

# Esercizi di Informatica Teorica

Linguaggi regolari: pumping lemma e  
proprietà di chiusura

a cura di

Luca Cabibbo e Walter Didimo

# Sommario

- pumping lemma
- proprietà di chiusura dei linguaggio regolari

notazioni sul livello degli esercizi: (\*) facile, (\*\*) non difficile  
(\*\*\*) media complessità, (\*\*\*\*) difficile, (\*\*\*\*\*) quasi impossibile

# Pumping lemma per linguaggi regolari

pumping lemma: se  $L$  è un linguaggio regolare allora  $\exists n > 0$  tale che

$\forall z \in L$  con  $|z| \geq n \exists u, v, w :$

1)  $z = uvw$

2)  $|uv| \leq n$

3)  $|v| \geq 1$

4)  $z_i = uv^i w \in L \forall i \in \mathbf{N}$  (cioè  $i = 0, 1, 2, \dots$ )

osservazioni:

1.  $n$  dipende da  $L$  (viene fissato una volta per tutte sulla base di  $L$ )
2.  $u, v, w$  dipendono da  $z$  e da  $n$  ( $u, v, w$  sono scelti in base a  $z$  e ad  $n$ )
3.  $u$  e/o  $w$  possono anche essere stringhe vuote
4. poiché può anche essere  $i = 0$ , la stringa  $z_0 = uw$  deve appartenere ad  $L$  affinché la proprietà 4 del lemma sia soddisfatta

# Pumping lemma per linguaggi regolari

considerazioni per la scelta di  $n$ : si può sempre scegliere  $n$  uguale o superiore al minimo numero di stati necessari per costruire un ASF che riconosce  $L$

utilizzo del pumping lemma: il pumping lemma rappresenta una condizione necessaria ma non sufficiente affinché un linguaggio sia regolare:

- il pumping lemma non vale per  $L \Rightarrow L$  non è regolare
- il pumping lemma vale per  $L \Rightarrow$  non si può dire niente per  $L$

quindi il pumping lemma si utilizza per provare che un linguaggio è non regolare

osservazione: il pumping lemma è ovviamente vero per linguaggi finiti; basta scegliere  $n$  uguale alla lunghezza della stringa più lunga

# Pumping lemma per linguaggi regolari

osservazione: spesso, per dimostrare che il pumping lemma non vale si può adottare una tecnica (debole) che non usa tutte le ipotesi: si può mostrare che per stringhe  $z$  (“sufficientemente” lunghe) non esiste mai una suddivisione  $z = uvw$ , con  $|v| \geq 1$  tale che

$$z_i = uv^i w \in L \quad \forall i \in \mathbf{N}$$

(non sto usando l’ipotesi  $|uv| \leq n$  e sto addirittura supponendo che la suddivisione non sia mai possibile da un certo  $n$  in poi)

osservazione: se non si riesce ad usare con successo la tecnica debole, allora si deve tentare di negare il pumping lemma usando tutte le ipotesi

# Esercizi svolti sul pumping lemma

Esercizio 1(\*\*) verificare la validità del pumping lemma per i seguenti linguaggio regolari

- $L_1 = ab^*a$
- $L_2 = a(bc)^*ba$
- $L_3 = a(bc)^*ba + babab$

## Soluzione

- $L_1 = ab^*a$

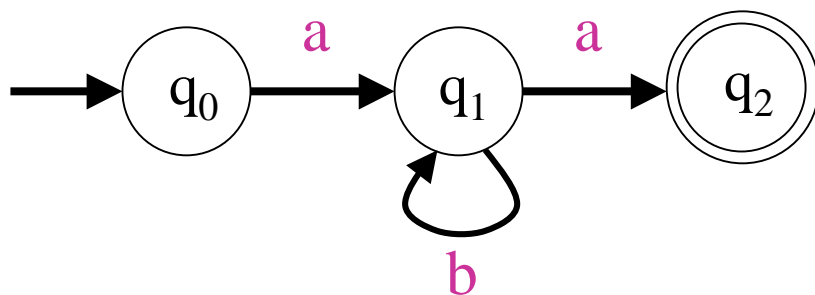
la stringa “aa” non può essere suddivisa secondo le regole del pumping lemma; ogni stringa di lunghezza  $> 2$  invece è del tipo “abbb..bba” e può sempre essere suddivisa al modo  $u = “a”$   $v = “b”$  e  $w =$  parte restante; allora basta scegliere  $n = 3$  affinché siano verificate le proprietà del pumping lemma

# Esercizi svolti sul pumping lemma

esempio:  $z = abbbba$   
                  ↑  
                  v

$z_6 = abbbbbbba$   
                  └───┬───┘  
                  v<sup>6</sup>

si osservi che un ASF con il minimo numero di stati ha tre 3 stati (escludendo lo stato “pozzo” fittizio)



# Esercizi svolti sul pumping lemma

- $L_2 = a(bc)^*ba$

la stringa “aba” non può essere suddivisa secondo le regole del pumping lemma; ogni stringa di lunghezza  $> 3$  invece è del tipo “abcbc..bcba” e può sempre essere suddivisa al modo  $u = “a”$   $v = “bc”$  e  $w =$  parte restante; allora basta scegliere  $n = 4$  (o  $n = 5$ ) affinché siano verificate le proprietà del pumping lemma (osserva che non possono esistere stringhe di lunghezza pari nel linguaggio)

esempio:  $z = abc bcba$        $z_3 = abc bc bc bcba$

esercizio: mostrare un ASF con 4 stati (escluso lo stato pozzo) che riconosce il linguaggio  $L_2$



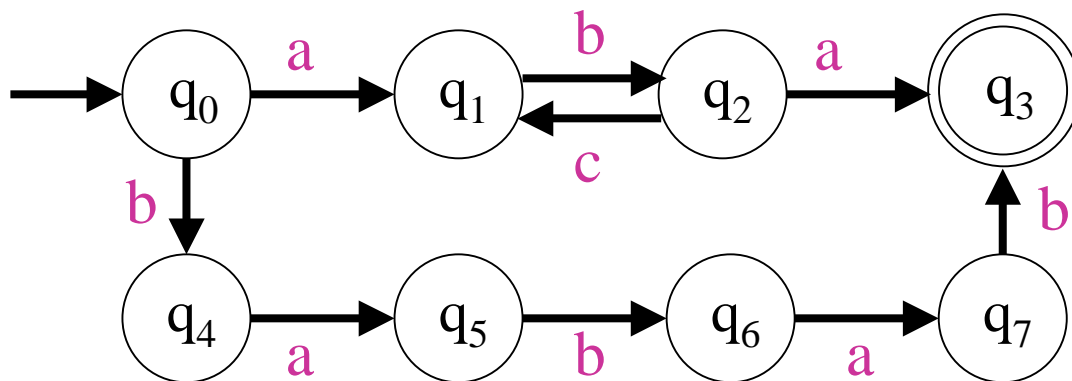
# Esercizi svolti sul pumping lemma

•  $L_3 = a(bc)^*ba + babab$

risulta  $L_3 = L_2 \cup \{babab\}$ , quindi per tutte le stringhe del linguaggio, tranne che per la stringa “babab”, si può ragionare come per il linguaggio  $L_2$ ;

tuttavia, la stringa “babab” non può essere suddivisa secondo le regole del pumping lemma, ed ha lunghezza 5;

quindi occorre scegliere  $n \geq 6$



# Esercizi svolti sul pumping lemma

Esercizio 2(\*\*\*) verificare la validità del pumping lemma per il seguente linguaggio regolare

