In questa lezione

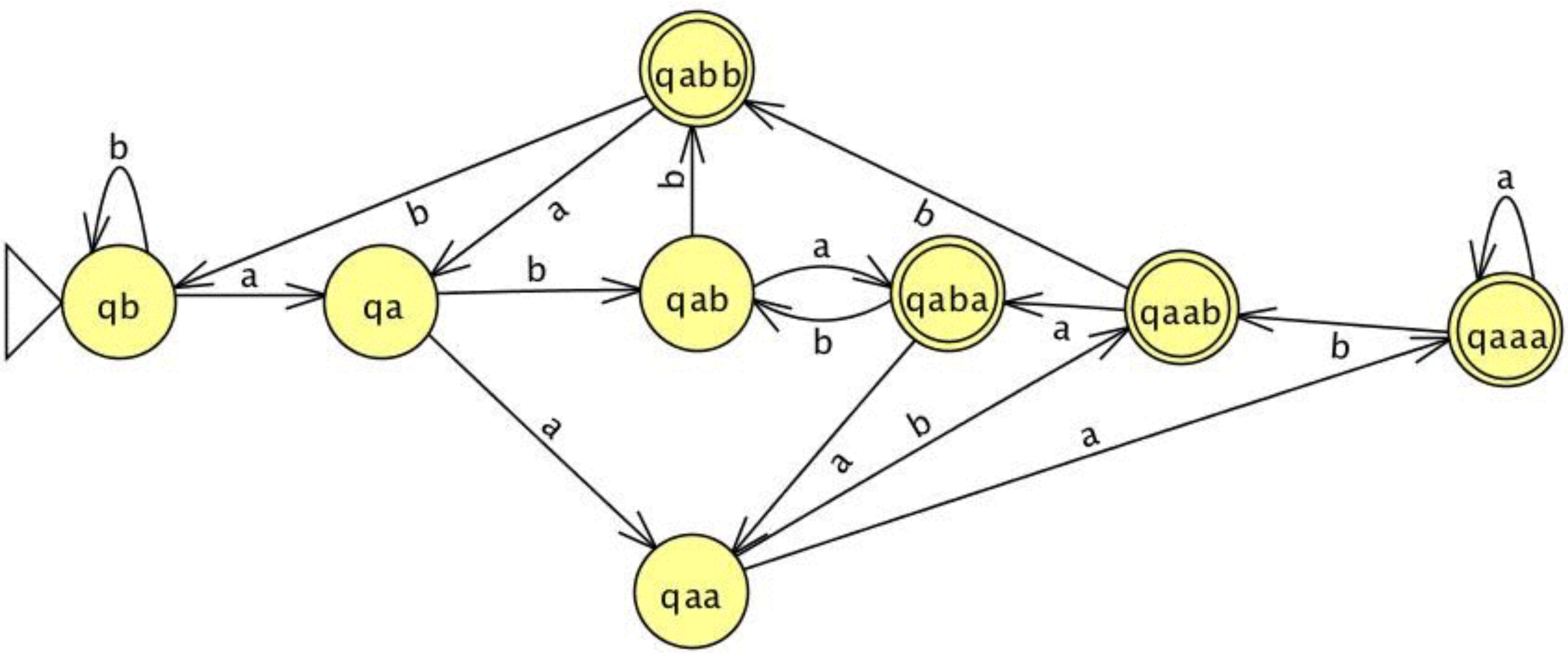
Le operazioni fondamentali sui linguaggi
Proprietà di chiusura

Convenzione: le prime lettere dell’alfabeto inglese, con eventualmente indici, indicano gli elementi dell’alfabeto, le ultime lettere dell’alfabeto indicano le parole o stringhe sull'alfabeto

