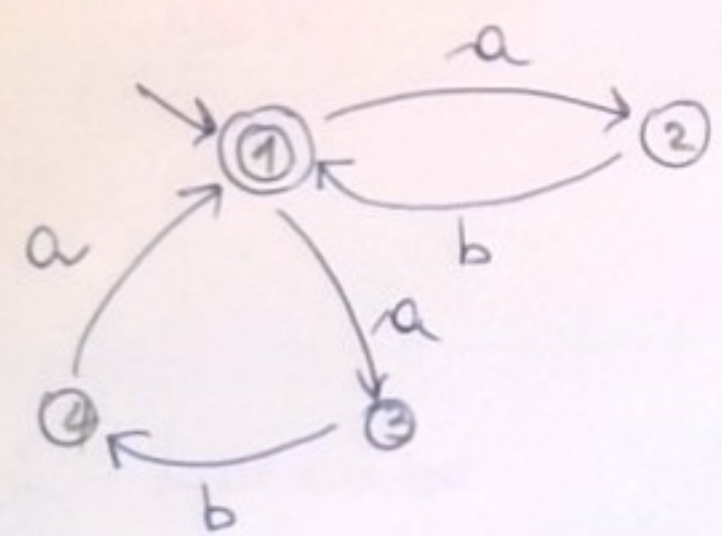
API - 9 - 3 - 2021

## - ASF$\underline{\underline{D}}$ -

**Es** Sia $\mathcal{L} = (\boxed{ab} + \boxed{aba})^*$, trovare un ASFD $\mathcal{A}$ tc $\mathcal{L}(\mathcal{A}) = \mathcal{L}$.

$\{\varepsilon, ab, aba, ababa, abaab, abababaabaab, \dots\}$

Primo tentativo:

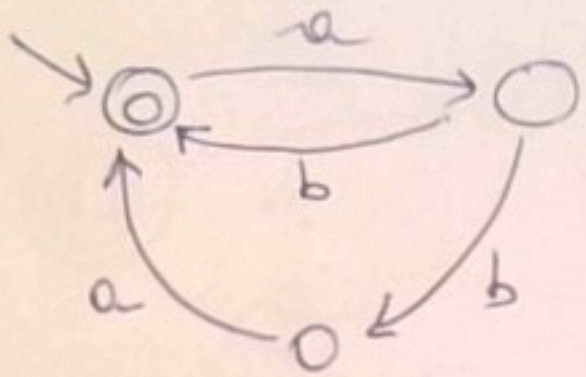


problema: non è deterministico:

$\tau(1, a) = ?$ → $\tau$ non è una funzione

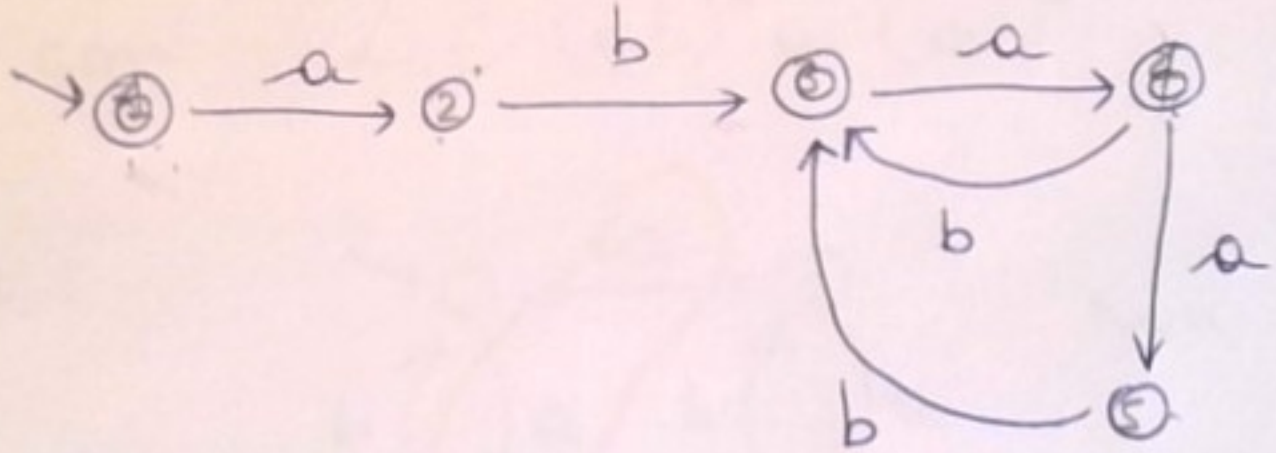
$\tau(1, a) = \{2, 3\}$

Idea: $L = (ab + aba)^* = (a(b + ba))^*$



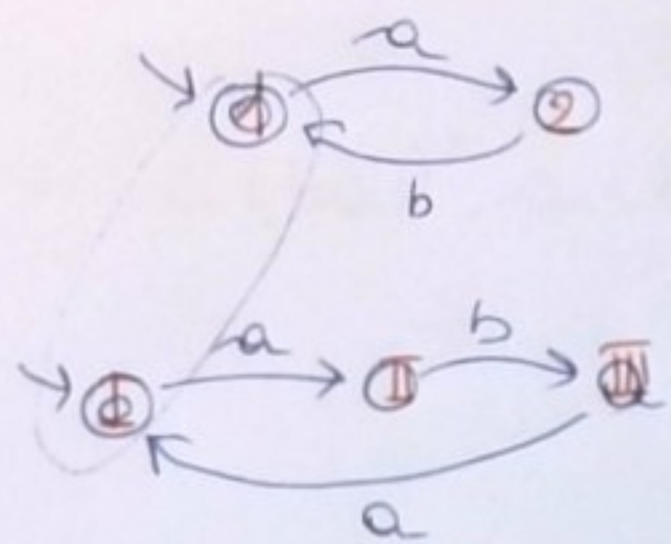

non ho risolto, ma sono sulla buona strada

Ora: $L = (ab + aba)^* = (a(b + ba))^* = (ab(\varepsilon + a))^*$




---

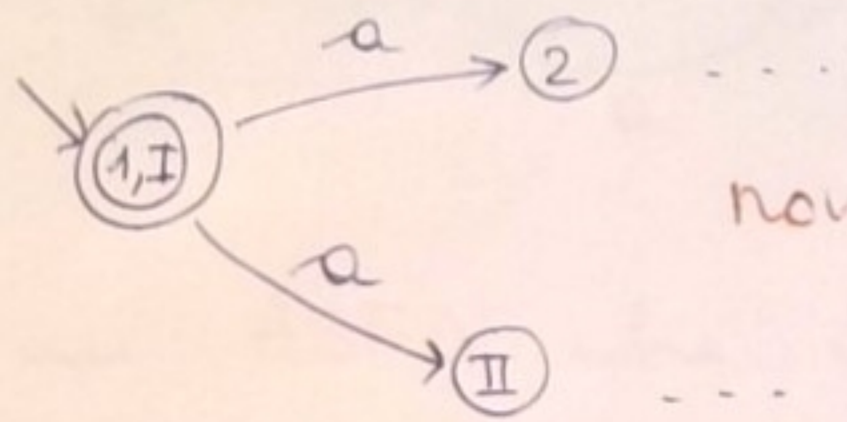
Idea di Lorenzo




$(ab)^*$

$(aba)^*$

Ora "unisco" gli automi
(fondo)




non viene deterministico

... e poi riconoscerebbe

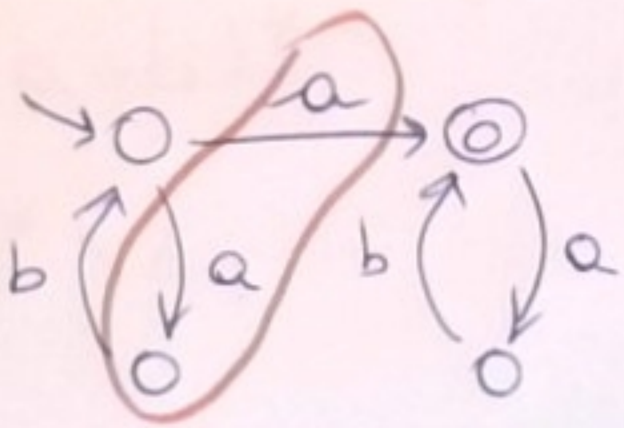
$(ab)^* + (aba)^*$ oppure $L$ ?

<u>Es</u> Idem con $L = (ab)^* a (ab)^*$

abab... ab a abab... ab

abab ab ... aba

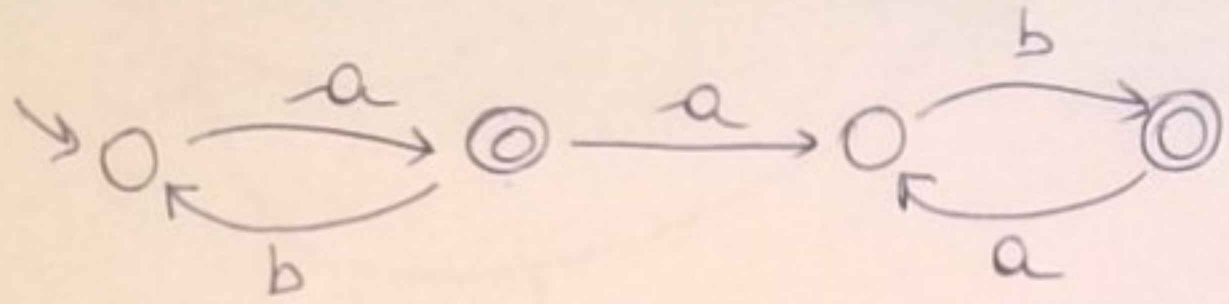
Tentativo:




non è deterministico

D'altra parte $L = a + a(ba)^* a(ba)^* b + a(ba)^+$

Allora un ASFD che riconosce $L$ è

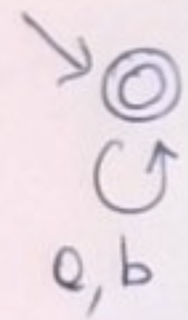



**Es** Costruire un ASFD che accetti

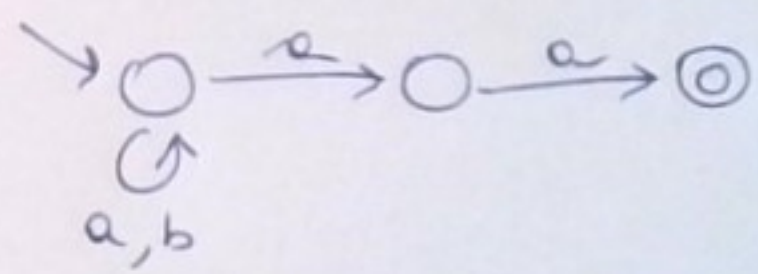
1. $L_0 = b^*(ab^*)^* = (a+b)^*$
2. $L_1 = (a+b)^* aa$
3. $L_2 = b^* a b^* a b^* a b^*$
4. $L_3 = b(a+b)^*$
5. $L_4 = \Sigma^* \setminus L_3$

sull'alfabeto $\Sigma = \{a, b\}$.