$L = \{s \in \{a,b\}^* : s \text{ contiene un numero pari di 'a' e dispari di 'b'}\}$

## Soluzione

- tutte le stringhe di L hanno lunghezza dispari
- se z è una stringa di L, in una qualunque suddivisione  $z = uvw$ ,  $v$  deve contenere un numero pari di 'a' e un numero pari di 'b', affinché  $z_i$  appartenga ancora ad L ( $i = 0, 1, 2, \dots$ )
- non posso suddividere la stringa "aba" con le regole sopra descritte, quindi  $n$  deve essere maggiore di 3
- è sempre possibile suddividere una stringa di L di lunghezza maggiore di 3 con le regole sopra dette ed in modo tale che  $|uv| \leq 4$  e  $|v| > 1$  (dimostrare formalmente studiando tutti i casi); quindi, se scegliamo  $n = 4$ , le proprietà del pumping lemma valgono

# Esercizi svolti sul pumping lemma

Esercizio 3(\*\*) dimostrare che  $L = \{a^h b^k c^{h+k} : h, k > 0\}$  è un linguaggio non regolare

## Soluzione

è possibile utilizzare entrambe le tecniche (debole o con utilizzo di tutte le ipotesi) per negare il pumping lemma:

- **utilizzo della tecnica debole (mostro che non posso mai suddividere)**
  - sia  $uvw$  ( $|v| \geq 1$ ) una suddivisione per la stringa  $z = a^h b^k c^{h+k}$ ;
  - $v$  non può essere fatta di sole ‘a’, perché altrimenti “pommando”  $v$  si avrebbe solo una variazione del numero di ‘a’, mentre il numero di ‘b’ e di ‘c’ rimarrebbe uguale (sbilanciamento)
  - analogamente a sopra,  $v$  non può essere fatta di sole ‘b’ o di sole ‘c’ (sbilanciamento)
  - infine,  $v$  non può prendere simboli misti, perché altrimenti si avrebbero delle alternanze

# Esercizi svolti sul pumping lemma

- **utilizzo di tutte le ipotesi**

supponiamo di poter fissare un  $n$  per cui valgano le proprietà del pumping lemma; consideriamo allora una stringa  $z = a^h b^k c^{h+k}$  tale che  $h > n$ ; allora  $|z| > n$  e dovrebbe esistere una opportuna suddivisione per  $z$ ; tuttavia una qualunque suddivisione  $z = uvw$  tale che  $|uv| \leq n$  ( $|v| \geq 1$ ) implica che  $v$  è fatta di sole ‘a’; ma allora “pommando”  $v$  si avrebbe uno sbilanciamento del numero di ‘a’ rispetto al numero di ‘b’ e di ‘c’ (con  $i = 0$  le ‘a’ diminuiscono mentre per  $i > 0$  le ‘a’ aumentano).

esercizio dimostrare, utilizzando entrambe le tecniche del precedente esercizio, che il linguaggio  $L = \{(ab)^h (cd)^h : h > 0\}$  non è regolare

# Esercizi svolti sul pumping lemma

Esercizio 4(\*\*\*) dimostrare che  $L = \{ss \mid s \in \{a,b\}^*\}$  è un linguaggio non regolare

## Soluzione

- non si può utilizzare la tecnica debole: infatti esistono sempre delle stringhe  $z$  (lunghe a piacimento) che possono essere suddivise al modo  $z = uvw$  (con  $|v| \geq 1$ ) così che  $z_i$  appartenga ad  $L$  (esempio:  $z = aa\underline{aa} \dots aa = u\underline{v}w$ )
- utilizziamo tutte le ipotesi per negare il pumping lemma: supponiamo per assurdo che il linguaggio sia regolare; allora deve valere il pumping lemma, cioè deve esistere un  $n$  (opportunamente grande) tale che, per ogni  $z$  di  $L$  di lunghezza maggiore o uguale ad  $n$  è possibile scrivere  $z = uvw$  ( $|uv| \leq n$ ,  $|v| \geq 1$ ) in modo che  $z_i = uv^i w$  appartenga ad  $L$  per ogni naturale  $i$ ;

# Esercizi svolti sul pumping lemma

scegliamo allora la seguente stringa di  $L$ :  $z = a^k b a^k b$ , con  $k > n$ ;  
poiché  $|z| > n$ , cerco una suddivisione  $uvw$  “valida” per  $z$ ; questa suddivisione deve essere tale che  $|uv| \leq n$ , ed allora necessariamente sarà  $v = a^h$  con  $h < k$ ;

ma allora sarà  $z_i = s_1 s_2$ , dove  $s_1$  ha un numero di ‘a’ iniziali superiore a quello di  $s_2$  se  $i > 0$  ed inferiore se  $i = 0$ , e questo è assurdo per l’ipotesi fatta; il pumping lemma è dunque non valido, e perciò il linguaggio è non regolare

esercizio: dimostrare che il linguaggio delle stringhe palindrome di lunghezza pari su  $\{a,b\}$  non è regolare

# Esercizi svolti sul pumping lemma

Esercizio 5(\*\*\*) sia  $L = \{a^k : k \text{ è un numero primo}\}$ ; dimostrare che  $L$  è un linguaggio non regolare

## Soluzione

usiamo la tecnica debole: dimostriamo cioè che, se  $z = aaaa\dots aaa$  ha un numero primo di 'a', allora non è mai possibile suddividere  $z$  al modo  $z = uvw$ , così che  $|z_i| = |uv^i w|$  sia un numero primo, per ogni naturale  $i$ :

- sia dunque  $|z| = |a^k|$  un numero primo ( $|z| \geq 2$ , perché 1 non è primo)
- consideriamo una qualunque suddivisione  $z = uvw$ , con  $|v| \geq 1$
- risulta  $|z_i| = |u| + i|v| + |w|$
- per ogni  $i > 0$  si può riscrivere  $|z_i| = |z| + (i - 1)|v|$
- ma allora, per  $i = |z| + 1$  risulta  $|z_i| = |z| + |z| = |z|(1 + 1)$ , e quindi  $|z_i|$  non è numero primo (perché prodotto di due numeri maggiori di 1)

# Esercizi svolti sul pumping lemma

Esercizio 6(\*\*\*) sia  $L = \{a^{2^h} : h \geq 0\}$ ; dimostrare che  $L$  è un linguaggio non regolare

## Soluzione

usiamo la tecnica debole: dimostriamo che se  $|z| = |aaa...aa|$  è una potenza di 2, allora non è mai possibile suddividere  $z$  al modo  $z = uvw$  così che  $|z_i| = |uv^i w|$  sia una potenza di 2, per ogni naturale  $i$ :

- sia  $|z| = |a^{2^h}| = 2^h$  con  $h \geq 0$
- consideriamo una qualunque suddivisione  $z = uvw$ , con  $|v| \geq 1$
- risulta  $|z_i| = |u| + i|v| + |w|$
- per  $i > 0$  si può riscrivere  $|z_i| = |z| + (i - 1)|v| = 2^h + (i - 1)|v|$