ESEMPIO 1

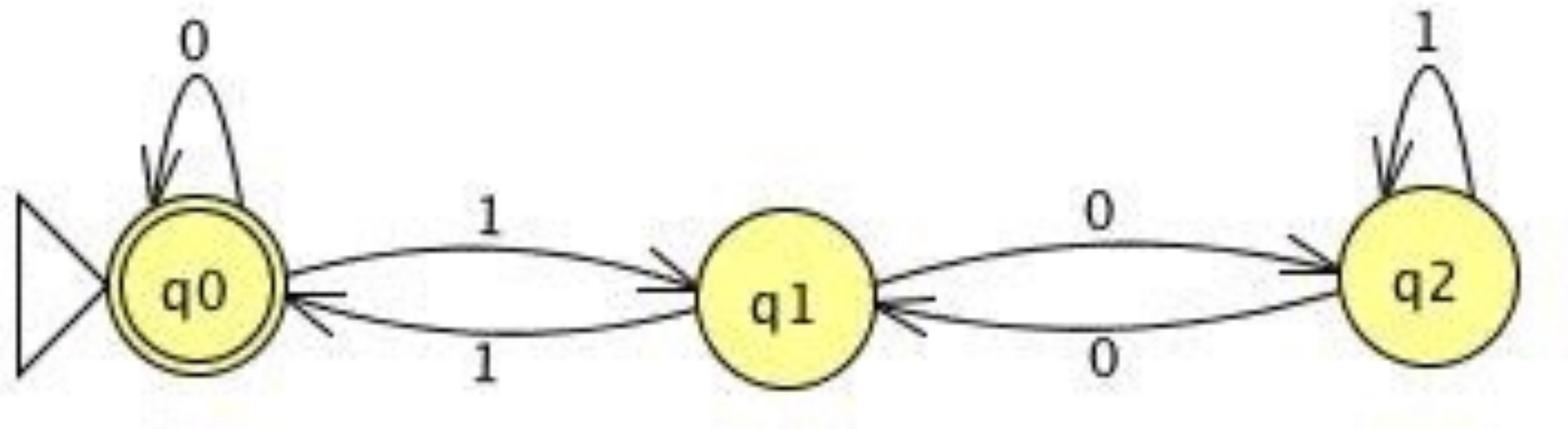
$\Sigma = \{a,b\}$

$L = \{c_1c_2\ldots c_n a c_{n+1}c_{n+2} \mid c_i \text{ in } \Sigma \text{ , per } 1 \leq i \leq n+2\}$ è l'insieme di tutte le parole che hanno una *a* nella terz'ultima posizione.



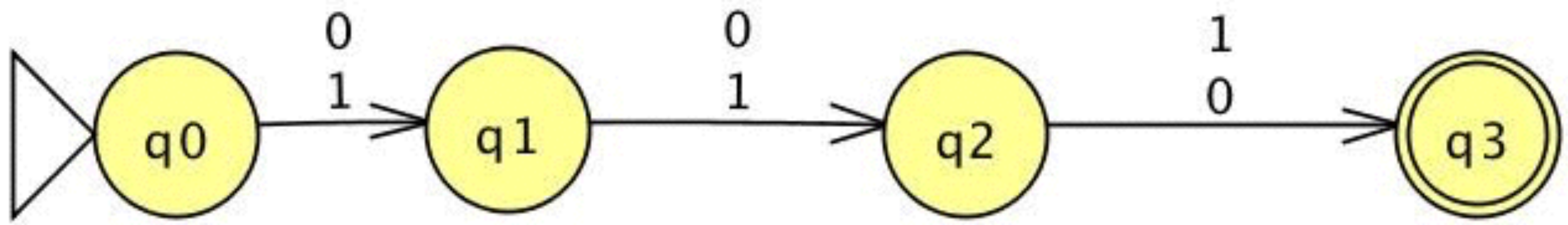
ESEMPIO 2

L= {x | x in {0,1}* e x rappresenta in binario un intero multiplo di 3}



ESEMPIO 3

L= { x | x in {0,1}* e |x| = 3} è l'insieme (finito) delle parole binarie di lunghezza 3



Operazioni sui linguaggi

Ricordiamo le operazioni insiemistiche:

$L, L' \subseteq \Sigma^*$

- **unione: $L \cup L'$**
- **intersezione: $L \cap L'$**
- **complemento: $\neg L = \Sigma^* - L$**