1. $b^*(ab^*)^* = \Sigma^*$

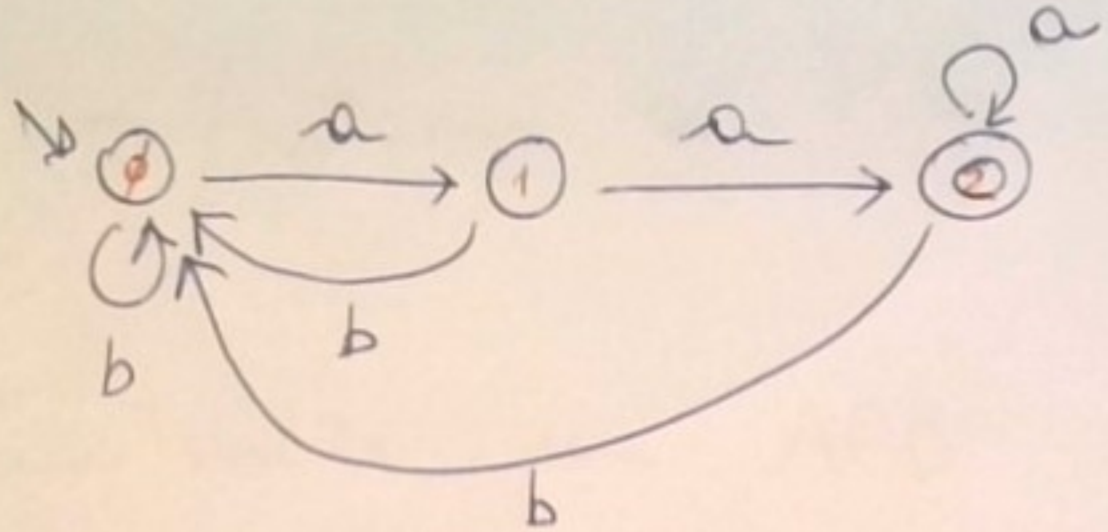



2.



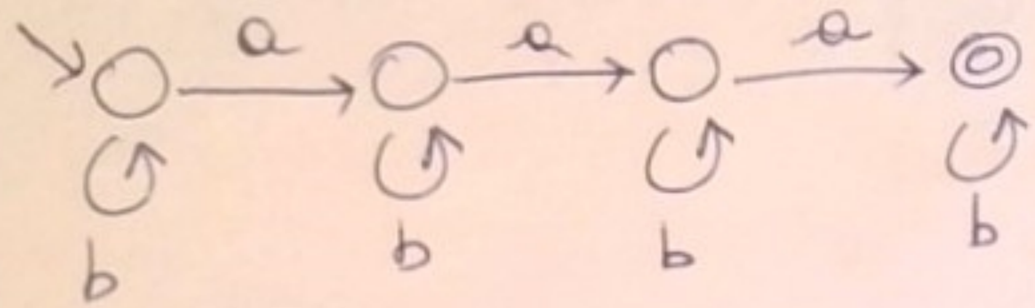

non è deterministico

2.

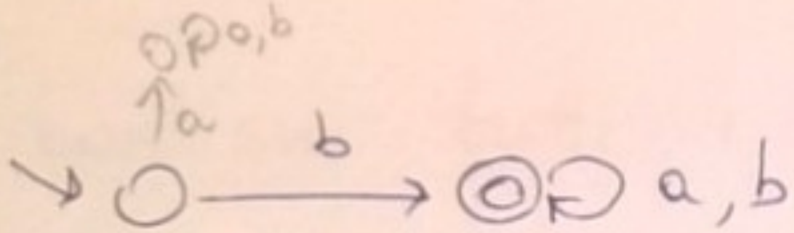


gli stati contano (con soglia 2) quante 'a' consecutive ho letto

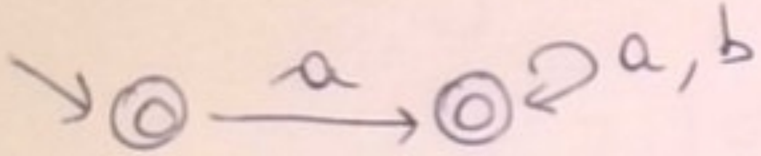
3.



4.



5.



