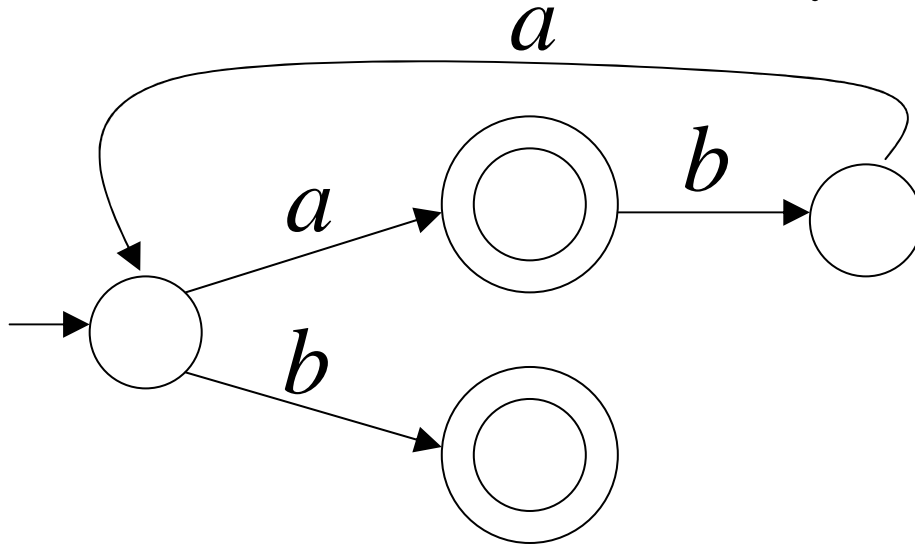


Um único estado final  
para AFN e AFD

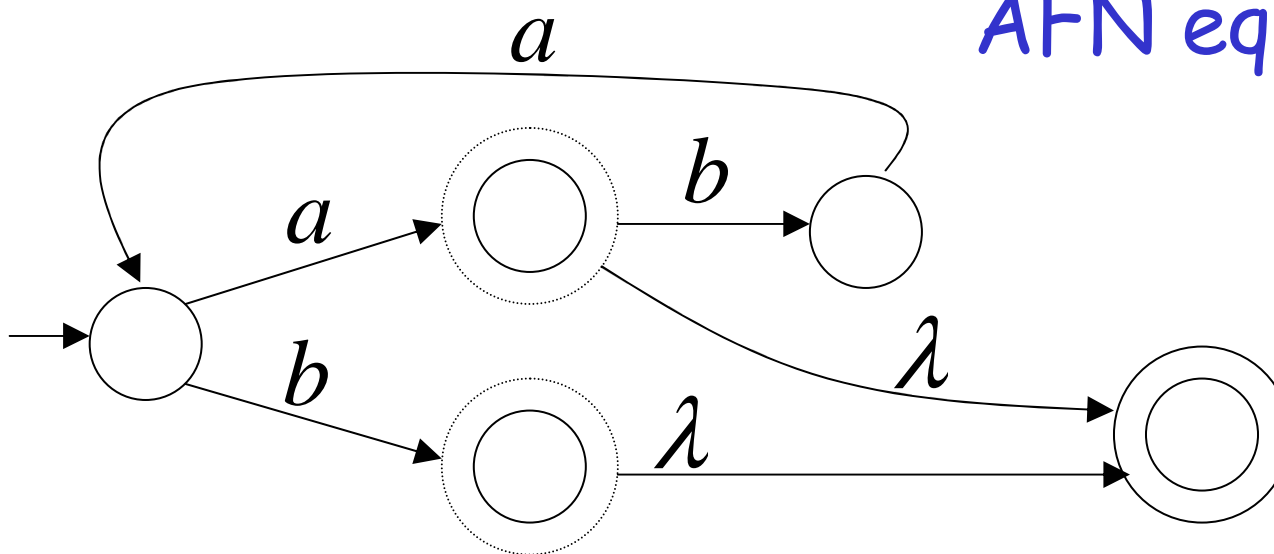
# Observação

Qualquer autômato finito (AFN ou AFD)  
pode ser convertido para um AFN equivalente  
com um único estado final