- **Ricordiamo che queste operazioni non sono indipendenti:**
- **$\neg(X \cup X') = \neg X \cap \neg X'$**
- **$\neg(X \cap X') = \neg X \cup \neg X'$**

Concatenazione

Concatenazione di due parole:

date x = a1a2...an e y = b1b2...bm in Σ^*, n,m ≥ 0

xy = a1a2...anb1b2...bm è la loro concatenazione

Definizione ricorsiva di concatenazione:

per x,y in Σ^* e *a* in Σ

$x\varepsilon = x$

$x(ya) = (xy)a$

La parola vuota concatenata a sinistra o a destra lascia invariata la parola, cioè è l'elemento neutro della concatenazione come 1 lo è per la moltiplicazione tra numeri

Concatenazione di linguaggi

Estensione della concatenazione ai linguaggi:

$L, L' \subseteq \Sigma^*$

$LL' = \{w \text{ in } \Sigma^* \mid \exists\, x \text{ in } L,\ y \text{ in } L' \text{ e } w = xy\}$

Esempi:

1. Se L = {a, ab,ba}, L' = {ab,b}, allora
   LL' = {aab,ab,abab,abb,baab,bab}.

bab è già presente

2. Se L = {b,ba}, L' = {ab,b},
   allora LL' = {bab,bb,baab}.

Proprietà:

$\{\varepsilon\}L = L = L\{\varepsilon\}$

$\varnothing L = \varnothing = L\varnothing$

Elevazione a potenza di parole

Potenza di una parola
per x in Σ^* $x^n = xx...x$ (n volte)

Definizione ricorsiva di potenza:
$x^0 = \varepsilon$
$x^{n+1} = x^n x$ per $n \geq 0$

Potenza di linguaggi

Estensione della potenza ai linguaggi:

per $L \subseteq \Sigma^*$ $L^n = LL...L$ (n volte)

$L^0 = \{\varepsilon\}$

Esempi:

aba

1. **se $L = \{a,ab,ba\}$ allora**
   $L^2 = \{aa,aab,aba,abab,abba,baa,baab,baba\}$.

2. **Se $L = \{ab,b\}$ allora**
   $L^3 = \{ababab,abbab,babab,bbab,ababb,abbb, babb,bbb\}$.

Stella di Kleene

$$L \subseteq \Sigma^*, \quad L^* = \bigcup_{n \geq 0} L^n$$

Quindi $L^* = \{\varepsilon\} \cup L \cup LL \ldots$

Esempi:

1. $\{a,b\}^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, abb, baa, \ldots.\}$

2. se $L=\{a,ab,ba\}$,
allora$L^*=\{\varepsilon, a, ab, ba, aa, aab, aba, abab, abba, baa, baab, baba, \ldots\}$.

Stephen Cole Kleene (1909 - 1994)

Ha insegnato alla Wisconsin University.

La sua attività di ricerca riguarda la teoria degli algoritmi, i modelli di calcolo e le funzioni ricorsive. A lui si devono alcuni dei più importanti risultati della teoria degli automi.

Il concetto di CHIUSURA

Una classe di linguaggi C è chiusa rispetto a un' operazione binaria ^ se, dati X,Y in C, allora anche X^Y è in C.

La definizione è analoga in casi di diversa arità (numero di argomenti) dell'operazione

L(DFA) è chiusa rispetto a varie importanti operazioni: unione, intersezione, complemento, concatenazione e stella di Kleene.

PROPRIETÀ DI CHIUSURA

Utilizzando le proprietà di chiusura si possono costruire automi molto complessi la cui correttezza è relativamente facile da provare:

1. si costruiscono gli automi di partenza, di una semplicità tale che la loro correttezza possa essere verificata facilmente in modo diretto

2. si utilizzano gli algoritmi di chiusura per le opportune operazioni ottenendo gli automi voluti.

La correttezza degli automi risultanti è garantita dalla correttezza degli algoritmi di chiusura.