NB: $\varepsilon \in \mathcal{L}_4$

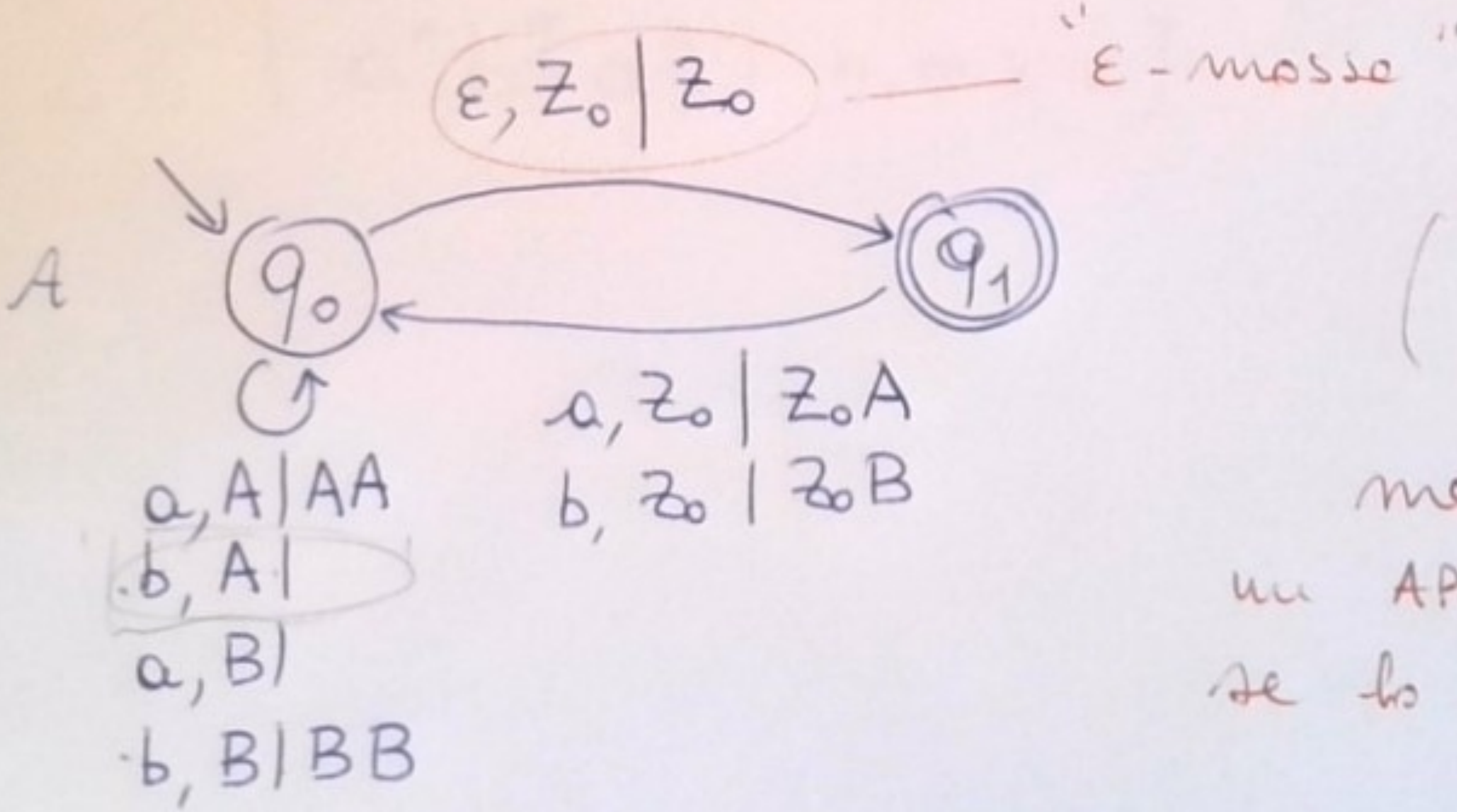
# — Automi a Pila (Deterministici) —

**Es** Costruire un APD che riconosca

$$L = \{ w \in \{a,b\}^* \mid \#_a(w) = \#_b(w) \} \leftarrow \quad \varepsilon \in L$$

numero di 'a' in w

L contiene tutti i soli gli anagrammi delle stringhe di $\{a^n b^n \mid n \in \mathbb{N}\}$




(NB: si potrebbe prendere $q_1$ come stato iniziale, ma volevo farvi vedere che un APD può accettare ε anche se lo stato iniziale non è finale)

Come fa A ad accettare abbbaa ?

$$
\begin{array}{ccccccccccccccc}
q_0 & \xrightarrow{\varepsilon} & q_1 & \xrightarrow{a} & q_0 & \xrightarrow{b} & q_0 & \xrightarrow{\varepsilon} & q_1 & \xrightarrow{b} & q_0 & \xrightarrow{b} & q_0 & \longrightarrow \\
\boxed{Z_0} & & \boxed{Z_0} & & \boxed{\begin{matrix} A \\ Z_0 \end{matrix}} & & \boxed{Z_0} & & \boxed{Z_0} & & \boxed{\begin{matrix} B \\ Z_0 \end{matrix}} & & \boxed{\begin{matrix} B \\ B \\ Z_0 \end{matrix}} &
\end{array}
$$

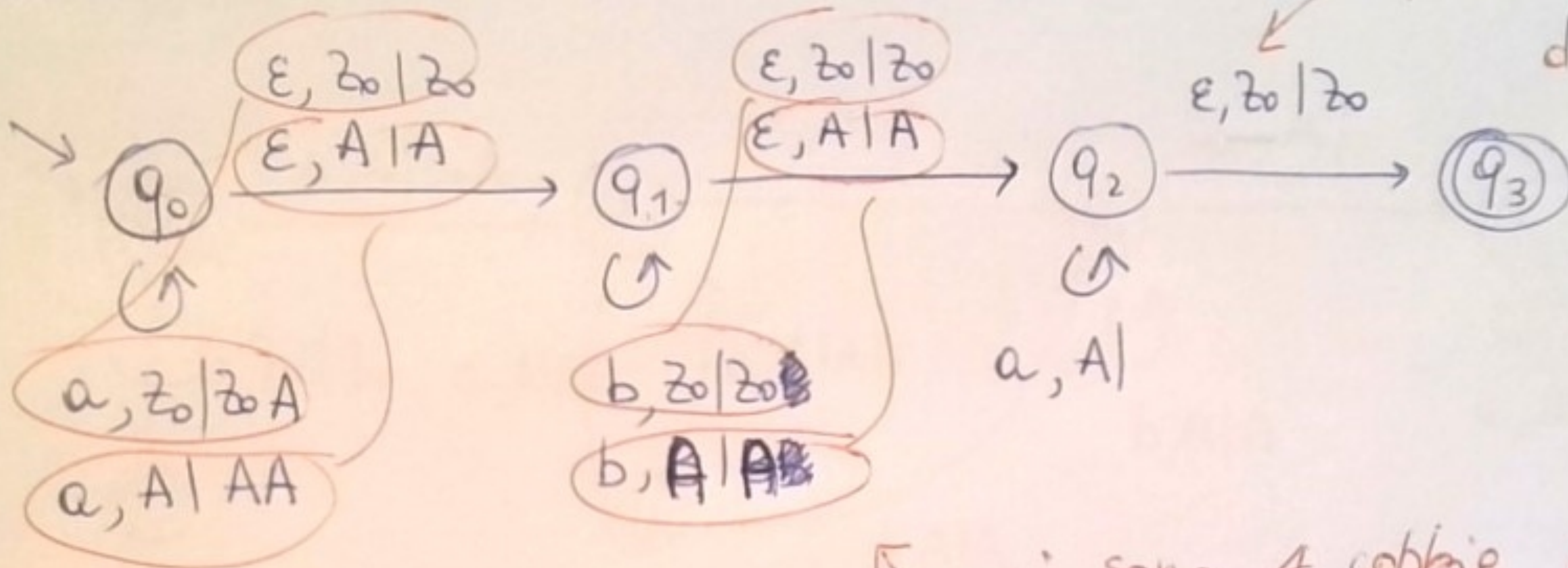
$$
\begin{array}{ccccccc}
\xrightarrow{a} & q_0 & \xrightarrow{a} & q_0 & \xrightarrow{\varepsilon} & q_1 \\
 & \boxed{\begin{matrix} B \\ Z_0 \end{matrix}} & & \boxed{Z_0} & & \boxed{Z_0}
\end{array}
$$

