13068. Teoria d'Autòmats i Llenguatges Formals

2. Autòmats Finits i Conjunts Regulars

Francesc J. Ferri
Dept. d'Informàtica. Universitat de València

30 d'octubre de 2002

El problema de l'anàlisi

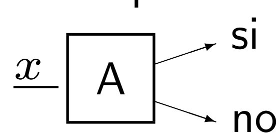
Donada una cadena x i una descripció d'un llenguatge L, $x \in L$?

Enfocament operacional $\exists$ algorisme?

Enfocament lògic es pot demostrar?

$\exists$ algun model de càlcul per al qual es puga construir un "programa" que responga $x \in L$?

- model de càlcul $\equiv$ autòmat
- "programa" $\equiv$ descripció d'un autòmat particular.



Autòmats

• Un exemple d'autòmat podria ser un computador i un autòmat particular d'aquest tipus seria cada programa concret quan està executant-se.

• Preguntes:

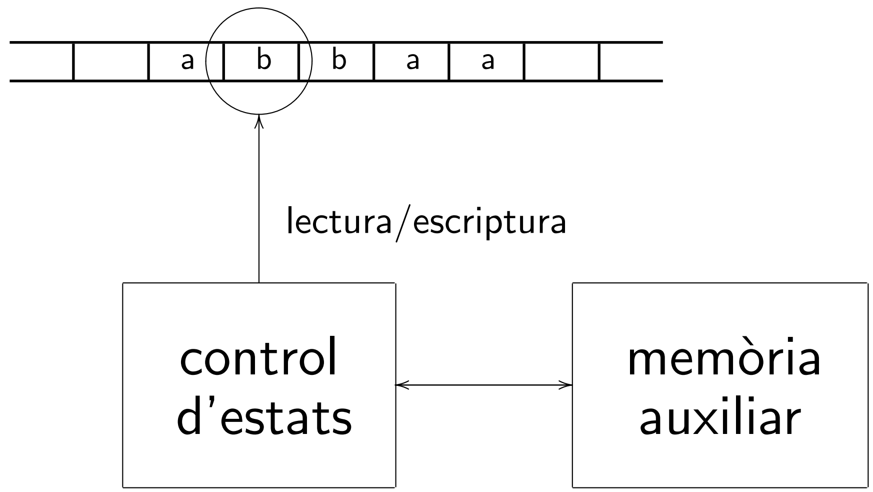
Existeixen descripcions (“programes”) capaços de acceptar/decidir tot llenguatge?

Hi ha tipus d'autòmats capaços de decidir famílies diferents de llenguatges?

Observació: Una descripció (finita) d'un autòmat és una forma alternativa de definir un llenguatge.

Models abstractes de càlcul

Models basats en estats o en control finit



- En cada estat es produeix un cicle:

lectura-escriptura-actualització(mem.)-moviment-canvi(estat)

[Models abstractes de càlcul]

El funcionament de qualsevol autòmat es pot descriure mijantçant una *funció de transició*.

$$\delta\,(\underbrace{e}_{\text{estat}},\ \underbrace{s}_{\text{símbol}},\ \underbrace{m}_{\substack{\text{configuració}\\ \text{de la}\\ \text{memòria}}})=(\underbrace{e'}_{\substack{\text{estat}\\ \text{següent}}},\ \underbrace{s'}_{\substack{\text{nou}\\ \text{símbol}}},\ \underbrace{m'}_{\substack{\text{actualització}\\ \text{de la}\\ \text{memòria}}},\ \underbrace{\mathrm{D}}_{\substack{\text{direcció}\\ \text{de}\\ \text{moviment}}})$$

Com que tant els estats com els símbols com les possibles "configuracions" de la memòria seran finits, la descripció d'un autòmat particular serà sempre finita. La representarem mijantçant una taula de transicions:

	símbols $\times$ configuracions
estats	*transicions*

[Models abstractes de càlcul]

Si però, què calcula un autòmat?

- El tipus de càlcul que més estudiarem és la decisió o acceptació de cadenes.

- El resultat de la computació és SI/NO.

- Normalment, distingirem alguns dels estats com **d'acceptació** i definirem, també, algun criteri de parada.

- Si l'autòmat para en un estat d'acceptació contesta SI. Si no NO.

Autòmats finits

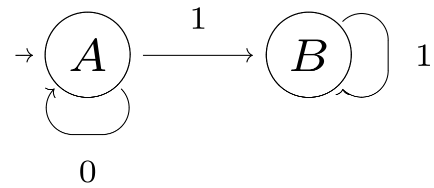
Autòmat més senzill que estudiarem:

- El capçal sempre es mou a la dreta.
- L'autòmat para quan s'acaben els símbols.
- No hi ha memòria auxiliar.
- No es poden escriure símbols.

$$\delta\,(\underbrace{e}_{\text{estat}},\ \underbrace{s}_{\text{símbol}}) = \underbrace{e'}_{\substack{\text{estat}\\\text{següent}}}$$

Exemple

	0	1
$\rightarrow A$	A	B
B	-	B



- Designarem un estat com a “inicial”.

- L'autòmat va llegint símbols i va canviant d'estat segons δ.

- Si en algun moment δ no està definida l'autòmat aborta la computació i contesta NO.