PRIME CHIUSURE: OPERAZIONI INSIEMISTICHE

L(DFA) è chiusa rispetto alle operazioni insiemistiche: unione, intersezione, complemento.

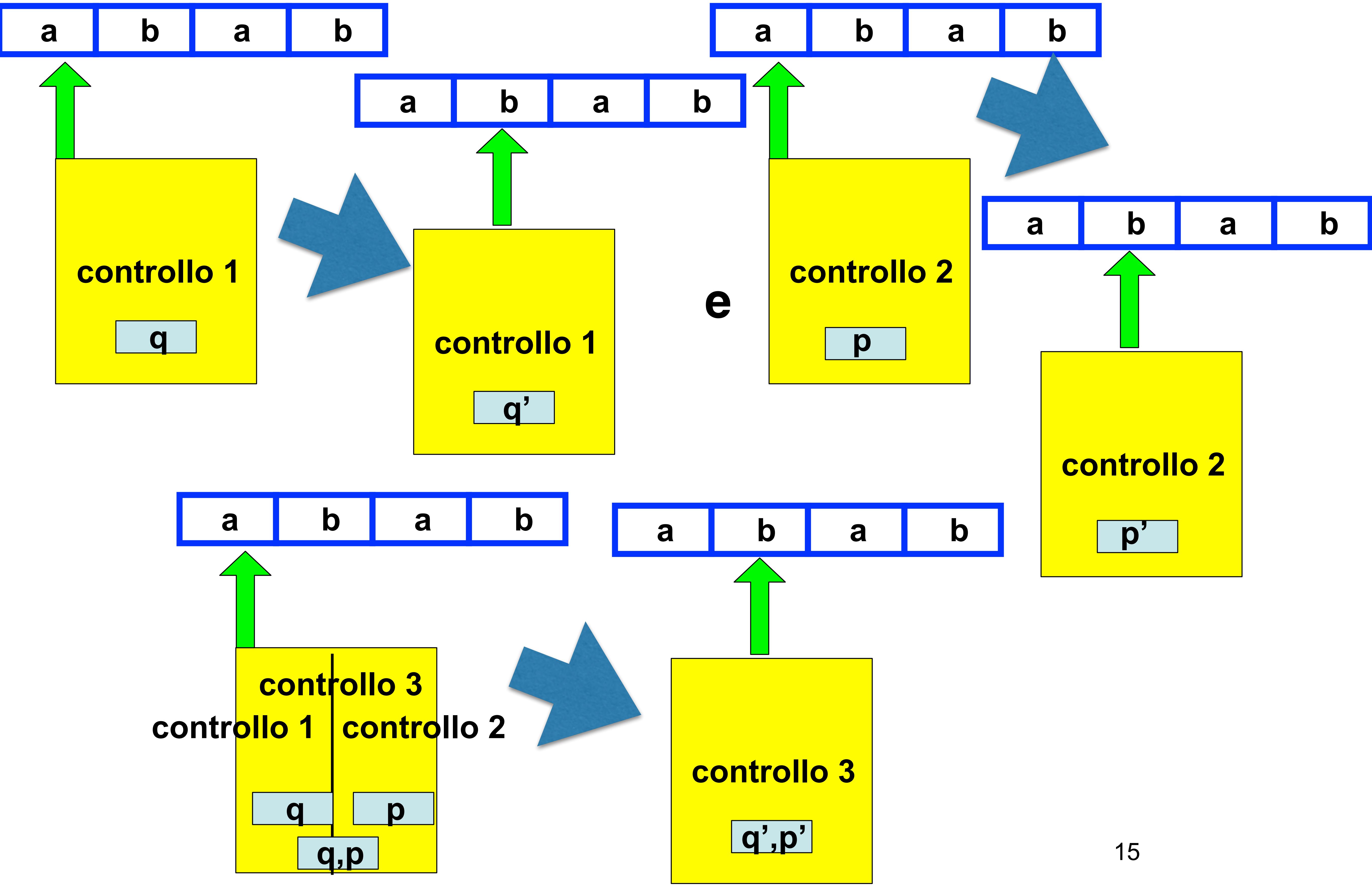
Proveremo quindi che