__Es__ Costruire un APD che riconosce

$$\mathcal{L} = \{ a^n b^m a^n \mid n, m \geq 0 \}.$$

Idea:

questa $\varepsilon$-mosse è deterministica

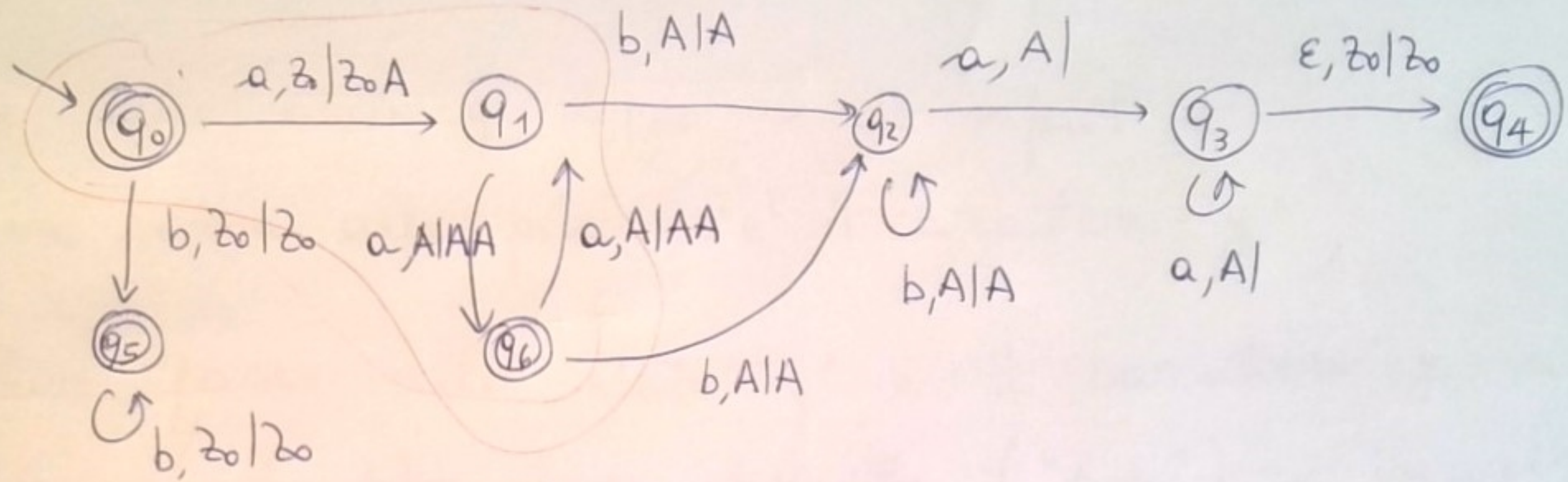



ci sono 4 coppie di mosse in alternativa: è un automa NON deterministico

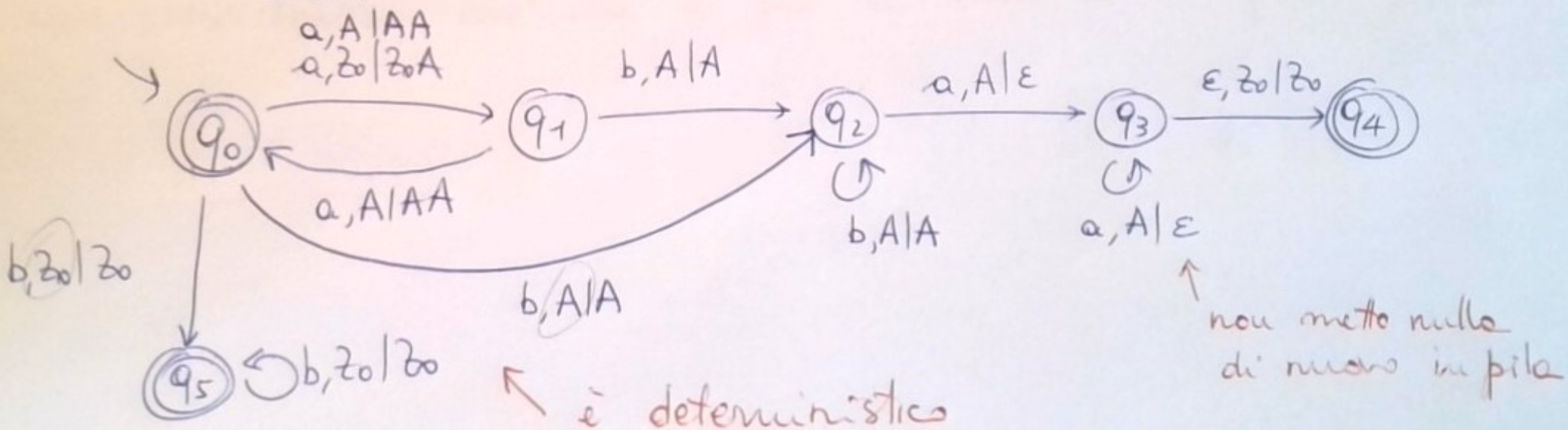
Proviamo a riscrive $\mathcal{L}$:

$$\mathcal{L} = \varepsilon + b^+ + (aa)^+ + \mathcal{L}' \quad \text{con} \quad \mathcal{L}' = \{ a^n b^m a^n \mid n, m > 0 \}$$

Così è più facile:




Si può fare diversamente, anche riducendo il numero di stati:

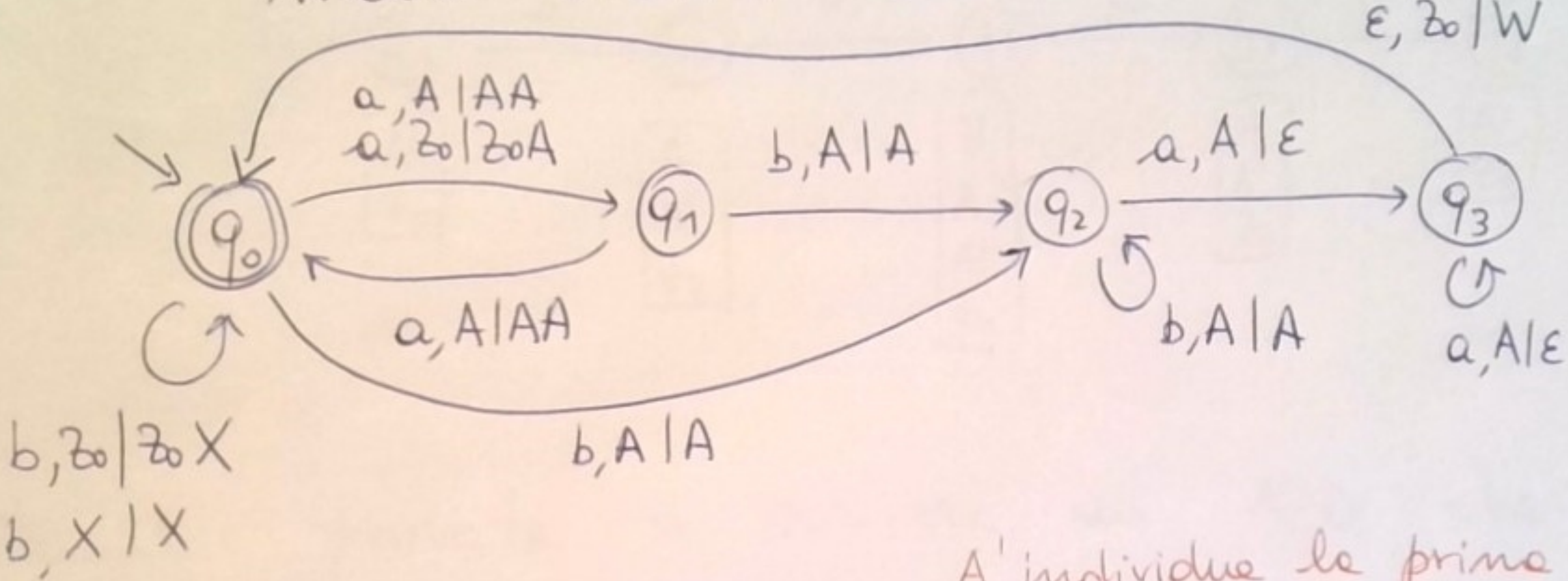


Come "si legge" $q \xrightarrow{x,\, y \,/\, w} q'$ :

Se nello stato $q$ leggo $x$ da input e
in cima alla pila c'è il carattere $y$,

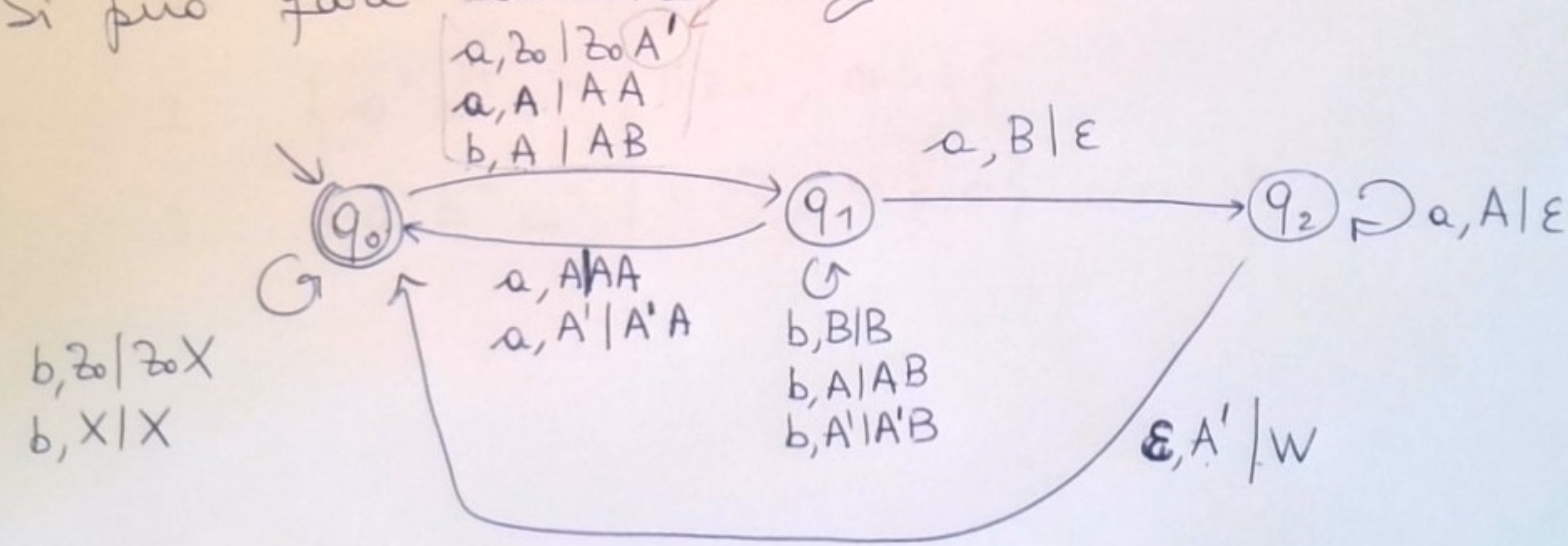
allora passo allo stato $q'$ e il carattere $y$ in
cima alla pila viene distrutto ("pop") e in cima
alla pila viene inserita la stringa $w$ avvengono simultaneamente
con in basso ciò che è più a sinistra ("push")