# Exemplo



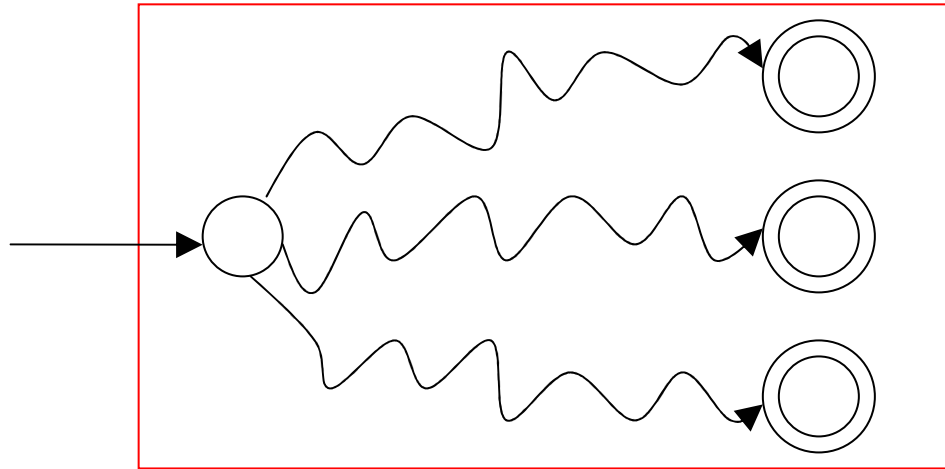
AFN



AFN equivalente

# Em geral

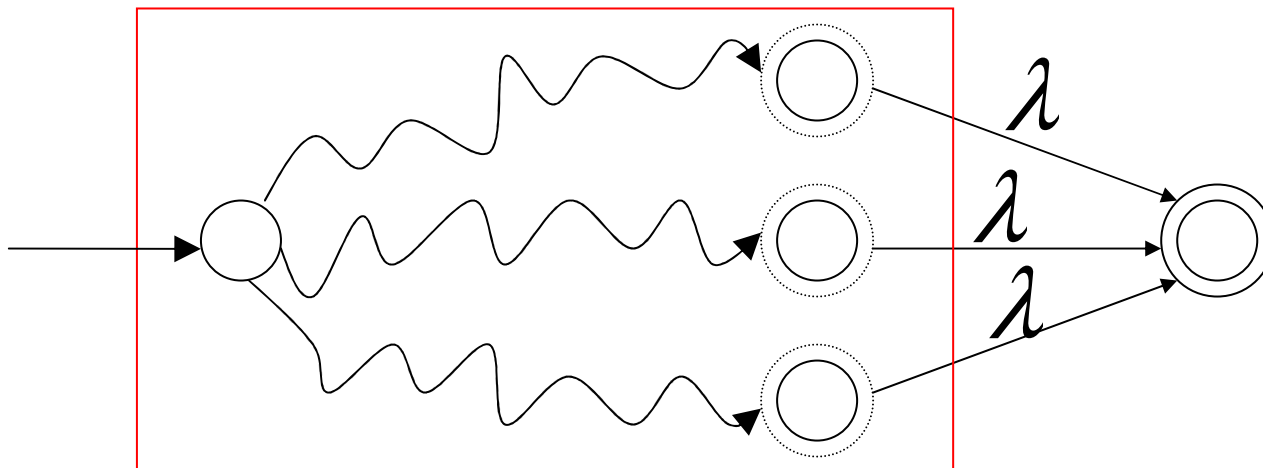
AFN



Atenção:

$$\lambda = \varepsilon$$

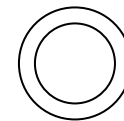
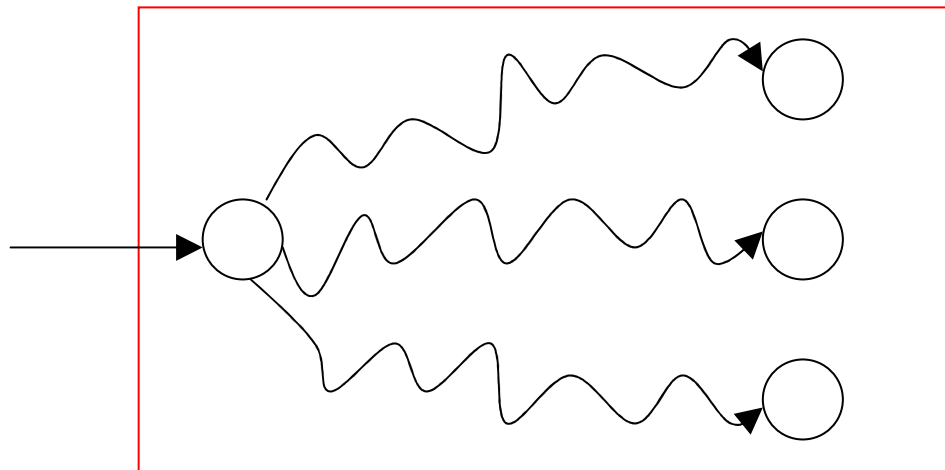
AFN equivalente



Único estado final

# Caso extremo

AFN sem estado final



Adicione um estado final sem transição

# Algumas propriedades de linguagens regulares

# Propriedades

Para linguagens regulares  $L_1$  e  $L_2$   
provaremos:

União:  $L_1 \cup L_2$

Concatenação:  $L_1 L_2$

Estrela:  $L_1^*$

São linguagens  
regulares

Dizemos que:

Linguagens regulares são **fechadas sobre**

União:  $L_1 \cup L_2$

Concatenação:  $L_1 L_2$

Estrela:  $L_1^*$

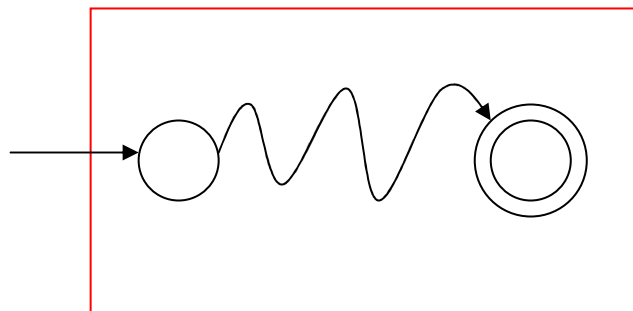
Linguagem regular  $L_1$

Linguagem regular  $L_2$

$$L(M_1) = L_1$$

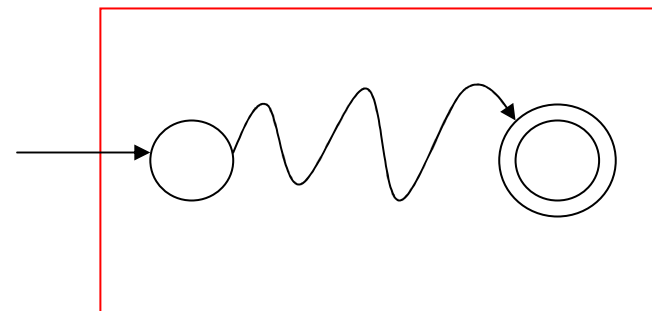
$$L(M_2) = L_2$$

AFN  $M_1$



Único estado final

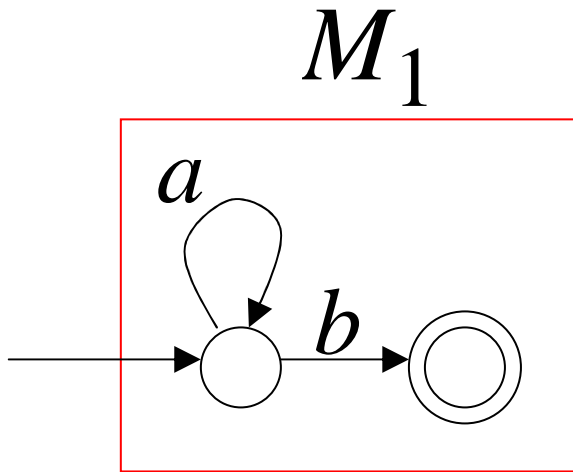
AFN  $M_2$



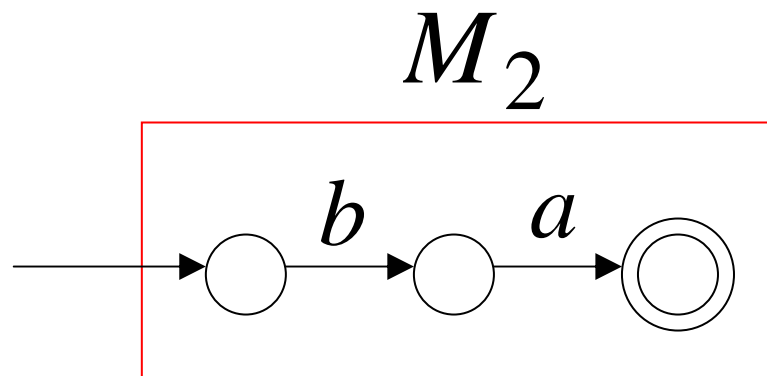
Único estado final

# Exemplo

$$L_1 = \{a^n b\}$$

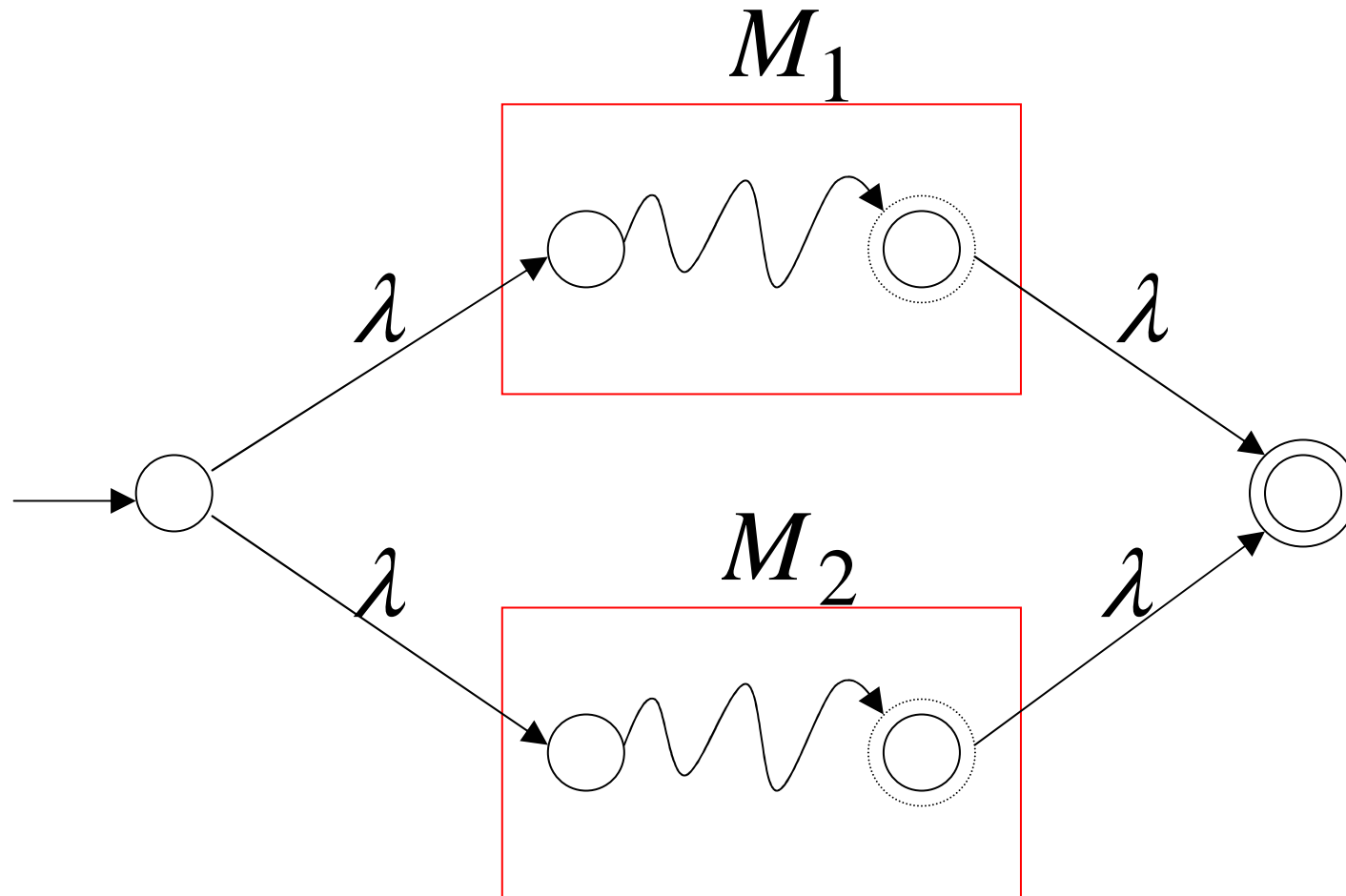


$$L_2 = \{ba\}$$



# União

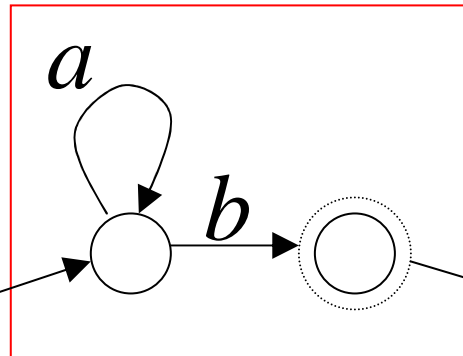
AFN para  $L_1 \cup L_2$



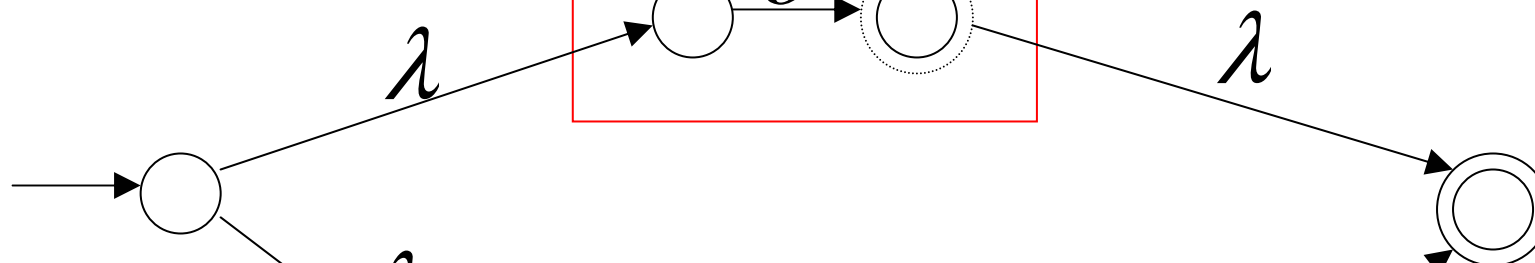
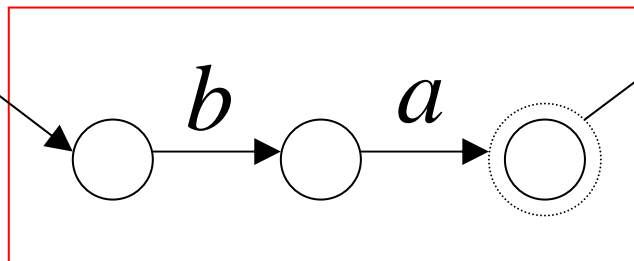
# Exemplo