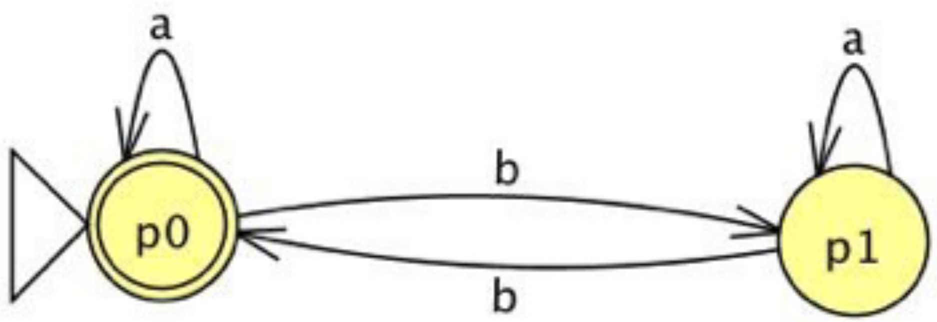
$L, L' \in \mathscr{L}(DFA) \Rightarrow L \cup L' \in \mathscr{L}(DFA)$

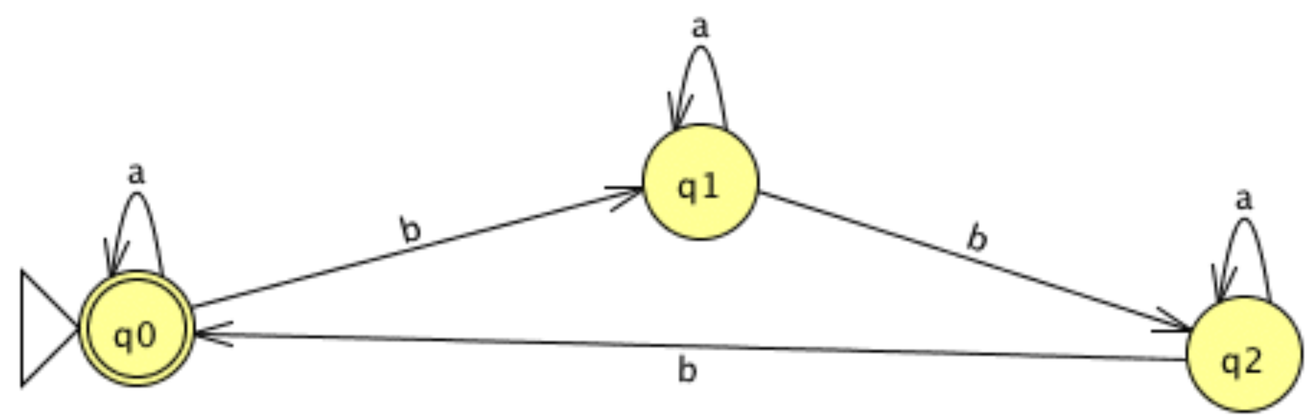
$L, L' \in \mathscr{L}(DFA) \Rightarrow L \cap L' \in \mathscr{L}(DFA)$