Torniamo a L; si può fare con ancora meno stati? Sì

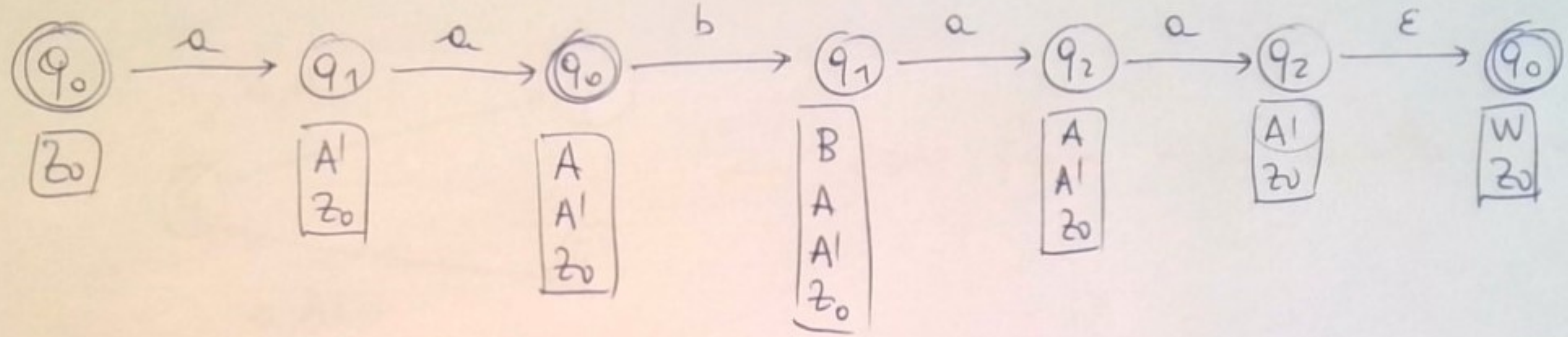



Si può fare ancora meglio? sì.

A' individua la prima 'a' letta



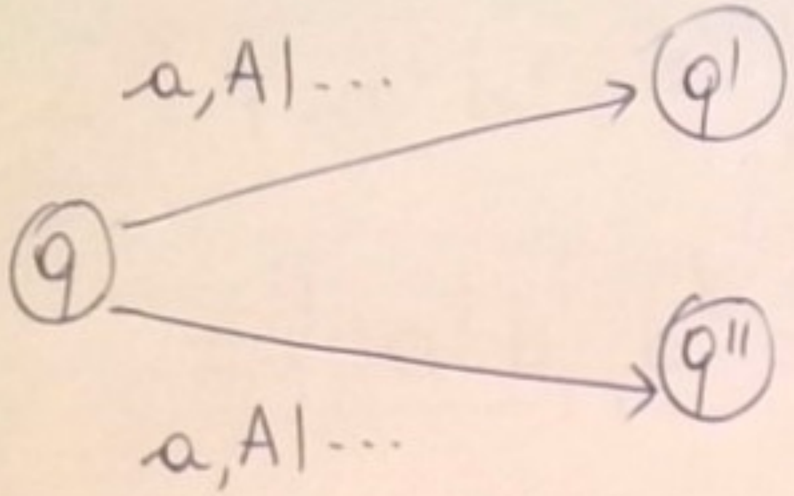
Proviamo aabaa



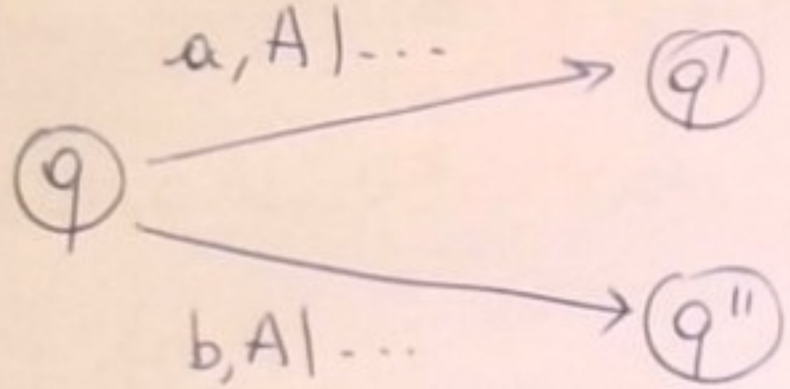

Compito: provate a scrivere un APD che riconosca

1. $\{a^n b^m a^n \mid n, m > 0\}$
2. $\{a^n b^m a^n \mid n \geq 0,\ m > 0\}$
3. $\{a^n b^m a^n \mid n > 0,\ m \geq 0\}$