# Esercizi svolti sul pumping lemma

- sono possibili due casi per  $|v|$ :
  - $|v|$  è un numero dispari; in questo caso per  $i = 2$  risulta  $|z_i| = 2^h + (i - 1)|v| = 2^h + |v|$ , che è un numero dispari maggiore di 2 e quindi non può essere una potenza di 2
  - $|v|$  è un numero pari; in questo caso per  $i = (2^h + 1)$  risulta  $|z_i| = 2^h + (i - 1)|v| = 2^h + 2^h|v| = 2^h(1 + |v|)$  che ancora una volta non può essere una potenza di due perché  $(1 + |v|)$  è dispari (e quindi contiene almeno un fattore diverso da 2).

esercizio: dimostrare che il linguaggio delle stringhe su  $\{a,b,c\}$ , tali che il numero di 'a' al quadrato più il numero di 'b' al quadrato è uguale al numero di 'c' al quadrato (cioè  $(\#a)^2 + (\#b)^2 = (\#c)^2$ ) è non regolare

# Esercizi da svolgere sul pumping lemma

Esercizio 7(\*\*\*) provare la validità del pumping lemma per i seguenti linguaggi regolari, stabilendo qual'è il minimo  $n$  utilizzabile per la prova

- $L_1 = aa(bb)^*$
- $L_2 = abc + accb + a(cc)^*ba$
- $L_3 =$  insieme delle stringhe in  $\{a,b\}^+$  con un numero pari di 'a'

Esercizio 8(\*\*\*) dimostrare, utilizzando il pumping lemma, che i seguenti linguaggi non sono regolari:

- $L_1 = \{a^k b a^k : k > 0\}$
- $L_2 = \{a^h b^k : k > h > 0\}$
- $L_3 = \{a^k b^h : k > h > 0\}$
- $L_4 = \{s \in \{a,b\}^* : \text{il numero di 'a' è maggiore del numero di 'b'}\}$

# Proprietà di chiusura dei linguaggi regolari

**teorema:** se  $L_1$  ed  $L_2$  sono due linguaggi regolari  $\Rightarrow$  anche i seguenti linguaggi sono regolari

- $L = L_1 \cup L_2$  (unione)
- $L = L_1 \bullet L_2$  (concatenazione)
- $L = L_1^*$  (chiusura stella)
- $L = \Sigma_1 - L_1$  (complementazione)
- $L = L_1 \cap L_2$  (intersezione)
- $L = L_1 - L_2$  (differenza)

# Automa che riconosce il linguaggio unione

$A_1 = \langle \Sigma_1, K_1, F_1, \delta_{N1}, q_{01} \rangle$  ASFND che riconosce  $L_1$

$A_2 = \langle \Sigma_2, K_2, F_2, \delta_{N2}, q_{02} \rangle$  ASFND che riconosce  $L_2$

$A = \langle \Sigma, K, F, \delta_N, q_0 \rangle$  ASFND che riconosce  $L = L_1 \cup L_2$

$$\Sigma = \Sigma_1 \cup \Sigma_2$$

$$K = K_1 \cup K_2 \cup \{q_0\}$$

$$F = \begin{cases} F_1 \cup F_2 & \text{se } q_{01} \notin F_1 \text{ e } q_{02} \notin F_2 \\ F_1 \cup F_2 \cup \{q_0\} & \text{se } q_{01} \in F_1 \text{ o } q_{02} \in F_2 \end{cases}$$

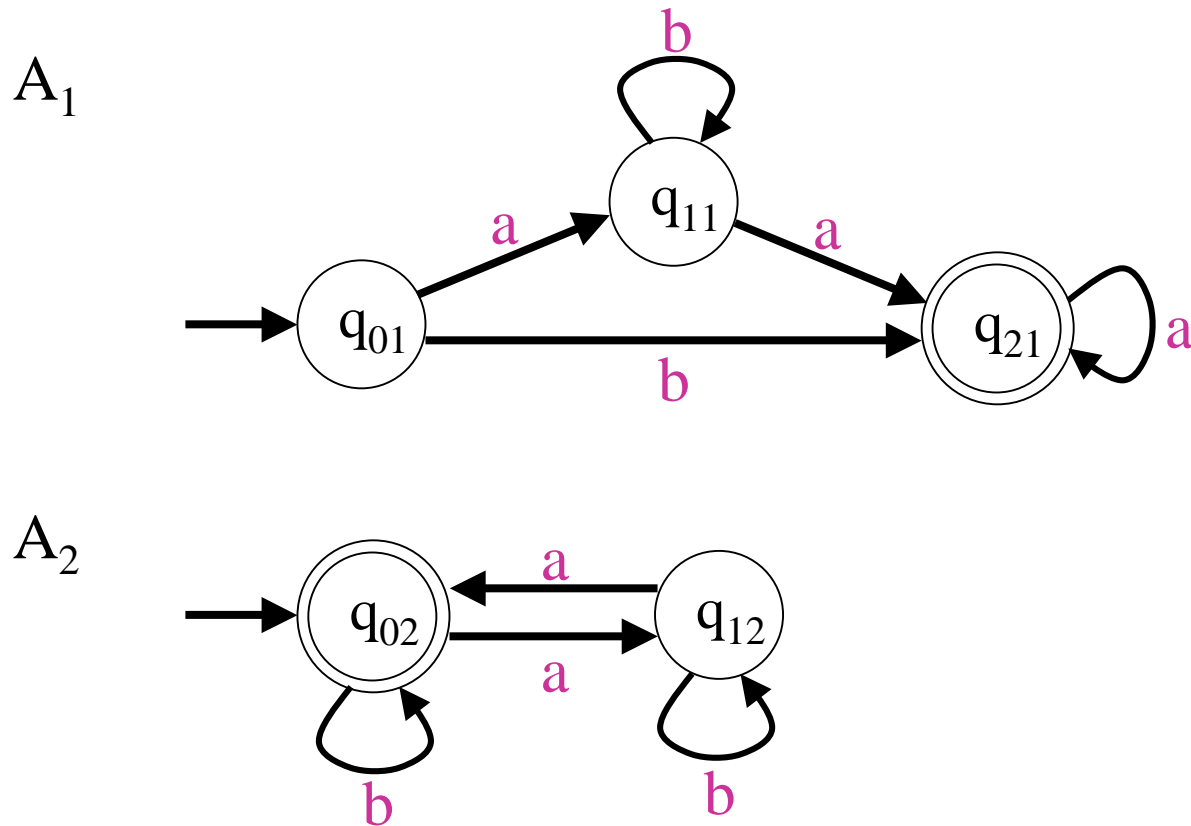
$$\delta_N(q, a) = \delta_{N1}(q, a) \quad \forall q \in K_1 \quad \forall a \in \Sigma_1$$

$$\delta_N(q, a) = \delta_{N2}(q, a) \quad \forall q \in K_2 \quad \forall a \in \Sigma_2$$

$$\delta_N(q_0, a) = \delta_{N1}(q_{01}, a) \cup \delta_{N2}(q_{02}, a) \quad \forall a \in \Sigma$$