- Aquest autòmat accepta les cadenes de $\{0^n 1^m | n, m \geq 0\}$

Autòmat finit. Definició

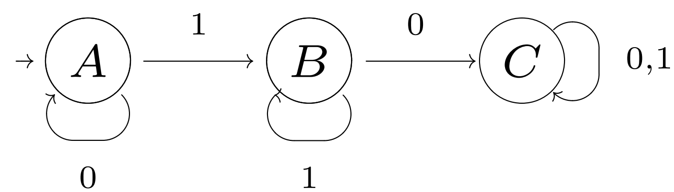
Un **autòmat finit**, A, és una estructura de la forma $\langle Q, \Sigma, \delta, q_0, F \rangle$ on

- Q (conjunt finit d'estats) $\{A, B\}$
- Σ (alfabet d'entrada) $\{0, 1\}$
- $q_0 \in Q$ (estat inicial) A
- $\delta : Q \times \Sigma \longrightarrow Q$ (funció de transició) $\delta(A, 0) = A$, $\delta(A, 1) = \delta(B, 1) = B$
- $F \subseteq Q$ (estats finals o d'acceptació) $\{A, B\}$

Autòmats complets

L'autòmat de l'exemple és equivalent a:

	0	1
$\rightarrow A$	A	B
B	C	B
C	C	C



• Si la funció δ és total, l'anomenare autòmat complet.

• Un estat no final amb transicions a ell mateix $\forall a \in \Sigma$ s'anomena estat d'error.

Funcions de transició

Funció de transició sobre cadenes: el resultat d'aplicar δ sobre cada símbol de la cadena d'esquerra a dreta i sobre l'estat resultant de cada aplicació.

$$\hat{\delta} : Q \times \Sigma^{\star} \longrightarrow Q$$

$$\begin{cases} \hat{\delta}(q, \varepsilon) = q & \forall q \in Q \\ \hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a) & \forall a \in \Sigma, w \in \Sigma^{\star}, q \in Q \end{cases}$$

$$\delta(A, 001) = \delta(\delta(A, 00), 1) = \delta(\overbrace{\delta(\underbrace{\delta(A, 0)}_{A}, 0)}^{A}, 1) = B$$

$$\delta_A(A, 010) = \text{indefinit}$$

$$\delta_{A'}(A, 010) = C$$

Configuració i moviments

Configuració (descripció instantània): "fotografia" de l'autòmat en un moment donat.

- estat actual
- símbols que encara no ha llegit (subcadena pendent).

$$(q, w)$$

Moviment: canvi en la configuració produït per l'aplicació d'una transició
$(q, aw) \longmapsto (q', w) \iff \delta(q, a) = q'$

Exemple: $(A, 001) \underset{A}{\longmapsto} (A, 01) \underset{A}{\longmapsto} (A, 1) \underset{A}{\longmapsto} (B, \varepsilon)$

Abreviadament: $(A, 001) \overset{\star}{\underset{A}{\longmapsto}} (B, \varepsilon)$

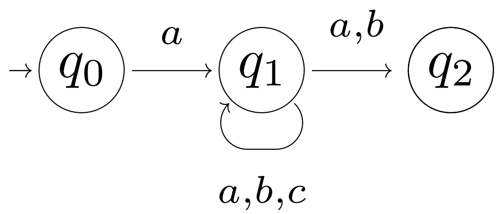
Llenguatge acceptat per un autòmat

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

$$L(A) = \{w \in \Sigma^\star \mid (q_0, w) \overset{\star}{\vdash} (q, \varepsilon), q \in F\}$$

$$L(A) = \{w \in \Sigma^\star \mid \delta(q_0, w) \in F\}$$

Autòmats finits indeterministes



	a	b	c
$\rightarrow q_0$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\{q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_1\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$

- $\delta : Q \times \Sigma \longrightarrow 2^Q$

$$(q_0, aabcb) \vdash (q_1, abcb) \vdash \begin{cases} (q_1, bcb) \vdash \begin{cases} (q_1, cb) \vdash (q_1, b) \vdash \begin{matrix} (q_1, \varepsilon) \\ (q_2, \varepsilon) \end{matrix} \\ (q_2, cb) \end{cases} \\ (q_2, bcb) \end{cases}$$

$$(\{q_0\}, aabcb) \vdash (\{q_1\}, abcb) \vdash (\{q_1, q_2\}, bcb) \vdash (\{q_1, q_2\}, cb) \vdash (\{q_1\}, b) \vdash (\{q_1, q_2\}, \varepsilon)$$

[Autòmats finits indeterministes]

- Ara: $\begin{cases} \hat{\delta}(q,\varepsilon) = \{q\} & \forall q \in Q \\ \hat{\delta}(q,wa) = \displaystyle\bigcup_{p\in\hat{\delta}(q,w)} \delta(p,a) & \forall a \in \Sigma, w \in \Sigma^\star, q \in Q \end{cases}$

- A més a més: $\tilde{\delta}(P,w) = \displaystyle\bigcup_{q\in P} \delta(q,w)$

$$
\begin{aligned}
\hat{\delta}(q_0,abcb) &= \bigcup_{p\in\hat{\delta}(q_0,abc)} \delta(p,b) = \delta(q_1,b) = \underline{\{q_1,q_2\}} \\
\hat{\delta}(q_0,abc) &= \bigcup_{p\in\hat{\delta}(q_0,ab)} \delta(p,c) = \delta(q_1,c) \cup \delta(q_2,c) = \{q_1\} \cup \emptyset = \{q_1\} \\
\hat{\delta}(q_0,ab) &= \bigcup_{p\in\hat{\delta}(q_0,a)} \delta(p,b) = \delta(q_1,b) = \{q_1,q_2\} \\
\hat{\delta}(q_0,a) &= \delta(q_0,a) = \{q_1\}
\end{aligned}
$$

Llenguatge acceptat per un AFI

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

$$L(A) = \{w \in \Sigma^\star \mid (\{q_0\}, w) \vdash^{\star} (Q', \varepsilon),\ Q' \cap F \neq \emptyset\}$$

$$L(A) = \{w \in \Sigma^\star \mid \tilde{\delta}(q_0, w) \cap F \neq \emptyset\}$$

Com que es "com si" estiguera en un conjunt finit d'estats sempre hi haurà un AF equivalent on cada estat "serà" un conjunt d'estats.