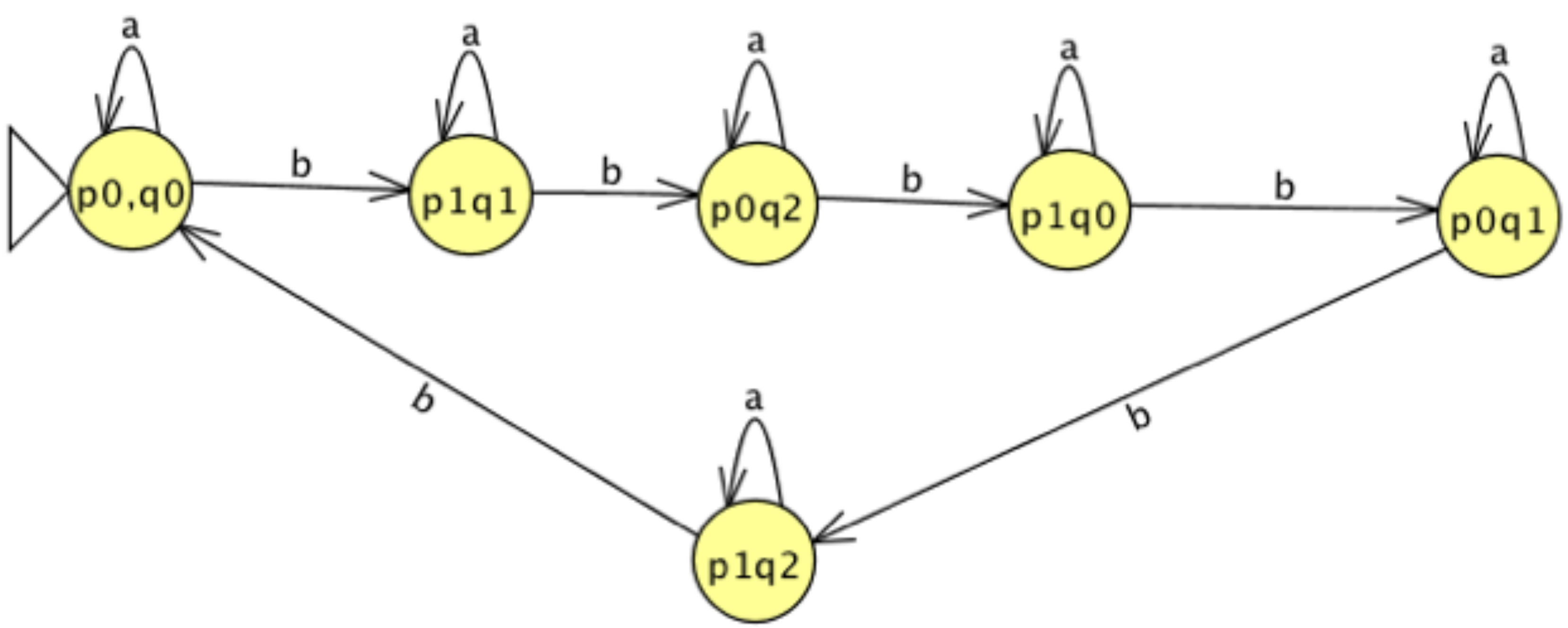
$L \in \mathscr{L}(DFA) \Rightarrow \neg L \in \mathscr{L}(DFA)$