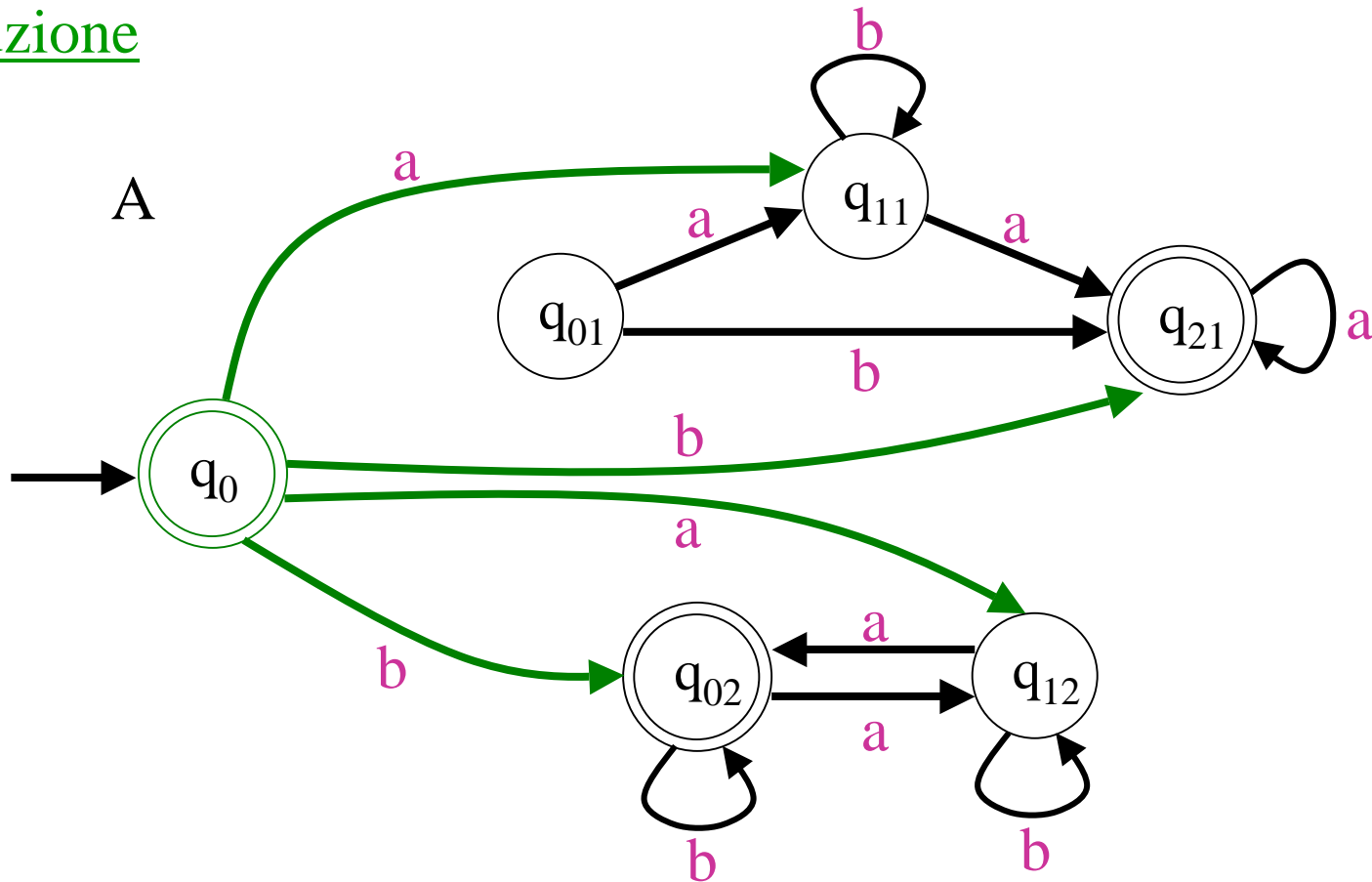
# Esercizi sull'unione di automi

Esercizio 9(\*\*) determinare l'automa  $A = A_1 \cup A_2$  e dire quale linguaggio riconosce, scrivendolo sotto forma di espressione regolare



# Esercizi sull'unione di automi

## Soluzione



$$L = (ab^*a + b)a^* + b^*(ab^*ab^*)^*$$

# Automa che riconosce il linguaggio concatenazione

$A_1 = \langle \Sigma_1, K_1, F_1, \delta_{N1}, q_{01} \rangle$  ASFND che riconosce  $L_1$

$A_2 = \langle \Sigma_2, K_2, F_2, \delta_{N2}, q_{02} \rangle$  ASFND che riconosce  $L_2$

$A = \langle \Sigma, K, F, \delta_N, q_0 \rangle$  ASFND che riconosce  $L = L_1 \cdot L_2$

$$\Sigma = \Sigma_1 \cup \Sigma_2$$

$$K = K_1 \cup K_2$$

$$F = \begin{cases} F_2 & \text{se } \varepsilon \notin L_2 \\ F_1 \cup F_2 & \text{se } \varepsilon \in L_2 \end{cases}$$

$$q_0 = q_{01}$$

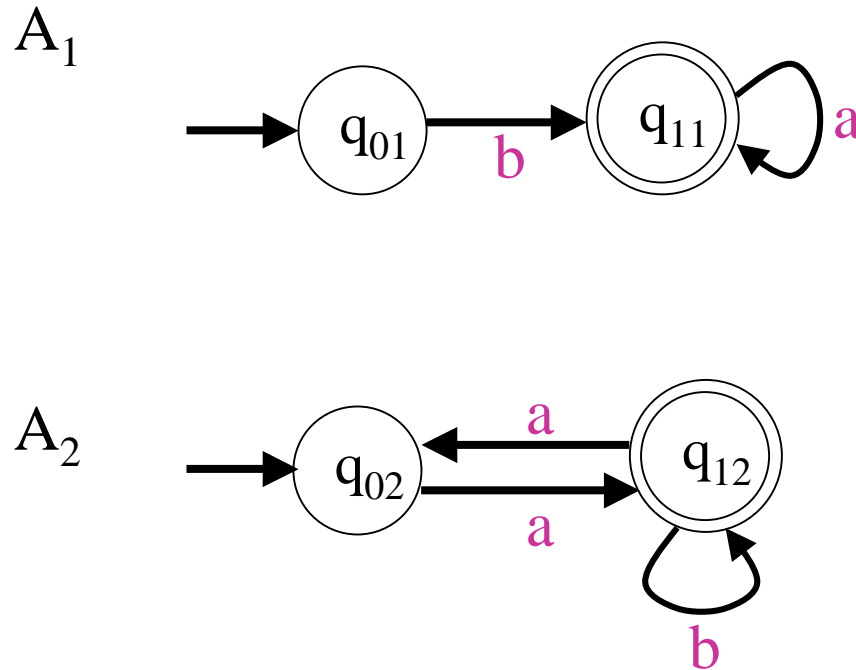
$$\delta_N(q, a) = \delta_{N1}(q, a) \quad \forall q \in K_1 - F_1 \quad \forall a \in \Sigma_1$$

$$\delta_N(q, a) = \delta_{N1}(q, a) \cup \delta_{N2}(q_{02}, a) \quad \forall q \in F_1 \quad \forall a \in \Sigma$$

$$\delta_N(q, a) = \delta_{N2}(q, a) \quad \forall q \in K_2 \quad \forall a \in \Sigma_2$$