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

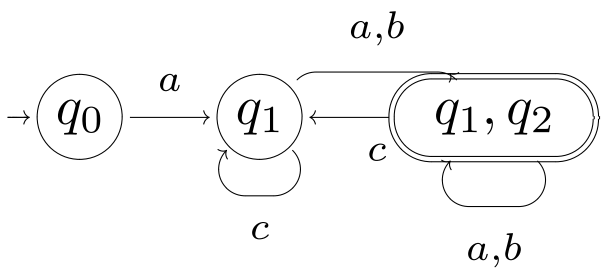
$$A' = \langle 2^Q, \Sigma, \tilde{\delta}, \{q_0\}, F' \rangle$$

$$F' = \{P \in 2^Q \mid P \cap F \neq \emptyset\}$$

Exemple. AFI

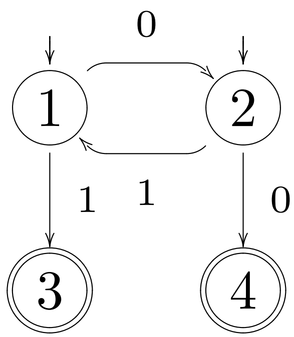
	a	b	c
$\rightarrow\{q_0\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
$\{q_1\}$	$\{q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_1\}$
$\{q_2\}$	$\emptyset$	$\emptyset$	$\emptyset$
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$\{q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_1\}$
...			

3



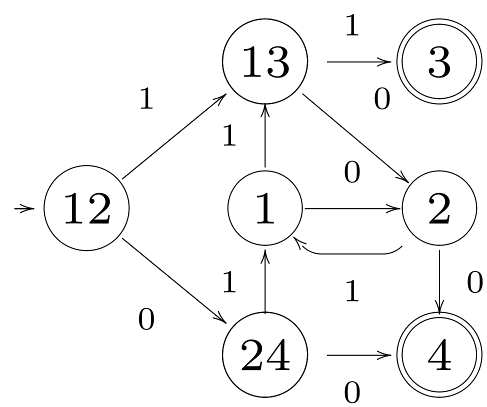
$\{q_2\}$	Estat inabastable

Exemple AFI

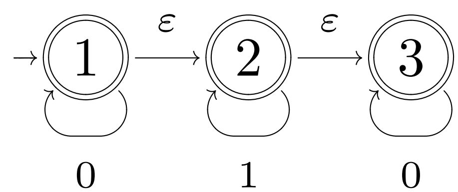


$$(\{1,2\}, 0100) \vdash (\{2,4\}, 100) \vdash (\{1\}, 00) \vdash (\{2\}, 0) \vdash (\{4\}, \varepsilon)$$

	0	1
1	2	3
2	4	1
3	-	-
4	-	-
→ 12	24	13
24	4	1
13	2	3



Autòmats ε



Tancament epsilon:
$1^\bullet = \{1,2,3\}$
$2^\bullet = \{2,3\}$
$3^\bullet = \{3\}$

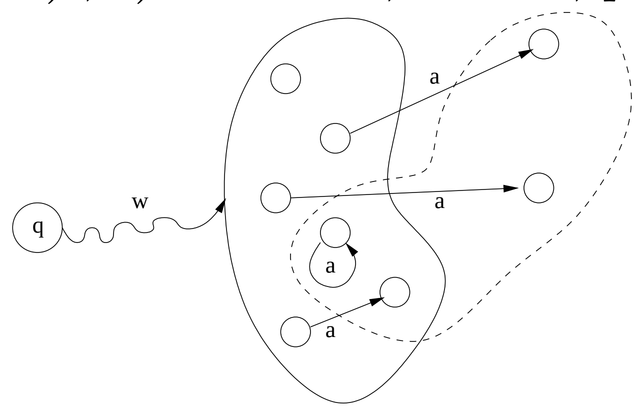
$$\begin{array}{ccccccc}(1^\bullet, 001) & \vdash & (1^\bullet \cup 3^\bullet, 01) & \vdash & (1^\bullet \cup 3^\bullet, 1) & \vdash & (2^\bullet, \varepsilon) \\ (\{1,2,3\}, 001) & \vdash & (\{1,2,3\}, 01) & \vdash & (\{1,2,3\}, 1) & \vdash & (\{2,3\}, \varepsilon)\end{array}$$

$$\begin{cases} \delta'(q,\varepsilon) = q^\bullet & \forall q \in Q \\ \delta'(q,wa) = \delta(\ \delta'(q,w)\ ,a)^\bullet & \forall a \in \Sigma, w \in \Sigma^\star, q \in Q \end{cases}$$

$$\delta'(P,w) = \bigcup_{q \in P} \delta'(q,w)$$

[Autòmats ε]

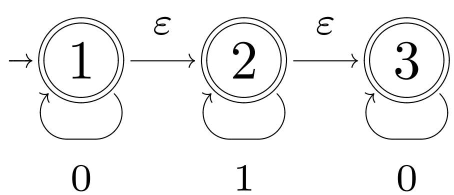
$$\begin{cases} \delta'(q,\varepsilon) = q^{\bullet} & \forall q \in Q \\ \delta'(q,wa) = \delta(\ \delta'(q,w)\ ,a)^{\bullet} & \forall a \in \Sigma, w \in \Sigma^{\star}, q \in Q \end{cases}$$



AFI equivalent a un AFε: El mateix autòmat amb δ'.