idea della prova chiusura unione

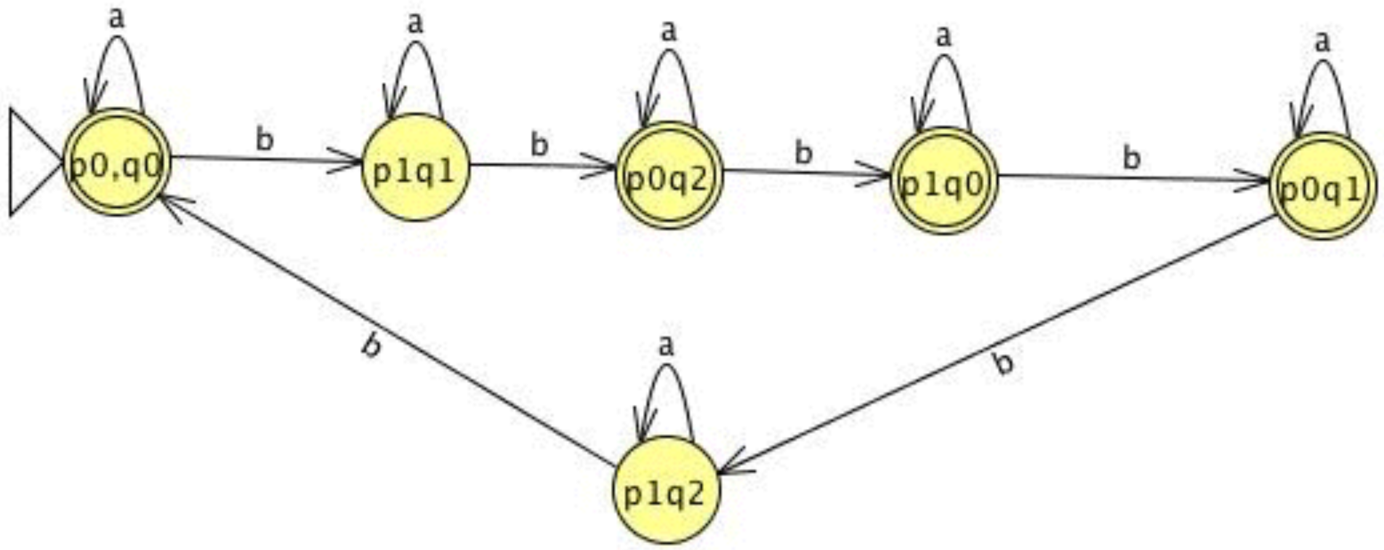


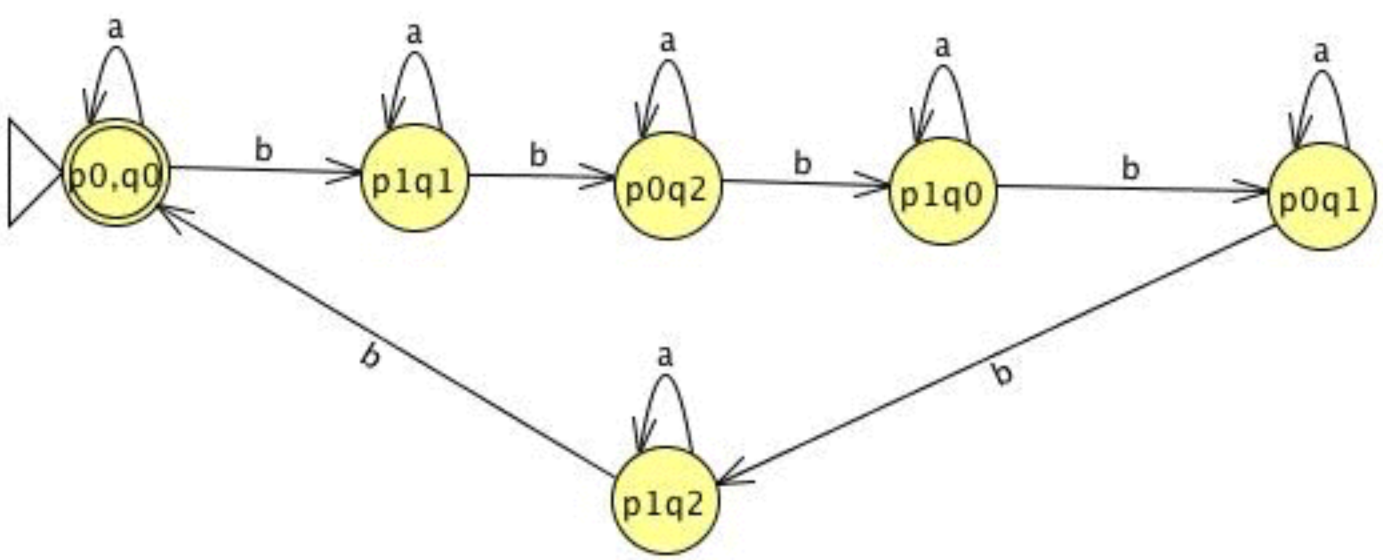

a
a
b
p0
p1
b


a
a
a
q1
b
b
q0
q2
b


a
a
a
a
a
b
b
b
b
p0,q0
p1q1
p0q2
p1q0
p0q1
b
b
a
p1q2

Unione e intersezione





Costruzione formale

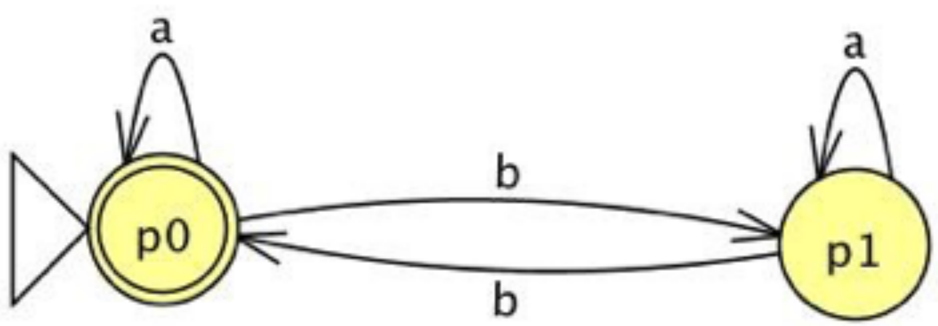
Dati due DFA $M_1 = (Q_1,\Sigma,\delta_1,q_0,F1)$ e $M_2 = (Q_2,\Sigma,\delta_2,p_0,F_2)$, è possibile costruire due DFA M e M' che accettano $L(M_1) \cup L(M_2)$ e $L(M_1) \cap L(M_2)$, rispettivamente, e cioè con $L(M) = L(M_1) \cup L(M_2)$ e $L(M') = L(M_1) \cap L(M_2)$.

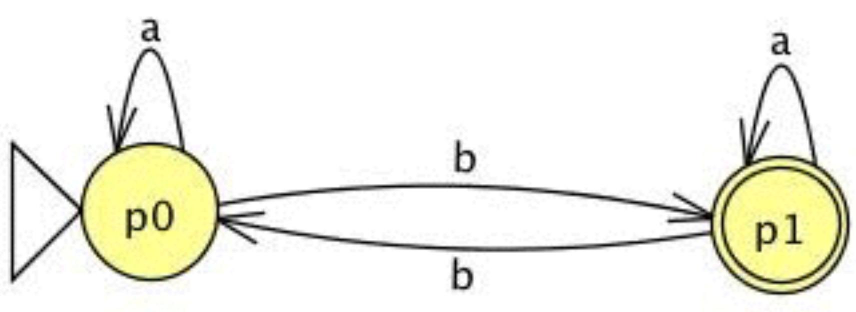
Prova. Definiamo il DFA $M = (Q_1 X Q_2,\Sigma,\delta,(q_0,p_0),F)$,
con $\delta((q,p),a) = (\delta_1(q,a),\delta_2(p,a))$.
Con questa definizione di funzione di transizione, si può dimostrare che $\delta^*((q,p),x) = (\delta_1^*(q,x),\delta_2^*(p,x))$ per ogni $x \in \Sigma^*$ per induzione sulla lunghezza dell'input.