# Esercizi sulla concatenazione di automi

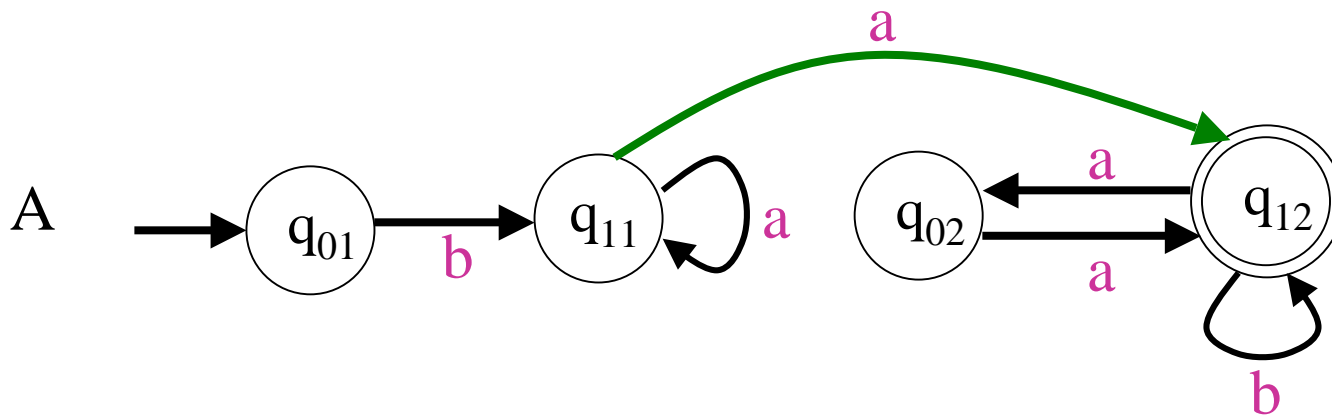
Esercizio 10(\*\*) determinare l'automa  $A = A_1 \cdot A_2$  e dire quale linguaggio riconosce, scrivendolo sotto forma di espressione regolare





# Esercizi sulla concatenazione di automi

Esercizio 10(\*\*) determinare l'automa  $A = A_1 \cdot A_2$  e dire quale linguaggio riconosce, scrivendolo sotto forma di espressione regolare



$$L = ba^*a(b^*(aa)^*)^*$$

# Automa che riconosce il linguaggio complementare

$A = \langle \Sigma, K, F, \delta, q_0 \rangle$  ASF che riconosce  $L$

$A' = \langle \Sigma', K', F', \delta', q'_0 \rangle$  ASF che riconosce  $L' = \Sigma^* - L$

$$\Sigma' = \Sigma$$

$K' = K \cup \{d\}$  ( 'd' serve solo se c'è qualche  $\delta(q,a)$  indefinito)

$$F' = K - F$$

$$q'_0 = q_{01}$$

$\delta'(q,a) = \delta(q,a) \quad \forall q \in K \text{ e } \forall a \in \Sigma : \delta(q,a) \text{ è definito}$

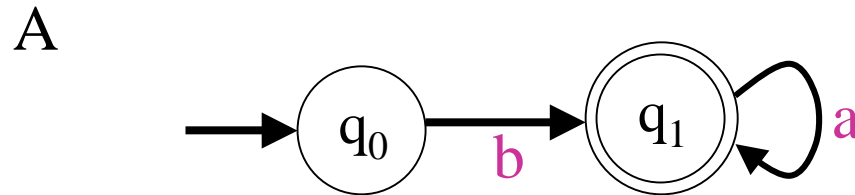
$\delta'(q,a) = d \quad \forall q \in K \text{ e } \forall a \in \Sigma : \delta(q,a) \text{ è indefinito}$

$\delta'(d,a) = d \quad \forall a \in \Sigma$

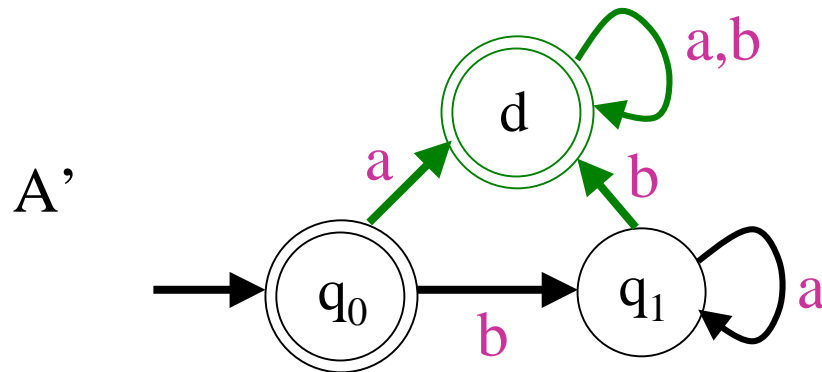
nota: si ricordi che dire che  $\delta(q,a)$  è indefinito è come dire che esiste uno stato (pozzo) non finale  $q'$  tale che  $\delta(q,a) = q' = \delta(q',x) \quad \forall x \in \Sigma$

# Esercizi sulla complementazione di automi

Esercizio 11(\*\*) determinare l'automa  $A'$  complementare di  $A$  e dire quale linguaggio riconosce, scrivendolo sotto forma di espressione regolare



Soluzione



$$L = (a+ba^*b)(a+b)^* + \epsilon$$

# Automa che riconosce il linguaggio chiusura stella

$A = \langle \Sigma, K, F, \delta_N, q_0 \rangle$       ASFND che riconosce  $L$

$A' = \langle \Sigma', K', F', \delta'_N, q'_0 \rangle$       ASFND che riconosce  $L^*$

$$\Sigma' = \Sigma$$

$$K' = K \cup \{q'_0\}$$

$$F' = F \cup \{q'_0\}$$

$$\delta'_N(q, a) = \delta_N(q, a) \quad \forall q \in K - F \text{ e } \forall a \in \Sigma$$

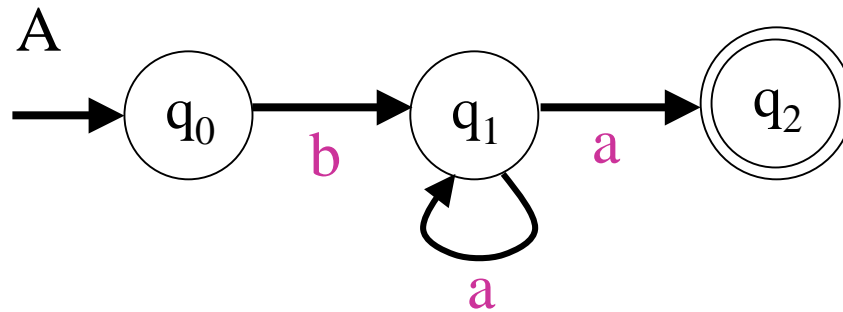
$$\delta'_N(q, a) = \delta_N(q, a) \cup \delta_N(q_0, a) \quad \forall q \in F \text{ e } \forall a \in \Sigma$$