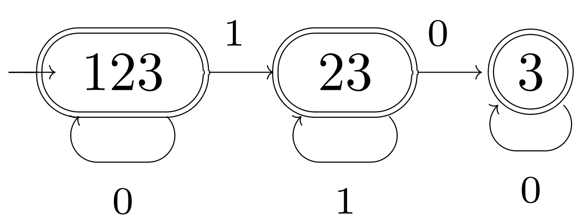
	Σ
Q	$\delta'(q,a) = \delta(q^{\bullet},a)^{\bullet}$

Exemple



	0	1	$q^\bullet$
$\rightarrow$ 1	123	23	123
2	3	23	23
3	3	-	3
23	3	23	
123	123	23	

222 2



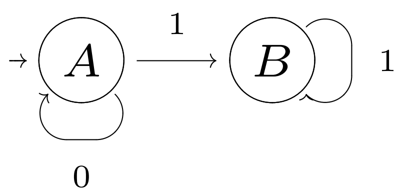
$$1 = 123, \ 2 = 23$$

$\boxed{\text{Estats idèntics}}$

D'autòmat finit a gramàtica regular

Donat un autòmat A, hi ha una gramàtica G tal que $L(G) = L(A)$

$$(A, 001) \quad \vdash \quad (A, 01) \quad \vdash \quad (A, 1) \quad \vdash \quad (B, \varepsilon)$$



$$\delta(A,0)=A \qquad \delta(A,0)=A \qquad \delta(A,1)=B$$

$$A \xRightarrow{A\to 0A} 0A \xRightarrow{A\to 0A} 00A \xRightarrow{A\to 1B} 001B \xRightarrow{B\to\varepsilon} 001$$

estats $\equiv$ no terminals

finals $\equiv$ produccions nul·les

estat inicial $\equiv$ axioma

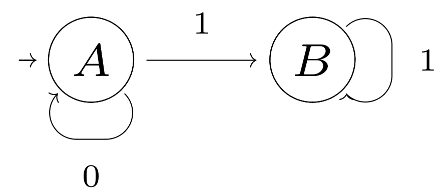
transicions $\equiv$ produccions

[D'autòmat finit a gramàtica regular]

Formalment, donat $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, es defineix

$G_A = \langle V_T, V_N, S, P \rangle$ on

- $V_T = \Sigma$
- $V_N = \{A_q | q \in Q\}$
- $S = A_{q_0}$
- $P = P_1 \cup P_2$
- $P_1 = \bigcup_{\forall a,p,q:\delta(p,a)=q} \{A_p \longrightarrow aA_q\}$
- $P_2 = \bigcup_{q \in Q} \{A_q \longrightarrow \varepsilon\}$



$A \longrightarrow 0A$
$A \longrightarrow 1B$
$B \longrightarrow 1B$

$A \longrightarrow \varepsilon$
$B \longrightarrow \varepsilon$

[D'autòmat finit a gramàtica regular]

Gràficament,

$$p \xrightarrow{a} q \quad \rightsquigarrow \quad A_p \rightarrow aA_q$$

$$q \quad \rightsquigarrow \quad A_q \rightarrow \varepsilon$$

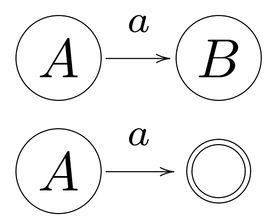
Atenció: l'argument és vàlid per a AFs, AFIs i AFεs

De gramàtica a autòmat finit

Podem aplicar l'anterior mètode al revés!

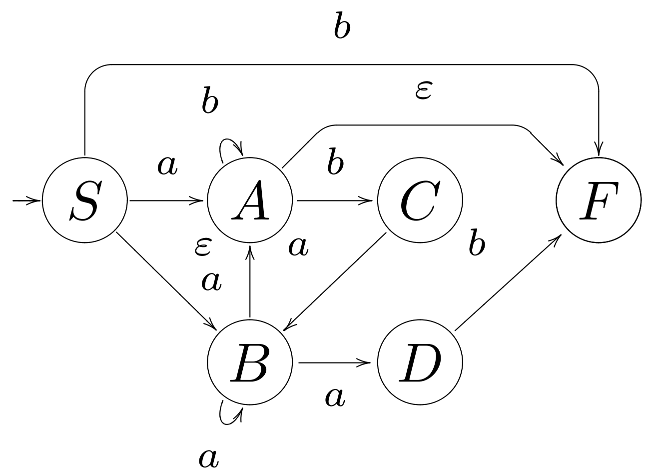
$$A \to aB \quad \rightsquigarrow$$

$$A \to a \quad \rightsquigarrow$$



$$S \to aA|B|b$$
$$A \to bA|baB|\varepsilon$$
$$B \to aA|aB|ab$$

$$A \to baB \equiv A \to bC,\ C \to aB$$
$$B \to ab \equiv B \to aD,\ D \to b$$

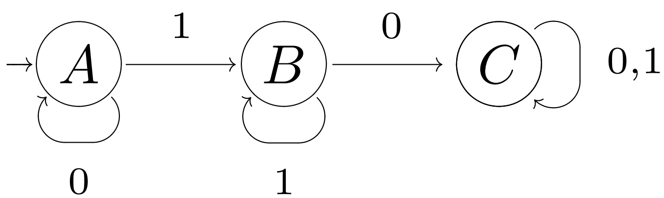


Autòmats finits i llenguatges regulars

Llenguatges regulars $\equiv$ llenguatges acceptats per AF

On estan els llenguatges coregulars?

El complementari de $\{0^n 1^m | n, m \geq 0\}$



(canviant finals per no finals)

$A \rightarrow 0A|1B$
$B \rightarrow 1B|0C$
$C \rightarrow 0C|1C|\varepsilon$

$$\boxed{L \text{ regular} \iff \overline{L} \text{ regular}}$$

Expressions Regulars (ER)

- Aspecte Generatiu: gramàtiques
- Aspecte Operacional: autòmats
- Aspecte Descriptiu: expressions regulars

Expressió regular: Una forma convenient i fàcil de descriure llenguatges regulars

Certes extensions tenen molta aplicació pràctica

ER. definició

Siga Σ un determinat alfabet:

1. ε, $\emptyset$ i a són expressions regulars, $\forall a \in \Sigma$
2. si α i β són expressions regulars, aleshores $\alpha + \beta$ també.
3. si α i β són expressions regulars, aleshores $\alpha \cdot \beta$ o $\alpha\beta$, també.
4. si α és una expressió regular, aleshores $\alpha^\star$ també.
5. si α és una expressió regular, aleshores (α) també.
6. només són expressions regulars les que s'obtenen mijantçant l'aplicació de les regles anteriors.