Poi definendo $F= F_1 X Q_2 \cup Q_1 X F_2$, si accettano tutte le parole accettate da M_1 o da M_2.

Il DFA $M' = (Q_1 X Q_2,\Sigma,\delta,(q_0,p_0),F')$ è identico a M, salvo per la scelta degli stati finali, $F' = F_1 X F_2$. M' accetta tutte e sole le parole accettate da entrambi.

Qui l'ipotesi della totalità della funzione di transizione semplifica la prova!

Complemento





Dato un DFA M = (Q,Σ,δ,q_0,F) è possibile costruire un DFA M' che accetta Σ^* - L(M) = ¬ L(M).

Prova. Si consideri il DFA M' = $(Q,\Sigma,\delta,q_0,Q\text{-}F)$.
Si può osservare che
$x \notin L(M) \Leftrightarrow \delta^*(q_0,x) \notin F \Leftrightarrow \delta^*(q_0,x) \in Q - F \Leftrightarrow x \in L(M')$

Anche qui l'ipotesi della totalità della funzione di transizione semplifica la prova!

Le chiusure rispetto alle altre operazioni sui linguaggi

L(DFA) è chiusa rispetto alle operazioni di concatenazione e stella di Kleene:

Proveremo quindi che

L,L' $\in$ L(DFA) => LL' $\in$ L(DFA)

L$\in$ L(DFA) => L* $\in$ L(DFA)