AFN para  $L_1 \cup L_2 = \{a^n b\} \cup \{ba\}$

$L_1 = \{a^n b\}$

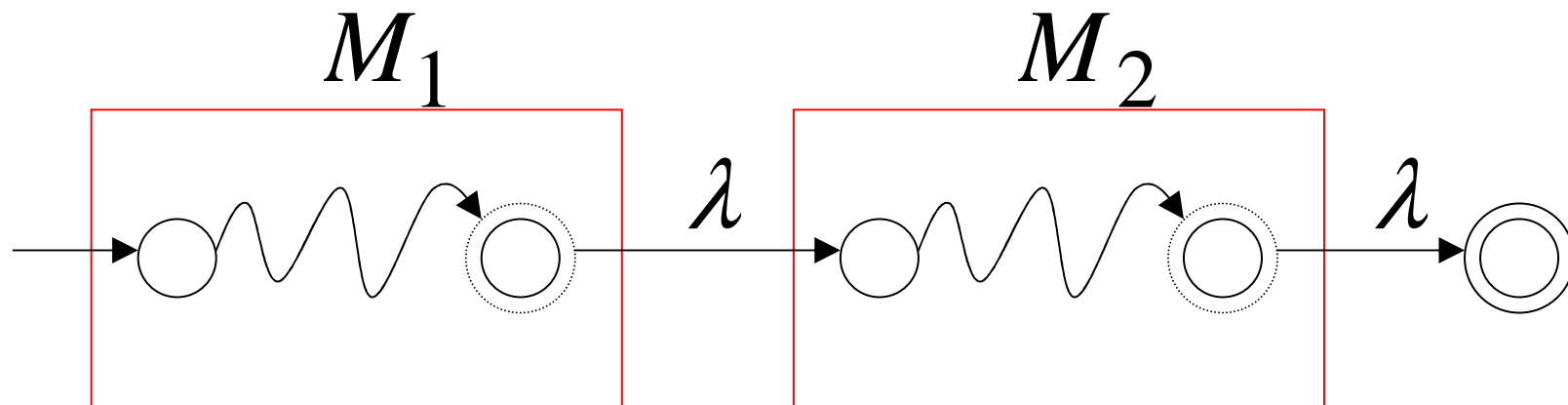


$L_2 = \{ba\}$



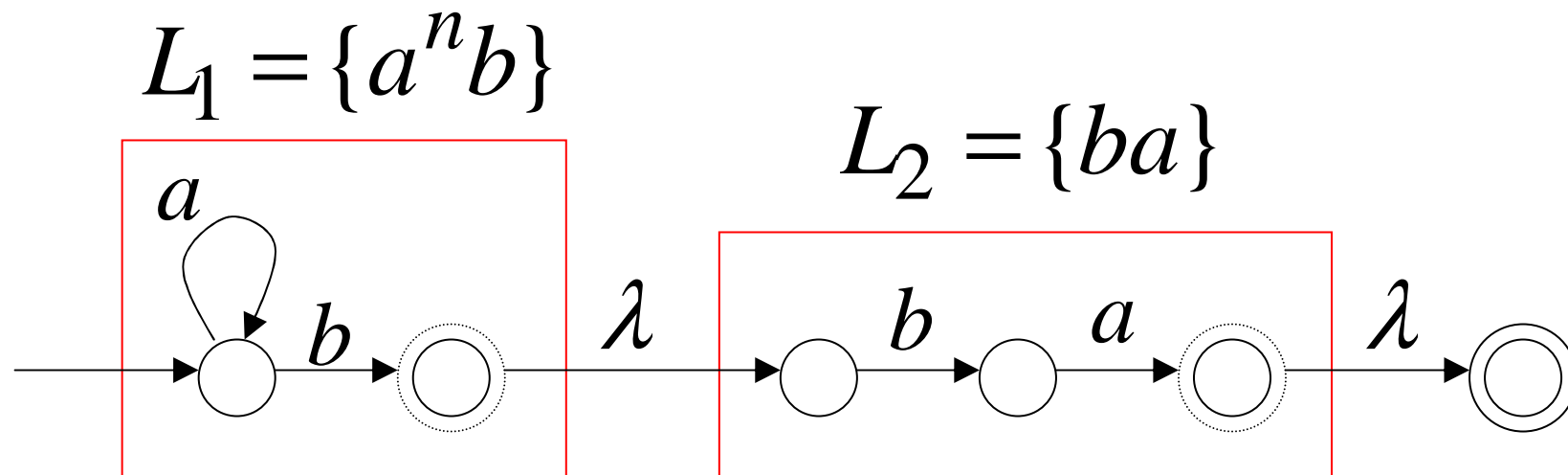
# Concatenação

AFN para  $L_1L_2$



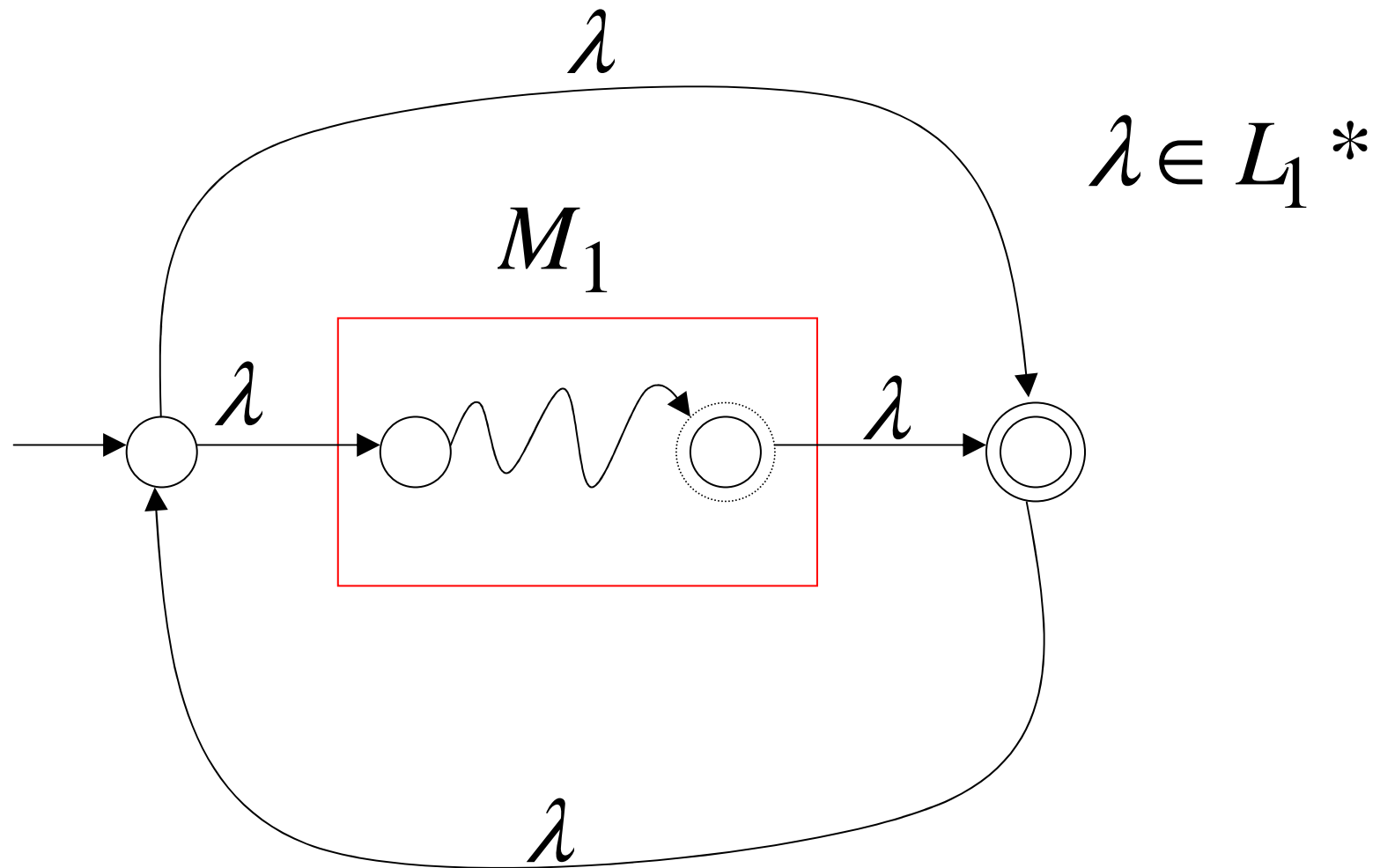