ER. Semàntica

$L(\alpha)$ és el llenguatge que representa la ER α.

1. $L(\varepsilon) = \{\varepsilon\}$, $L(\emptyset) = \emptyset$, i $L(a) = \{a\}$, $\forall a \in \Sigma$.
2. $L(\alpha + \beta) = L(\alpha) \cup L(\beta)$.
3. $L(\alpha\beta) = L(\alpha) \cdot L(\beta)$.
4. $L(\alpha^\star) = L(\alpha)^\star$.
5. $L((\alpha)) = L(\alpha)^\star$.

Exemples

- $(a+b)^\star$ representa el monoide lliure.
- $a(a+b)^\star$ representa totes les paraules que comencen per a.
- $(a+b)((a+b)(a+b))^\star$ representa totes les paraules de longitud senar.

Anomenarem **conjunt regular** de cadenes tot llenguatge L per al qual existesca una ER α tal que $L = L(\alpha)$.

Quina relació hi ha entre els llenguatges i els conjunts regulars?

Equivalències

$L(\alpha)$ **és regular** $\forall\alpha$

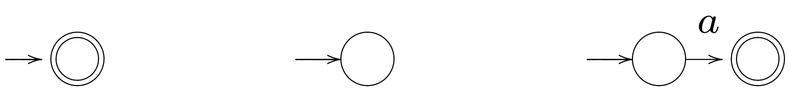
Es pot demostrar per inducció a partir de la definició de ER

Si L **regular aleshores** $\exists\alpha$ **tal que** $L = L(\alpha)$ **?**

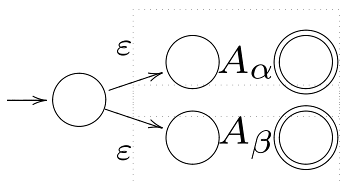
Açò requeriria demostrar que tot llenguatge regular es pot descomposar en un nombre finit d'unions, concatenacions i tancaments a partir de símbols i/o els llenguatges trivials $\emptyset$ i $\{\varepsilon\}$.

Autòmat equivalent a una ER

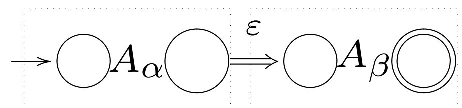
A_α: autòmat equivalent a la ER α.



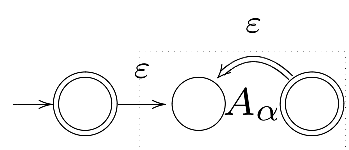
A_ε $A_\emptyset$ A_a



$A_{\alpha+\beta}$



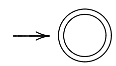
$A_{\alpha\beta}$

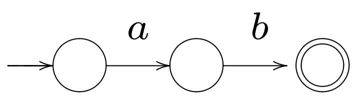


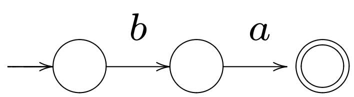
$A_{\alpha^\star}$

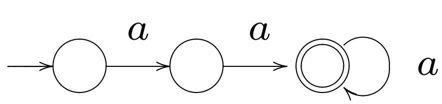
Exemple

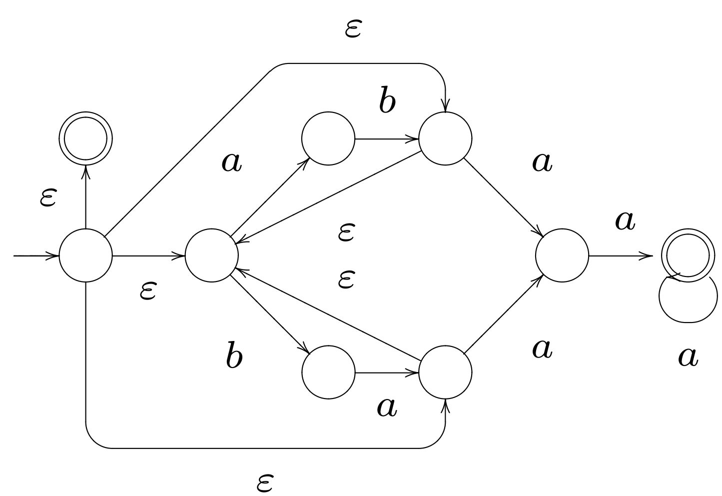
$$\varepsilon + (ab + ba)^{\star} aaa^{\star}$$











Propietats

$$\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma \qquad \alpha + \beta = \beta + \alpha \qquad \alpha + \emptyset = \alpha$$

$$\alpha(\beta\gamma) = (\alpha\beta)\gamma \qquad \alpha(\beta + \gamma) = \alpha\beta + \alpha\gamma \qquad \alpha\varepsilon = \alpha$$

$$(\beta + \gamma)\alpha = \beta\alpha + \gamma\alpha$$

$$\alpha\emptyset = \emptyset \qquad \alpha\alpha^\star = \alpha^\star\alpha = \alpha^+$$

$$(\alpha^\star + \beta^\star)^\star = (\alpha^\star\beta^\star)^\star = (\alpha + \beta)^\star \qquad (\alpha\beta)^\star\alpha = \alpha(\beta\alpha)^\star$$

$$(\alpha^\star\beta)^\star\alpha^\star = (\alpha + \beta)^\star \qquad (\alpha^\star\beta)^\star = (\alpha + \beta)^\star\beta + \varepsilon$$

ER equivalent a un autòmat

Donat un autòmat $A = \langle Q, \Sigma, \delta, q_1, F \rangle$ ...