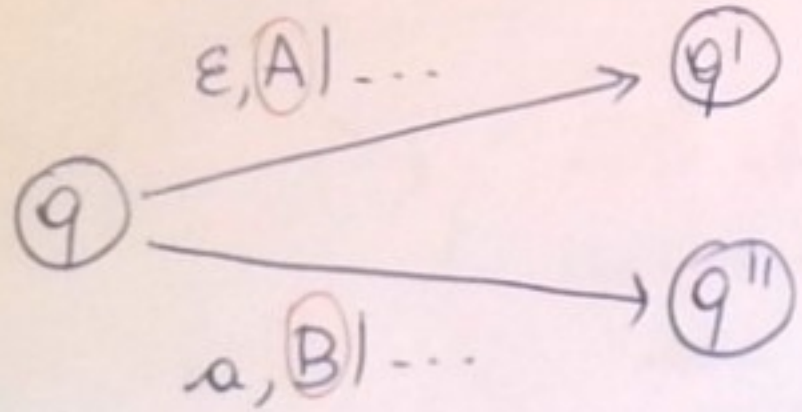
NB: ancora det. vs. non det.



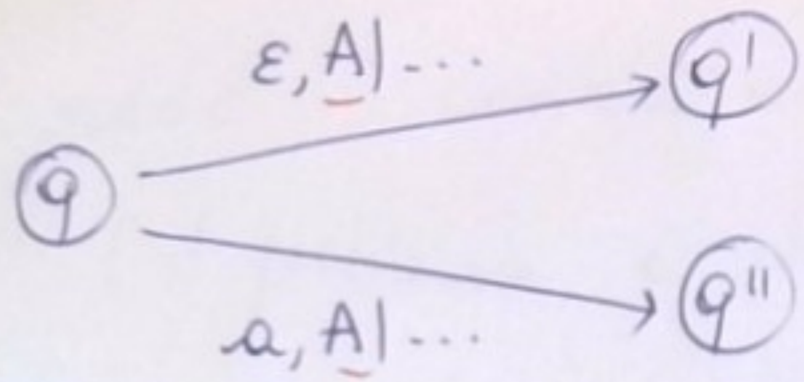

← devo fare una scelta: non det




← nessuna scelta: det




← c'è una ε-mossa, ma non devo fare alcuna scelta: det




← c'è una ε-mossa (non è un problema) ma devo fare una scelta: non det

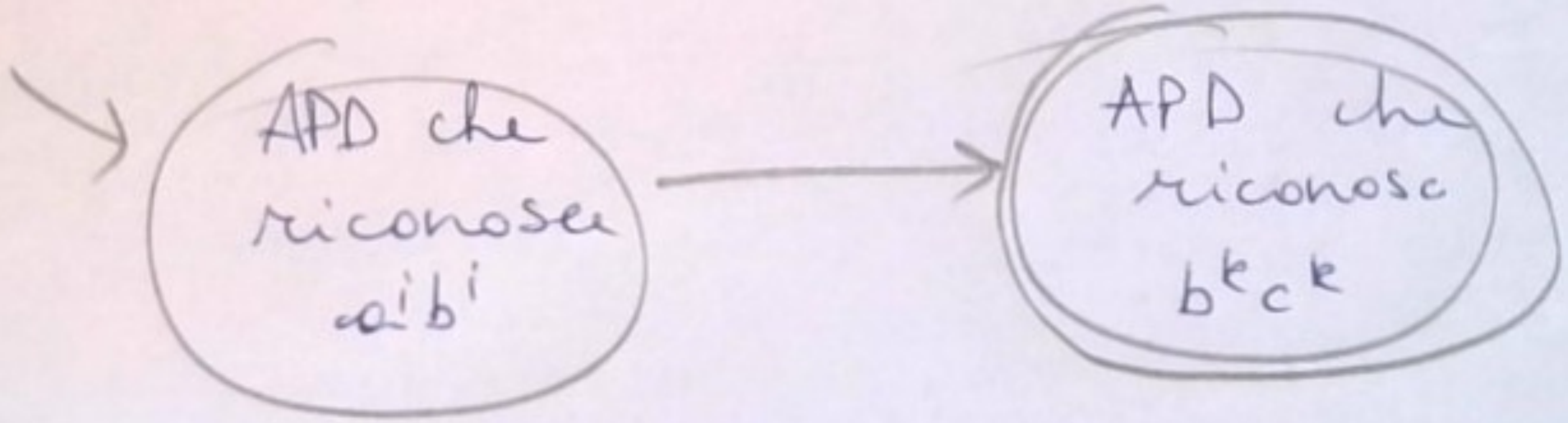
<u>Es</u> Costruire un APD che riconosce (se esiste un APD siffatto)

$$L = \{ a^i b^j c^k \mid i,j,k \in \mathbb{N},\ i+k=j \}$$

$$= \{ \underbrace{a^i b^i}\ \underbrace{b^k c^k} \mid i,k \in \mathbb{N} \}$$

Poiché $\{a^i b^i \mid i \geq 0\}$ e $\{b^k c^k \mid k \geq 0\}$ possono essere riconosciuti con APD, (forse) anche per $L$ si può fare

Idea




cioè "concateno" due APD: proviamoci
(il problema è che $i$ e $k$ possono essere 0).