# Exemplo

AFN para  $L_1L_2 = \{a^n b\} \{ba\} = \{a^n bba\}$



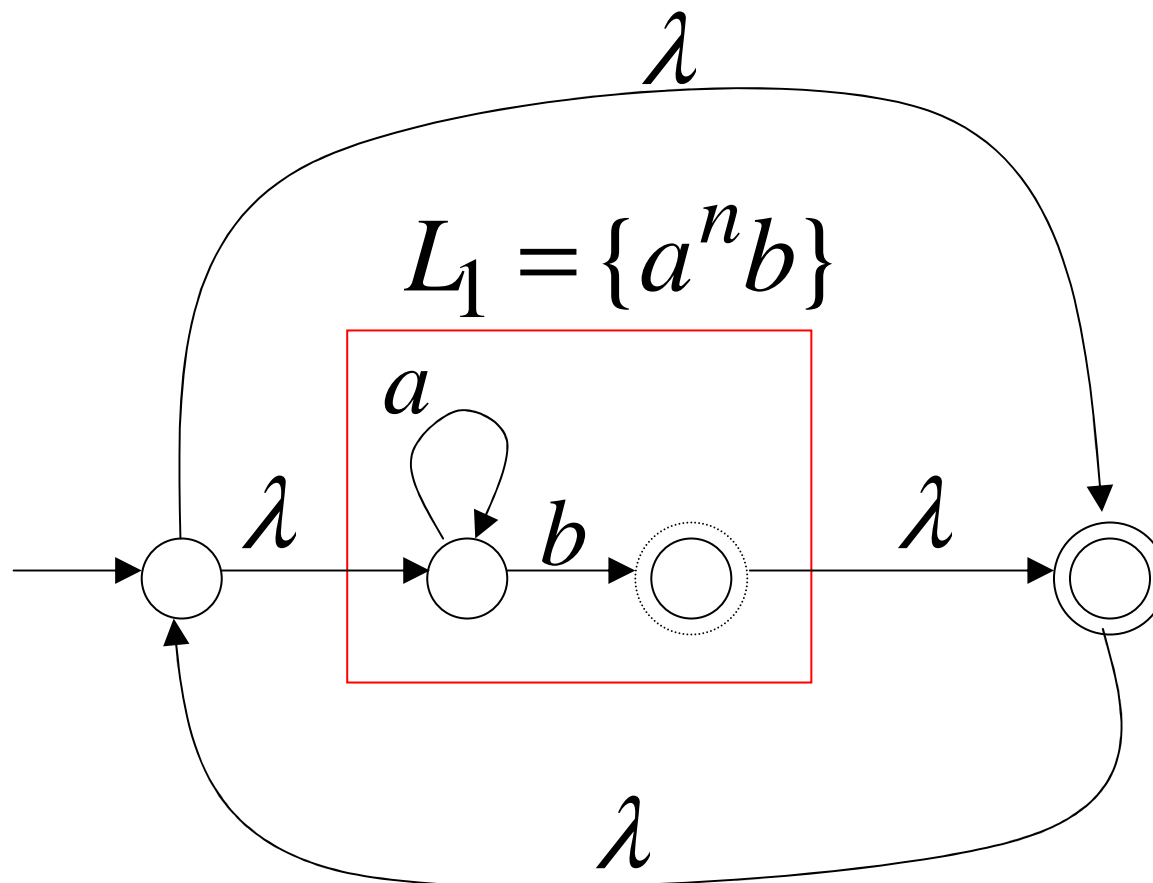
# Operação estrela

AFN para  $L_1^*$



# Exemplo

AFN para  $L_1^* = \{a^n b\}^*$



# Expressões Regulares

# Expressões Regulares

Expressões regulares (ER)

descrevem linguagens regulares (LR)