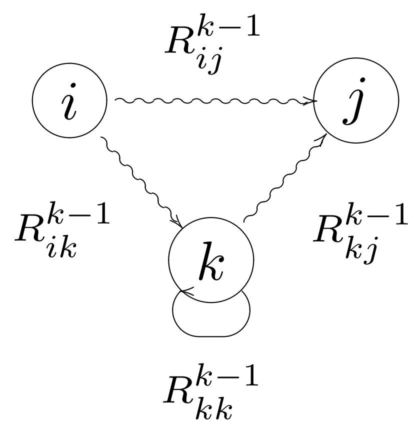
- numerem els estats $q_1, \ldots, q_n$.
- definim R_{ij}^k. Conjunt de cadenes que fan passar l'autòmat de l'estat i al j sense passar per cap estat "major" que k (com a estat intermig!)
- demostrarem que els R_{ij}^k són conjunts regulars.
- $L(A) = \bigcup_{j:q_j \in F} R_{1j}^n$, per tant $L(A)$ serà un conjunt regular.

R_{ij}^k és conjunt regular

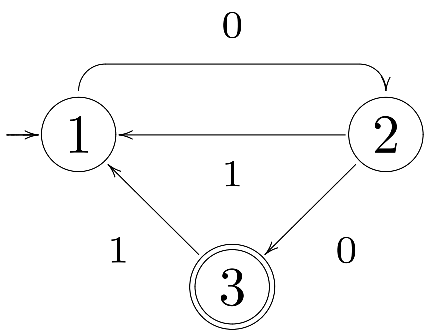
Inductivament,

$$R_{ij}^0 = \{a \in \Sigma \mid q_j \in \delta(q_i, a)\} \cup \{\varepsilon \ \underline{\texttt{si}} \ \ i = j\}$$

$$R_{ij}^k = R_{ij}^{k-1} \cup R_{ik}^{k-1} (R_{kk}^{k-1})^\star R_{kj}^{k-1}$$



Exemple



$R_{11}^{0} = \varepsilon \quad R_{12}^{0} = 0 \qquad R_{13}^{0} = \emptyset \quad R_{31}^{0} = 1$

$R_{12}^{1} = 0 \quad R_{12}^{2} = 0(10)^{\star} \qquad R_{12}^{3} = 0[(1+01)0]^{\star}$

$R_{13}^{1} = \emptyset \quad R_{13}^{2} = 0(10)^{\star}0 \qquad \boxed{R_{13}^{3} = 0[(1+01)0]^{\star}0}$

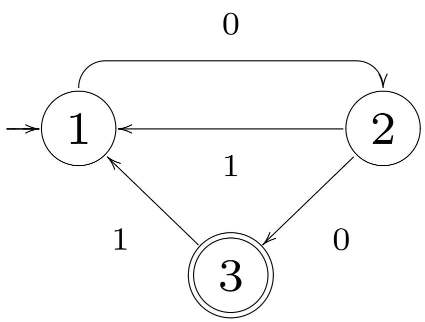
[Exemple]

$$R^3_{13} = R^2_{13} \cup R^2_{13} R^{2\,\star}_{33} R^2_{33} = 0(10)^*0 + 0(10)^*0[(10)^+0]^*[\varepsilon + (10)^+0]$$

$$= 0(10)^*0[\varepsilon + [(10)^+0]^*] = \boxed{0(10)^*0[(10)^+0]^*}$$

$$R^2_{13} = R^1_{13} \cup R^1_{12} R^{1\,\star}_{22} R^1_{23} = \emptyset + 0(10)^*0 = 0(10)^*0$$

$$R^2_{33} = R^1_{33} \cup R^1_{32} R^{1\,\star}_{22} R^1_{23} = \varepsilon + 10(10)^*0 = \varepsilon + (10)^+0$$



$$R_{12} = R_{12} \cup R_{11} R^{0\,\star}_{11} R_{12} = 0 + \varepsilon 0 = 0$$

$$R_{13} = R_{13} \cup R_{11} R^{0\,\star}_{11} R_{13} = \emptyset + \varepsilon\emptyset = \emptyset$$

$$R_{22} = R_{22} \cup R_{21} R^{0\,\star}_{11} R_{12} = \varepsilon + 1\varepsilon 0 = \varepsilon + 10$$

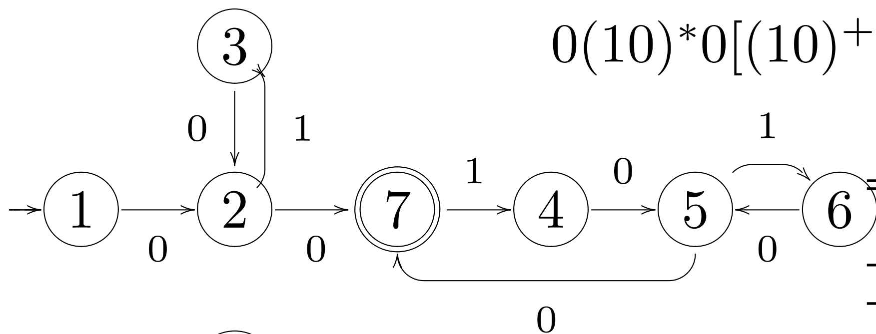
$$R_{23} = R_{23} \cup R_{21} R^{0\,\star}_{11} R_{13} = 0 + 1\varepsilon\emptyset = 0$$

$$\boxed{R^3_{13} = 0[(1+01)0]^\star 0} \; ? \; R_{32} = R_{32} \cup R_{31} R^{0\,\star}_{11} R_{12} = \emptyset + 1\varepsilon 0 = 10$$