$$\delta'_N(q'_0, a) = \delta_N(q_0, a) \quad \forall a \in \Sigma$$

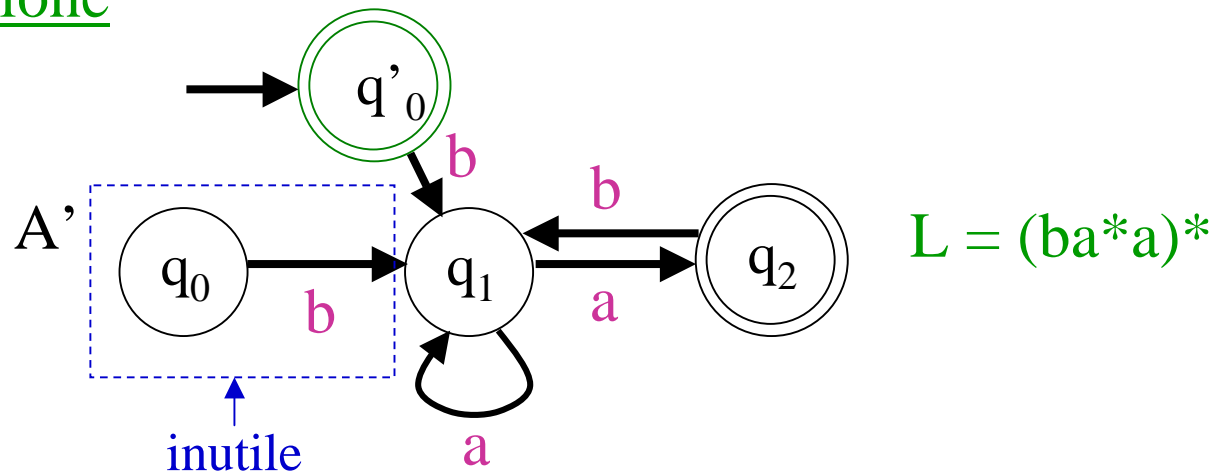
nota: lo stato  $q'_0$  è uno stato finale perché  $L^*$  contiene sempre la stringa vuota

# Esercizi sulla chiusura stella di automi

Esercizio 12(\*\*) determinare l'automa  $A'$  chiusura stella di  $A$  e dire quale linguaggio riconosce, scrivendolo sotto forma di espressione regolare



Soluzione



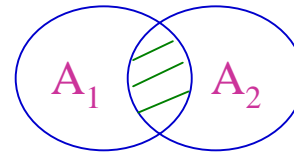
# Automati che riconoscono intersezione e differenza

$A_1 = \langle \Sigma_1, K_1, F_1, \delta_1, q_{01} \rangle$  ASF che riconosce  $L_1$

$A_2 = \langle \Sigma_2, K_2, F_2, \delta_2, q_{02} \rangle$  ASF che riconosce  $L_2$

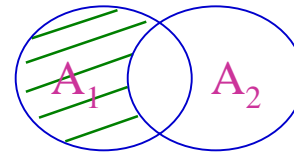
- ASFND che riconosce  $L = L_1 \cap L_2$  (intersezione)

$$A = A_1 \cap A_2 = c ( c(A_1) \cup c(A_2) )$$



- ASFND che riconosce  $L = L_1 - L_2$  (differenza)

$$A = A_1 - A_2 = c ( c(A_1) \cup A_2 )$$



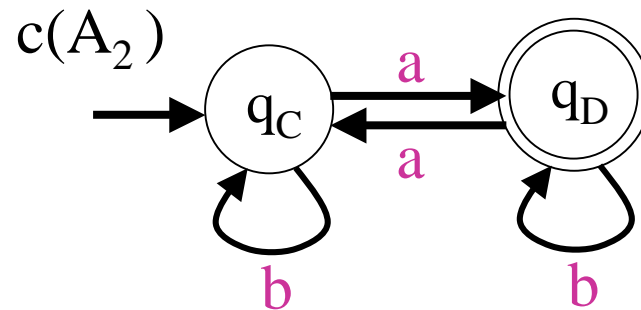
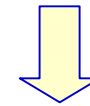
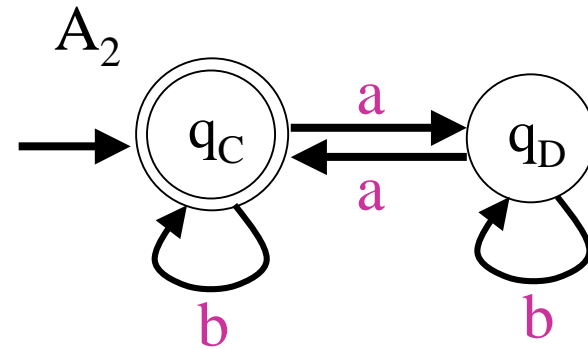
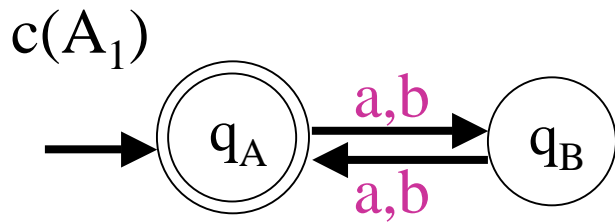
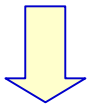
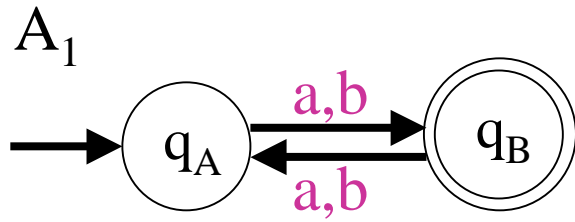
# Esercizi svolti sulle proprietà di chiusura

Esercizio 13(\*\*\*) dimostrare che il linguaggio  $L \subseteq \{a,b\}^*$  delle stringhe di lunghezza dispari e con un numero pari di 'a' è regolare; costruire poi un ASF che riconosce L