Exemplo:  $(a + b \cdot c)^*$

descreve a linguagem

$$\{a, bc\}^* = \{\lambda, a, bc, aa, abc, bca, \dots\}$$

# Definição Recursiva

ERs primitivas:  $\emptyset$ ,  $\lambda$ ,  $\alpha$

Dada uma ER  $r_1$  e  $r_2$

$r_1 + r_2$   
 $r_1 \cdot r_2$   
 $r_1^*$   
 $(r_1)$

São ERs

# Exemplos

É ER:  $(a + b \cdot c)^* \cdot (c + \emptyset)$

Não é ER:  $(a + b +)$

# Linguagens de ER

$L(r)$  : linguagens da ER  $r$

Exemplo

$$L((a + b \cdot c)^*) = \{\lambda, a, bc, aa, abc, bca, \dots\}$$

# Definição

Para ERs primitivas:

$$L(\emptyset) = \emptyset$$

$$L(\lambda) = \{\lambda\}$$

$$L(a) = \{a\}$$

## Definição (continuação)

Para ERs  $r_1$  e  $r_2$

$$L(r_1 + r_2) = L(r_1) \cup L(r_2)$$

$$L(r_1 \cdot r_2) = L(r_1) L(r_2)$$

$$L(r_1^*) = (L(r_1))^*$$

$$L((r_1)) = L(r_1)$$

# Exemplo

ER:  $(a + b) \cdot a^*$

$$\begin{aligned} L((a + b) \cdot a^*) &= L((a + b)) L(a^*) \\ &= L(a + b) L(a^*) \\ &= (L(a) \cup L(b)) (L(a))^* \\ &= (\{a\} \cup \{b\}) (\{a\})^* \\ &= \{a, b\} \{\lambda, a, aa, aaa, \dots\} \\ &= \{a, aa, aaa, \dots, b, ba, baa, \dots\} \end{aligned}$$

# Exemplo

ER  $r = (a + b)^* (a + bb)$

$$L(r) = \{a, bb, aa, abb, ba, bbb, \dots\}$$

# Exemplo

$$\text{ER} \quad r = (aa)^* (bb)^* b$$

$$L(r) = \{a^{2n} b^{2m} b : n, m \geq 0\}$$

# Exemplo

$$\text{ER } r = (0+1)^* 00 (0+1)^*$$

$L(r) = \{ \text{Todas as strings com pelo menos dois 0 consecutivos} \}$

# Exemplo

$$\text{ER } r = (1 + 01)^* (0 + \lambda)$$

$L(r) = \{ \text{Todas as strings sem} \\ \text{dois 0 consecutivos} \}$

# LR equivalentes

Definição:

As ERs  $r_1$  e  $r_2$

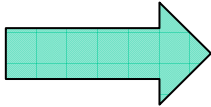
são equivalentes se  $L(r_1) = L(r_2)$

# Exemplo

$L = \{ \text{Todas as strings sem dois 0 consecutivos} \}$

$$r_1 = (1 + 01)^* (0 + \lambda)$$

$$r_2 = (1^* 011^*)^* (0 + \lambda) + 1^* (0 + \lambda)$$

$L(r_1) = L(r_2) = L$    $r_1$  e  $r_2$   
São ERs  
equivalentes

# Expressões regulares

## Linguagens regulares

# Teorema

$$\left\{ \begin{array}{l} \text{Linguagens} \\ \text{Geradas por} \\ \text{Expressões Regulares} \end{array} \right\} = \left\{ \begin{array}{l} \text{Linguagens} \\ \text{Regulares} \end{array} \right\}$$

# Teorema - Parte 1

$$\left\{ \begin{array}{l} \text{Linguagens} \\ \text{Geradas por} \\ \text{Expressões Regulares} \end{array} \right\} \subseteq \left\{ \begin{array}{l} \text{Linguagens} \\ \text{Regulares} \end{array} \right\}$$

1. Para qualquer expressão regular  $r$   
a linguagem  $L(r)$  é regular

## Teorema - Parte 2

$$\left\{ \begin{array}{l} \text{Linguagens} \\ \text{Geradas por} \\ \text{Expressões Regulares} \end{array} \right\} \equiv \left\{ \begin{array}{l} \text{Linguagens} \\ \text{Regulares} \end{array} \right\}$$

2. Para qualquer linguagem  $L$  existe  
uma ER  $r$  com  $L(r) = L$

# Prova - Parte 1

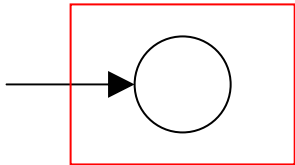
1. Para qualquer ER  $r$   
a linguagem  $L(r)$  é regular

Prova por indução no tamanho de  $r$

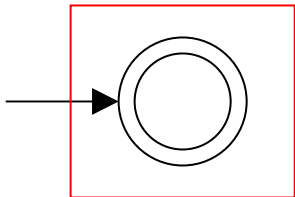
# Indutiva Básica

ERs primitivas:  $\emptyset$ ,  $\lambda$ ,  $a$

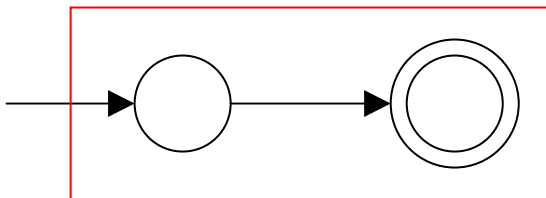
AFNs



$$L(M_1) = \emptyset = L(\emptyset)$$



$$L(M_2) = \{\lambda\} = L(\lambda)$$



$$L(M_3) = \{a\} = L(a)$$

Linguagens  
Regulares

# Hipotese Indutiva

Assuma

Para ERs  $r_1$  e  $r_2$

que

$L(r_1)$  e  $L(r_2)$  são LRs

# Passo indutivo

Devemos provar que:

$$L(r_1 + r_2)$$

$$L(r_1 \cdot r_2)$$

$$L(r_1^*)$$

$$L((r_1))$$

São LR's

ERs por definição:

$$L(r_1 + r_2) = L(r_1) \cup L(r_2)$$

$$L(r_1 \cdot r_2) = L(r_1) L(r_2)$$

$$L(r_1^*) = (L(r_1))^*$$

$$L((r_1)) = L(r_1)$$

Pela hipótese indutiva sabemos que:

$L(r_1)$  e  $L(r_2)$  são LRs

Sabemos também que:

LRs são fechadas sobre

*união*  $L(r_1) \cup L(r_2)$

*concatenação*  $L(r_1) L(r_2)$

*estrela*  $(L(r_1))^*$

Então:

$$L(r_1 + r_2) = L(r_1) \cup L(r_2)$$

$$L(r_1 \cdot r_2) = L(r_1) L(r_2)$$

$$L(r_1^*) = (L(r_1))^*$$

São LR's

E trivialmente:

$$L((r_1)) \text{ É}$$

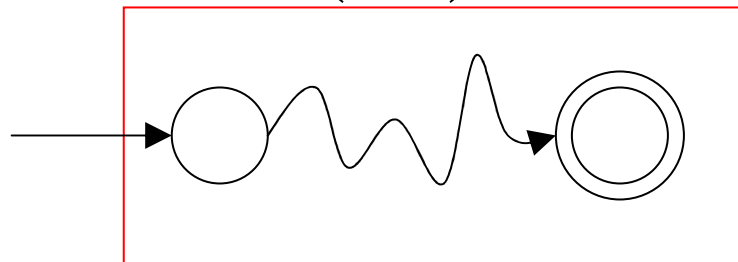
## Prova - Parte 2

2. Para qualquer LR  $L$  existe  
uma ER  $r$  com  $L(r) = L$

Prova por construção da ER

Desde que  $L$  é regular tomamos um AFN  $M$  que aceita  $L$

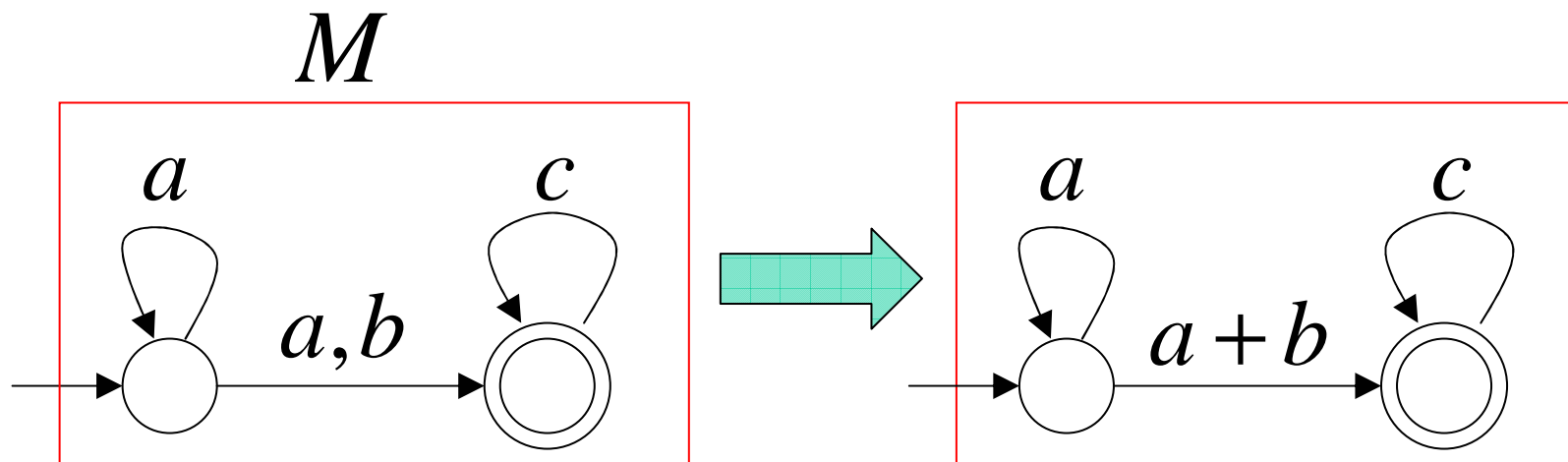
$$L(M) = L$$



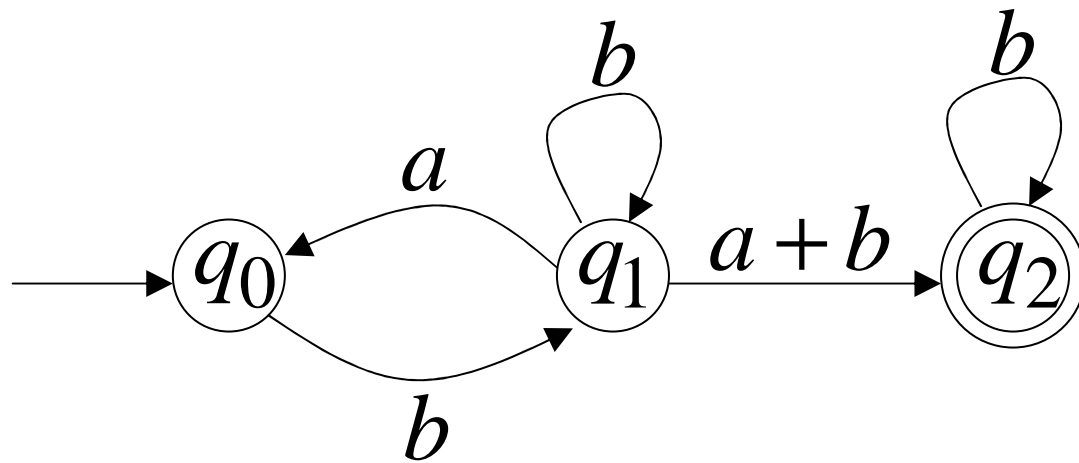
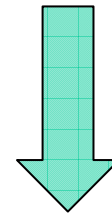
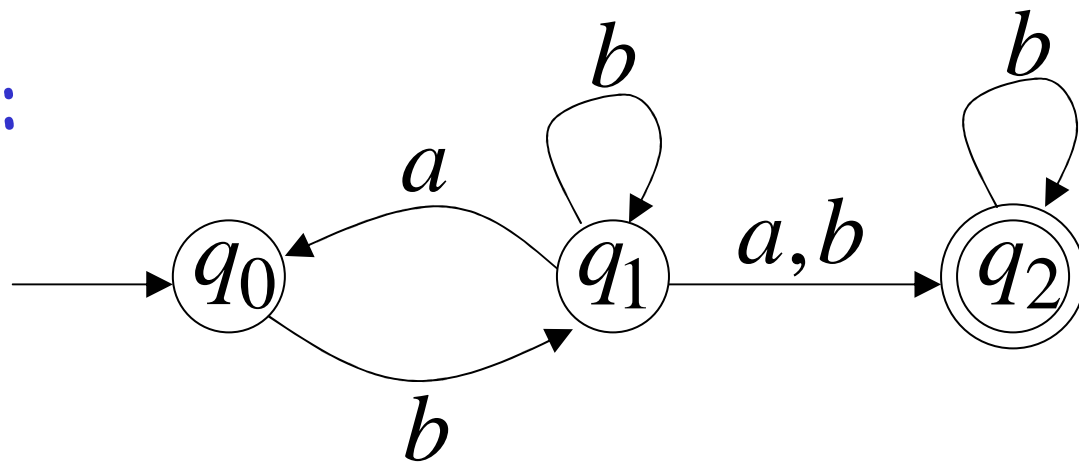
Único estado final

De  $M$  construímos um  
**Grafo de transição generalizado**  
rótulos das transições  
são expressões regulares