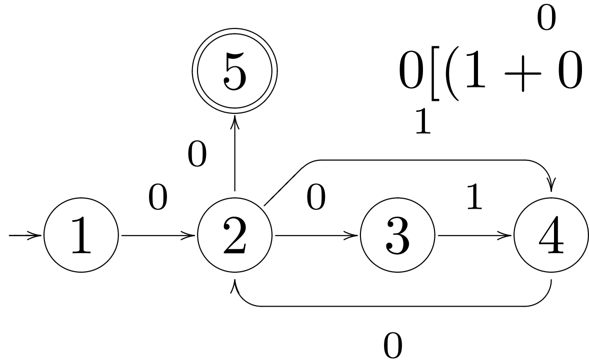
$$R_{33} = R_{33} \cup R_{31} R^{0\,\star}_{11} R_{13} = \varepsilon + 1\varepsilon\emptyset = \varepsilon$$

Son iguals?

$$0(10)^*0[(10)^+0]^*$$



$$0[(1+01)0]^\star 0$$

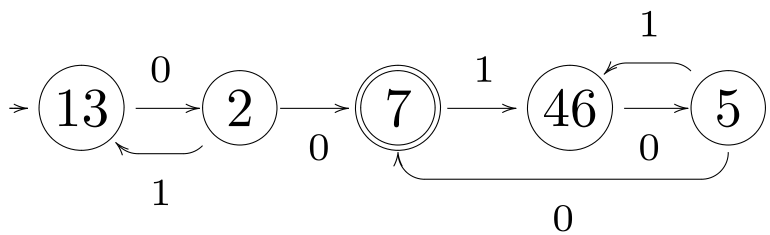


	0	1
→ 1	2	
2	35	4
3		4
4	2	
5		
35		4

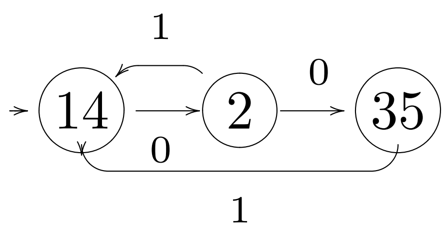
	0	1
→ 1	2	
2	7	3
3	2	
4	5	
5	7	6
6	5	
7		4

[Son iguals?]

$0(10)^*0[(10)^+0]^*$



$0[(1+01)0]^\star 0$

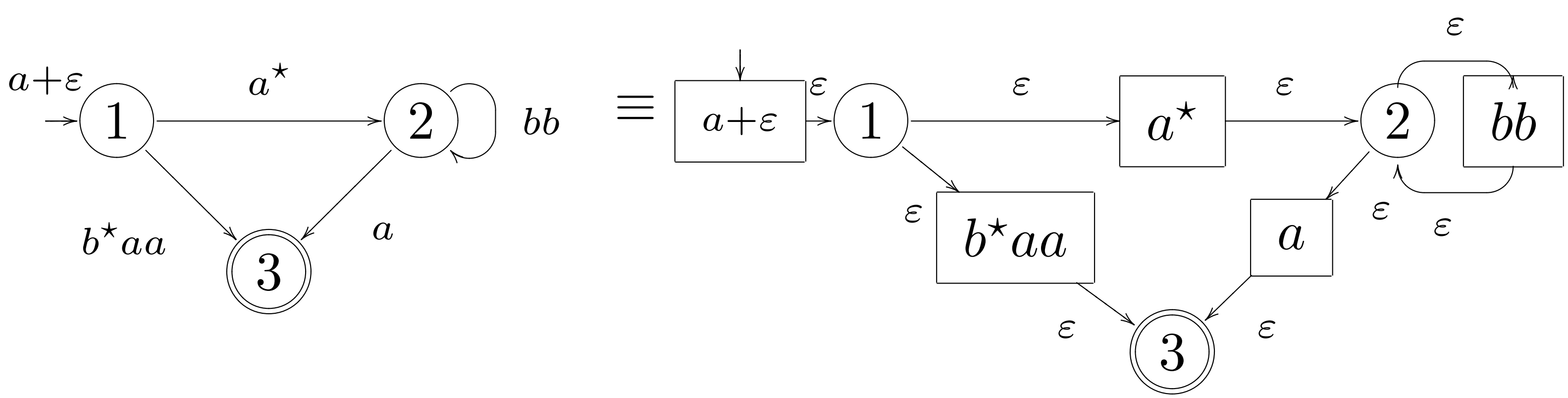


	0	1
→ 1=4	2	
2	35	4
35		4

	0	1
→ 1=3	2	
2	7	3
4=6	5	
5	7	6
7		4

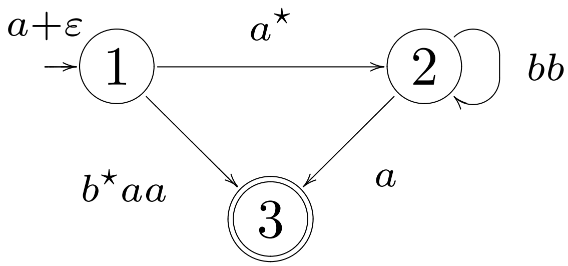
Mètode dels e-autòmats

Un **autòmat d'expressions regulars** (o **e-autòmat**) és un autòmat en el qual les transicions estan etiquetades amb expressions regulars. L'autòmat passa d'un estat a un altre en llegir de la cadena d'entrada qualsevol prefix que estiga en el conjunt regular de la corresponent transició.