## Soluzione

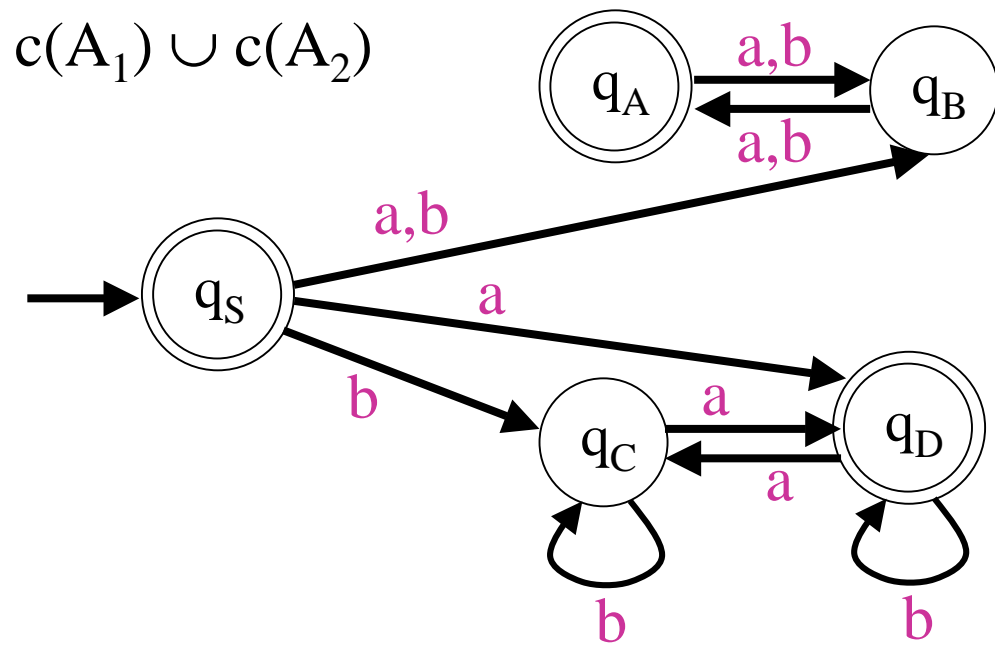
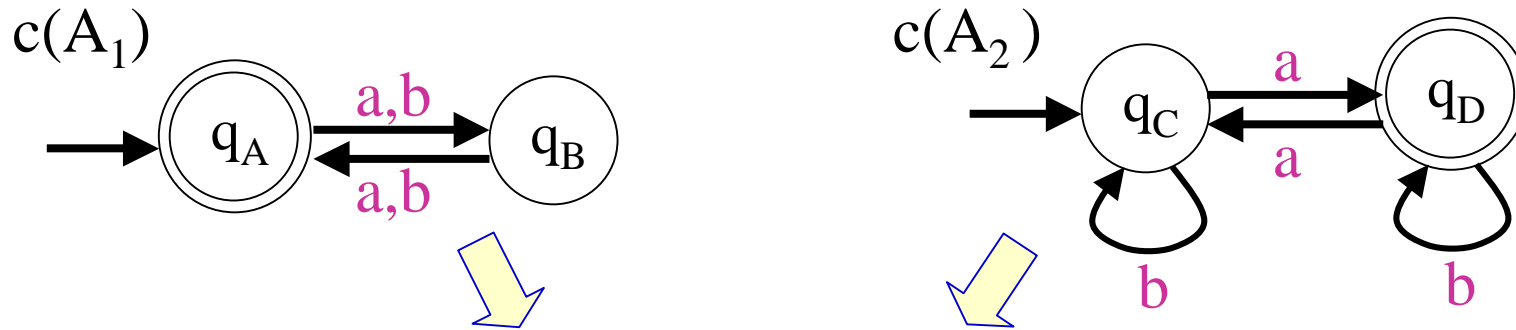
- il linguaggio  $L_1$  delle stringhe su  $\{a,b\}$  di lunghezza dispari è regolare, infatti:  $L_1 = (a+b)((a+b)(a+b))^*$
- il linguaggio  $L_2$  delle stringhe su  $\{a,b\}$  con un numero pari di 'a' è regolare, infatti:  $L_2 = b^*(ab^*ab^*)^*$
- il linguaggio L è l'intersezione di  $L_1$  ed  $L_2$ , cioè:  $L = L_1 \cap L_2$ , quindi è regolare per le proprietà di chiusura dei linguaggio regolari

