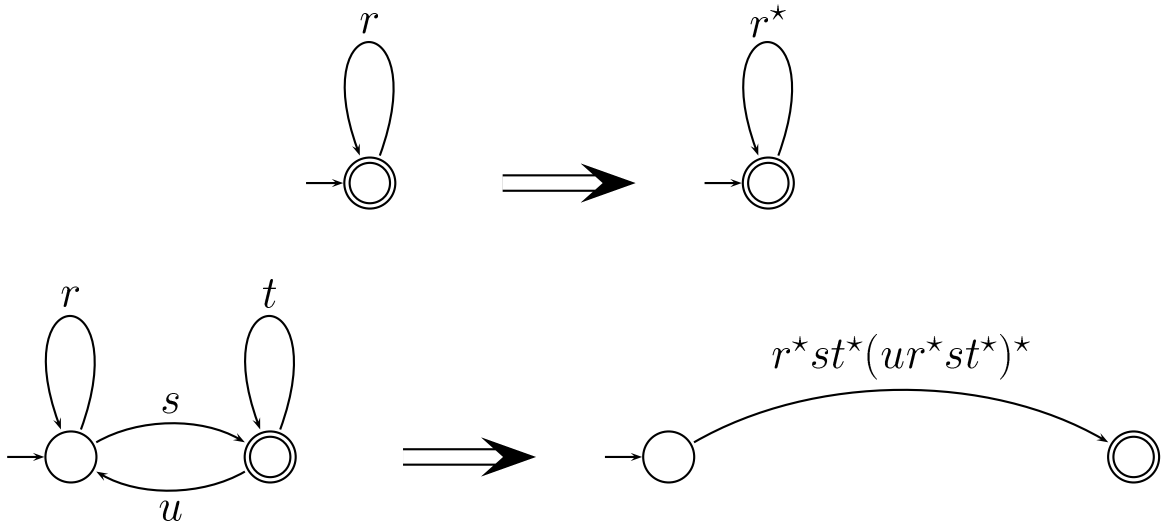
[1] Las definiciones de gráficas de transición se tomaron, adaptándolas para el curso, de: Zohar Manna, **Mathematical Theory of Computation**, McGraw-Hill Computer Science Series, 1974. [QA-76.6-M34, en biblioteca del IIMAS]

$\supseteq$) Sea $M = (S, \Sigma, \rho, \iota, F)$ un a.f. Como $L(M)$ se puede descomponer así:

$$L(M) = \bigcup_{t \in F} \{w \in \Sigma^\star : w \text{ es camino de } \iota \text{ a } t\},$$

por lo que basta considerar autómatas con solo un estado de aceptación t. El diagrama de transiciones de M puede verse como una gráfica de transiciones generalizada. Probemos por inducción sobre $n = |S - \{\iota, t\}|$ que esta gráfica puede transformarse en otra que acepta las mismas palabras y con un solo arco, cuya etiqueta será una e.r. que denotará al lenguaje aceptado.

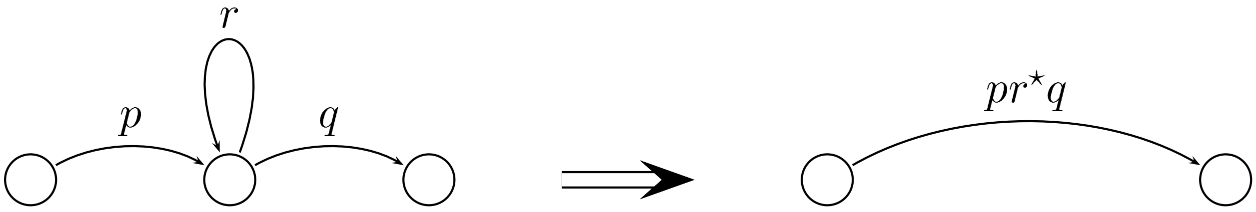
$n = 0$) La gráfica solo tiene vértices ι y t. Sea que $\iota = t$ o no, se puede obtener una de las dos gráficas mostradas a la izquierda en la siguiente figura; la gráfica obtenida puede a su vez ser transformada en la que se muestra a su derecha:



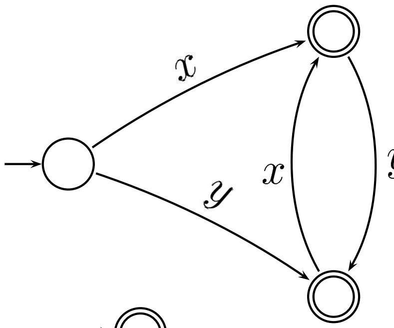
Si cualquier arco no apareciera en la gráfica, trátese como si su etiqueta fuera $\emptyset$. Ejemplos para $s = \emptyset$, $u = \emptyset$ y $r = \emptyset$:

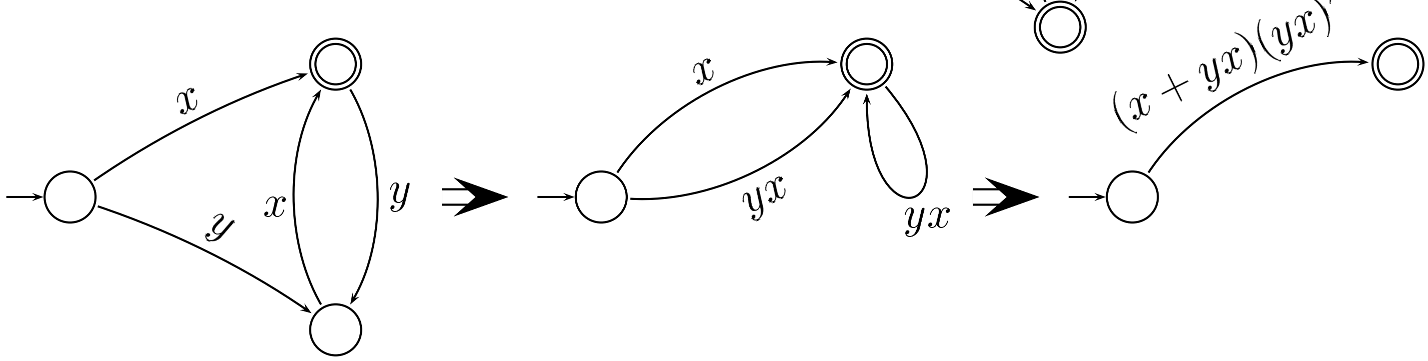
$$r^\star st^\star(ur^\star st^\star)^\star = r^\star \emptyset t^\star(ur^\star \emptyset t^\star)^\star = \emptyset$$
$$r^\star st^\star(ur^\star st^\star)^\star = r^\star st^\star(\emptyset r^\star st^\star)^\star = r^\star st^\star \emptyset^\star = r^\star st^\star \lambda = r^\star st^\star$$
$$r^\star st^\star(ur^\star st^\star)^\star = \emptyset^\star st^\star(u\emptyset^\star st^\star)^\star = \lambda st^\star(u\lambda st^\star)^\star = st^\star(ust^\star)^\star$$

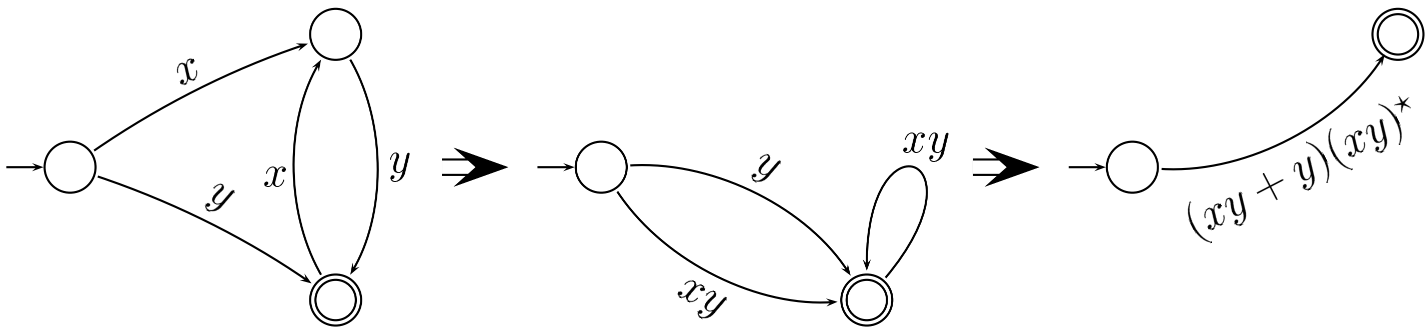
$n > 0$) Al menos hay un vértice v distinto a ι y t. Dicho vértice se puede eliminar de la siguiente manera, sin afectar al lenguaje aceptado por la gráfica original. Para cada pareja de arcos p y q, con p de $v' \neq v$ a v y q de v a $v'' \neq v$, agréguese el arco $pr^\star q$ de v' a v'', como se ilustra enseguida:



Ejemplo. Consideremos la gráfica







El lenguaje aceptado por la gráfica es el denotado por la e.r.

$$(x+yx)(yx)^\star + (xy+y)(xy)^\star$$