e-Autòmats

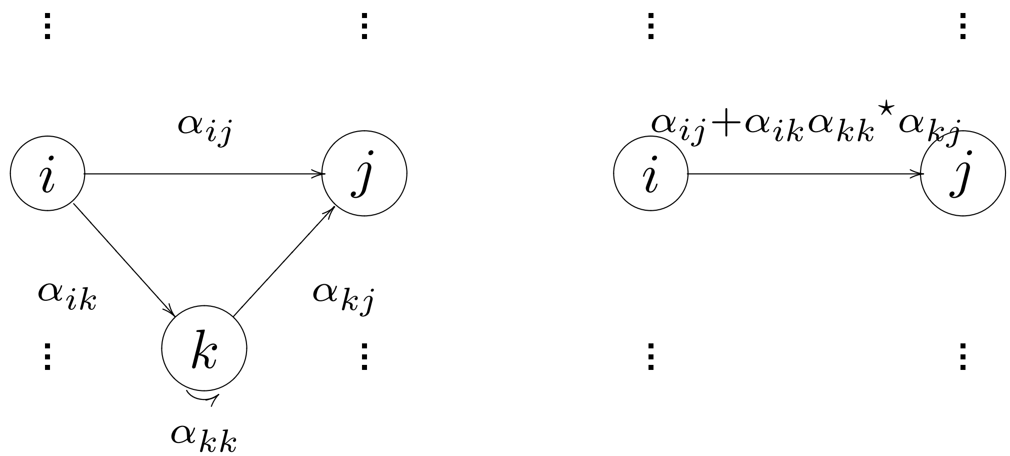
Un e-autòmat es pot representar mijantçant una taula d'expressions regulars.



	1	2	3
0	$a+\varepsilon$	$\emptyset$	$\emptyset$
1	ε	$a^\star$	$b^\star aa$
2	$\emptyset$	$\varepsilon + bb$	a
3	$\emptyset$	$\emptyset$	ε

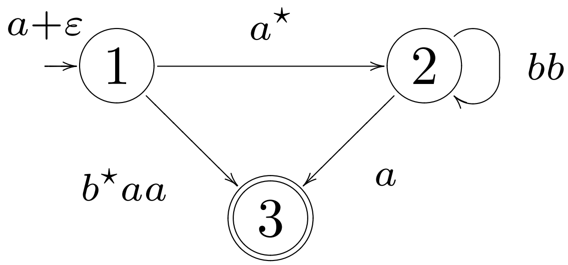
Eliminació d'estats

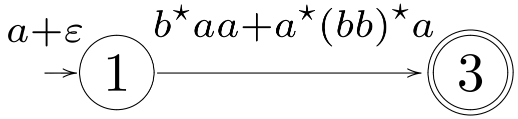
En un e-autòmat sempre es pot eliminar un estat si es recalculen de forma convenient les ER associades a parells d'estats *adjacents* a l'eliminat.

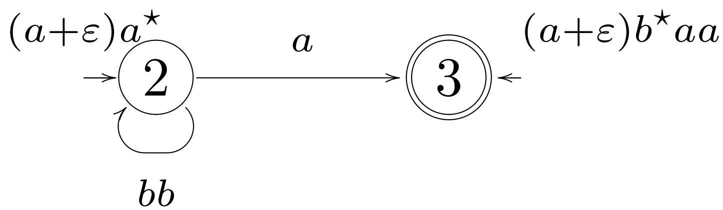


$$\boxed{\text{L'e-autòmat resultant és equivalent (si no eliminem un final!)}}$$

Exemple

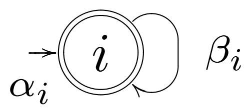

$a+\varepsilon$
$a^\star$
1
2
bb
$b^\star aa$
a
3


$a+\varepsilon$
$b^\star aa+a^\star(bb)^\star a$
1
3


$(a+\varepsilon)a^\star$
a
$(a+\varepsilon)b^\star aa$
2
3
bb

e-autòmats unitaris

Si eliminem tots els estats d'un e-autòmat llevat d'un, s'obté un e-autòmat que representa totes les cadenes que arriben a aquest estat en l'e-autòmat de partida.

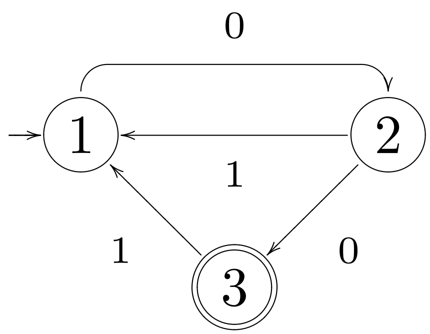


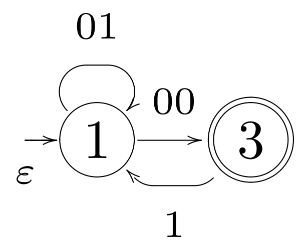
$$\equiv \qquad \alpha_i {\beta_i}^{\star}$$

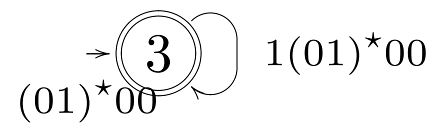
Aquest conjunt de cadenes és exactament R^n_{1i} si l'e-autòmat de partida és un autòmat normal de n estats. Aleshores, considerant només els finals:

$$L(A) = \bigcup_{i \in F} R^n_{1i} = \sum_{i \in F} \alpha_i {\beta_i}^{\star}$$

Exemple







$\boxed{(01)^\star 00[1(01)^\star 00]^\star}$ $\boxed{0(10)^* 0[(10)^+ 0]^*}$ (?) $\boxed{0[(1+01)0]^\star 0}$ (??)

(?) $\underbrace{0(10)^\star}_{(01)^\star 0}\ 0[\underbrace{(10)^\star 1}_{1(01)^\star}\ 00]^\star$

(??) $0[(1+01)0]^\star 0 = [0(1+01)]^\star 00 = [01+001]^\star 00 =$

$$= [(01)^\star 001]^\star (01)^\star 00 = (01)^\star 00[1(01)^\star 00]^\star$$