# Esercizi svolti sulle proprietà di chiusura



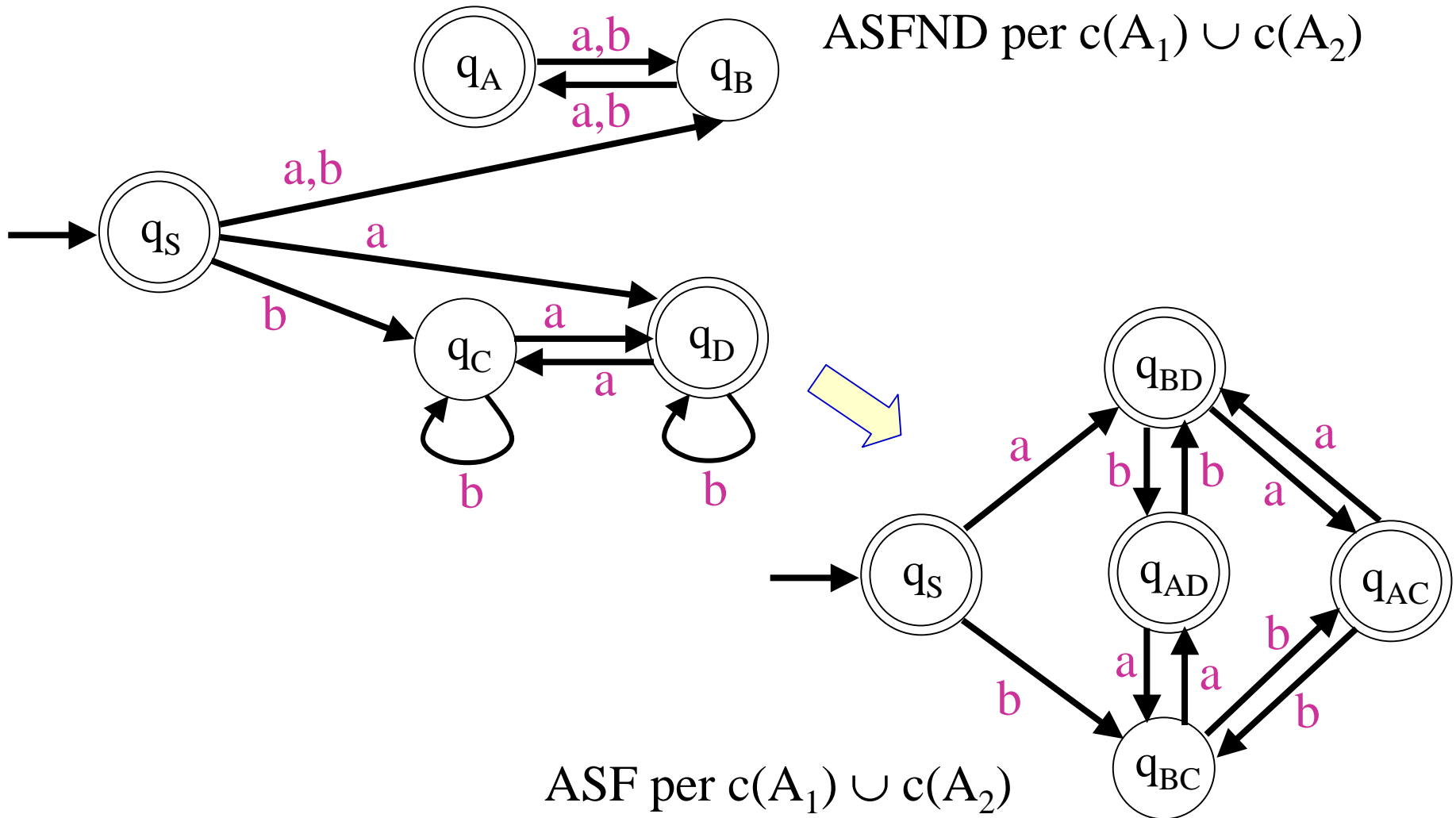


# Esercizi svolti sulle proprietà di chiusura

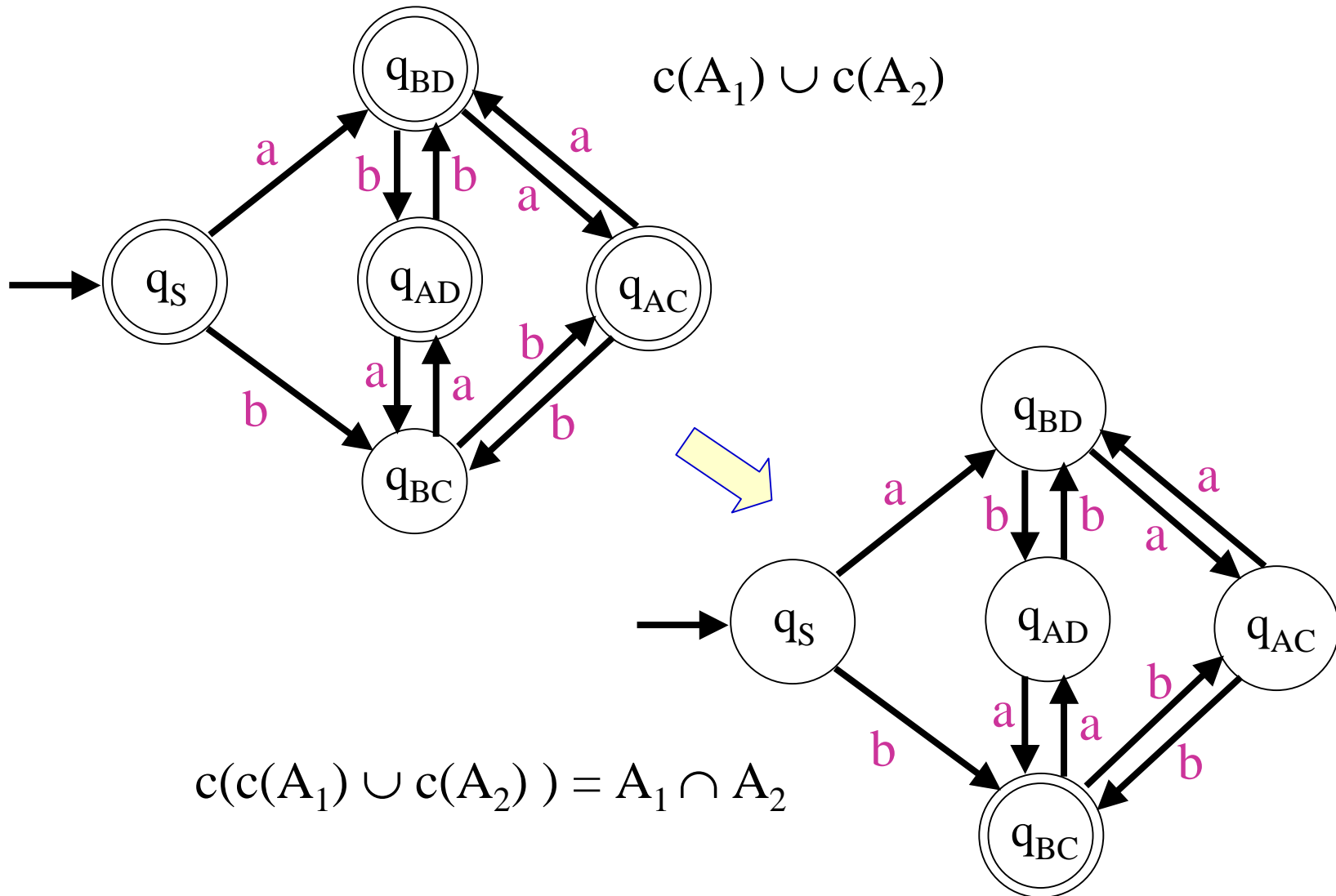


per poter costruire il complementare, occorre prima ricavare l'ASF dall'ASFND

# Esercizi svolti sulle proprietà di chiusura



# Esercizi svolti sulle proprietà di chiusura



# Esercizi svolti sulle proprietà di chiusura

Esercizio 14(\*\*\*) dimostrare, utilizzando le proprietà di chiusura dei linguaggi regolari, che il linguaggio  $L$  delle stringhe non vuote su  $\{a,b\}$  contenenti lo stesso numero di 'a' e di 'b' non è regolare

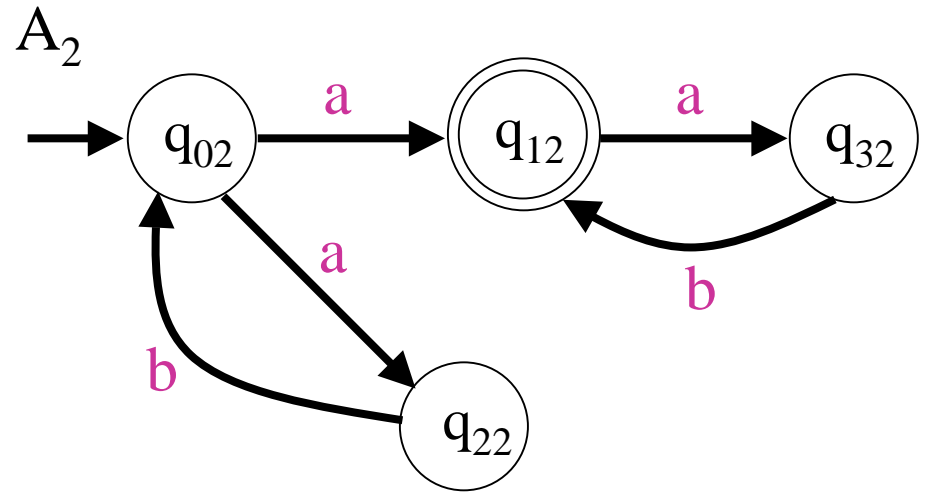
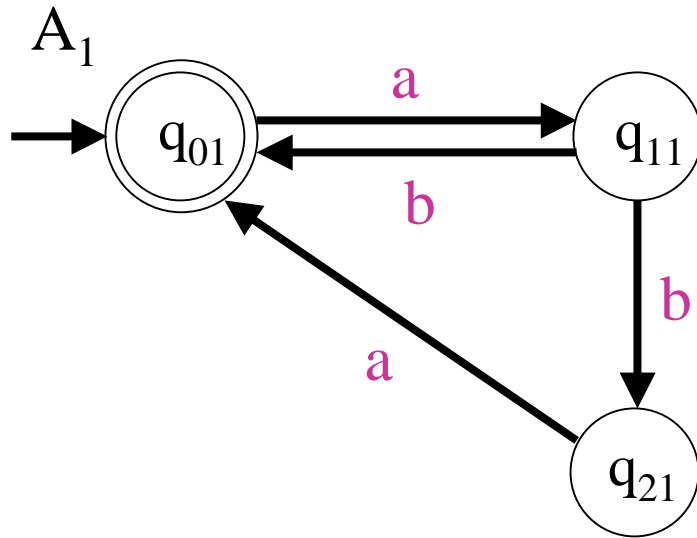
## Soluzione

- supponiamo per assurdo che  $L$  sia regolare
- il linguaggio  $L' = \{a^n b^m : n, m > 0\}$  è regolare, infatti  $L' = aa^*bb^*$
- allora, per le proprietà di chiusura dei linguaggi regolari, dovrebbe essere che  $L \cap L'$  è un linguaggio regolare; tuttavia risulta  $L \cap L' = \{a^n b^n : n > 0\}$ , il quale sappiamo essere un linguaggio non regolare
- da ciò l'assurdo

esercizio: dimostrare che  $L$  non è regolare usando il pumping lemma; si riesce ad applicare la tecnica debole per negare il pumping lemma?

# Esercizi da svolgere sulle proprietà di chiusura

Esercizio 15(\*\*\*) dati i seguenti ASFND



costruire gli ASFND unione e differenza di  $A_1$  e  $A_2$

Esercizio 16(\*\*\*) si dimostri, usando le proprietà di chiusura dei linguaggi regolari, che il linguaggio  $L$  delle stringhe su  $\{a,b,c\}$  con un numero pari di 'a' più 'b' è regolare; costruire inoltre un ASFND che riconosce  $L$