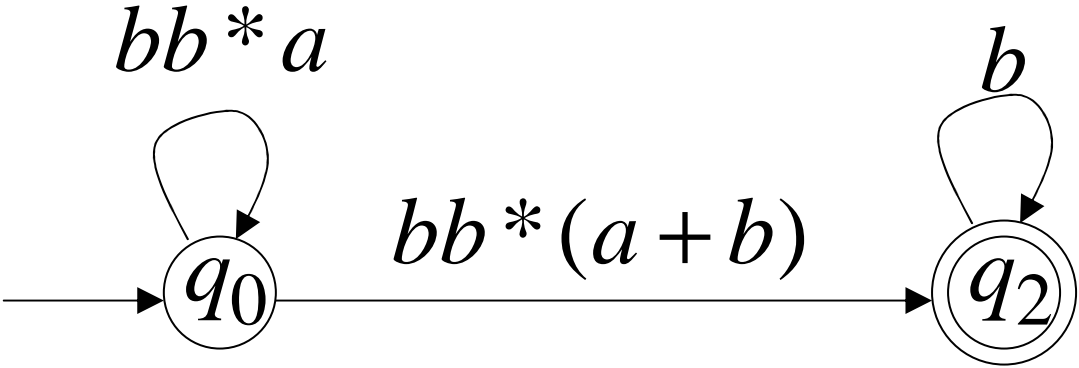
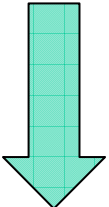
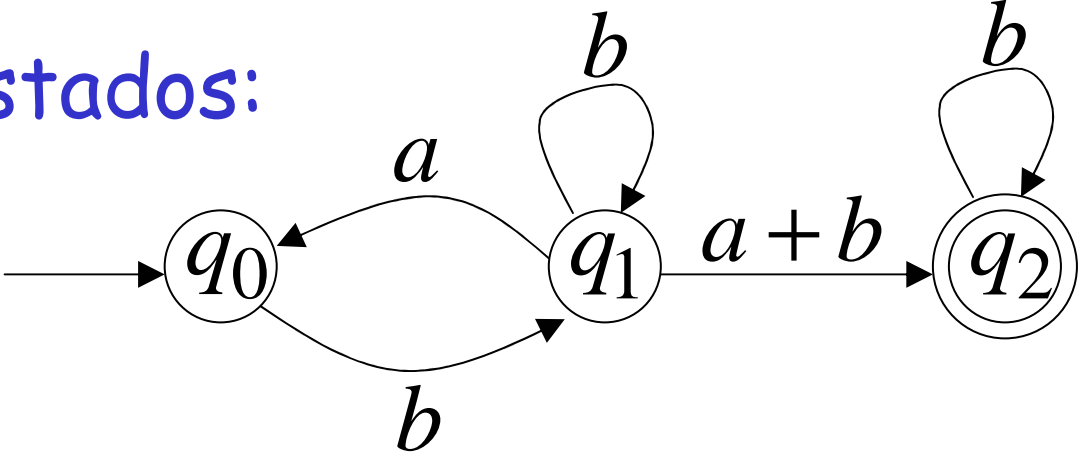
Exemplo:



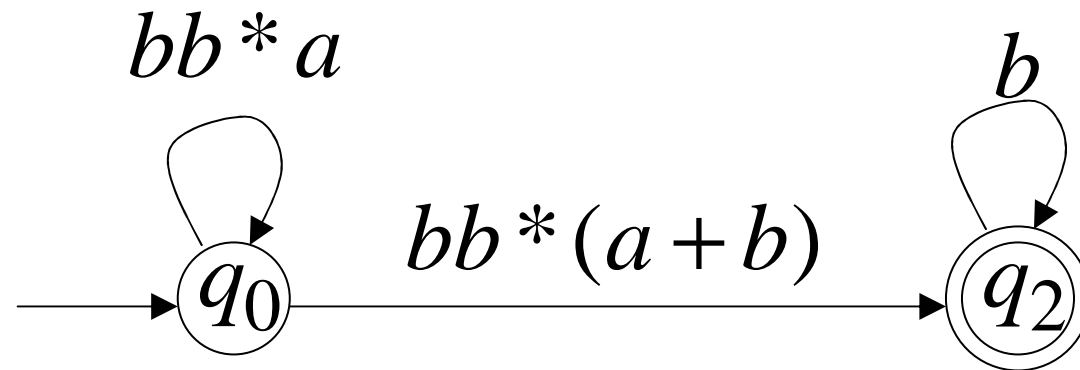
Outro exemplo:



Reduzindo os estados:



Resultando na Expressão Regular:

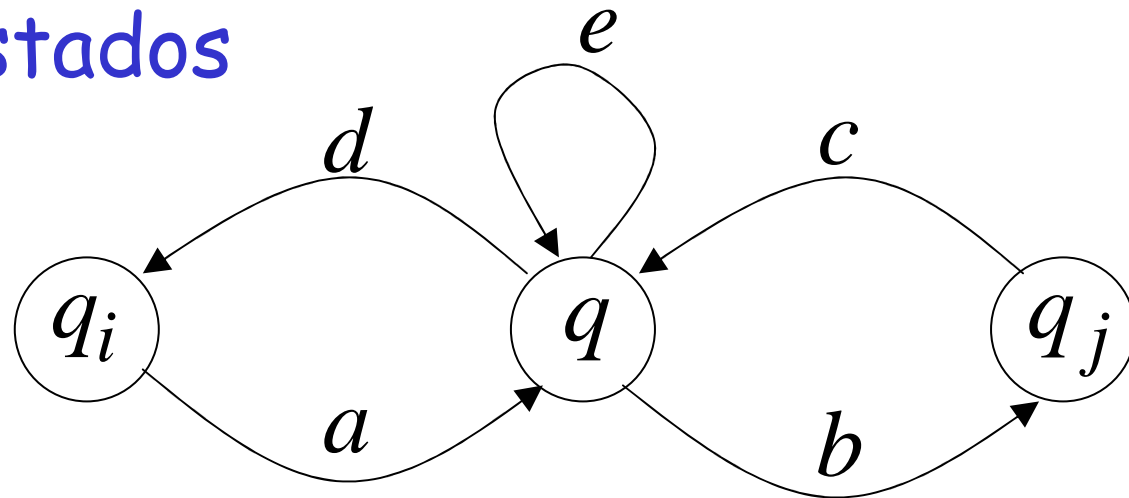


$$r = (bb^*a)^*bb^*(a+b)b^*$$

$$L(r) = L(M) = L$$

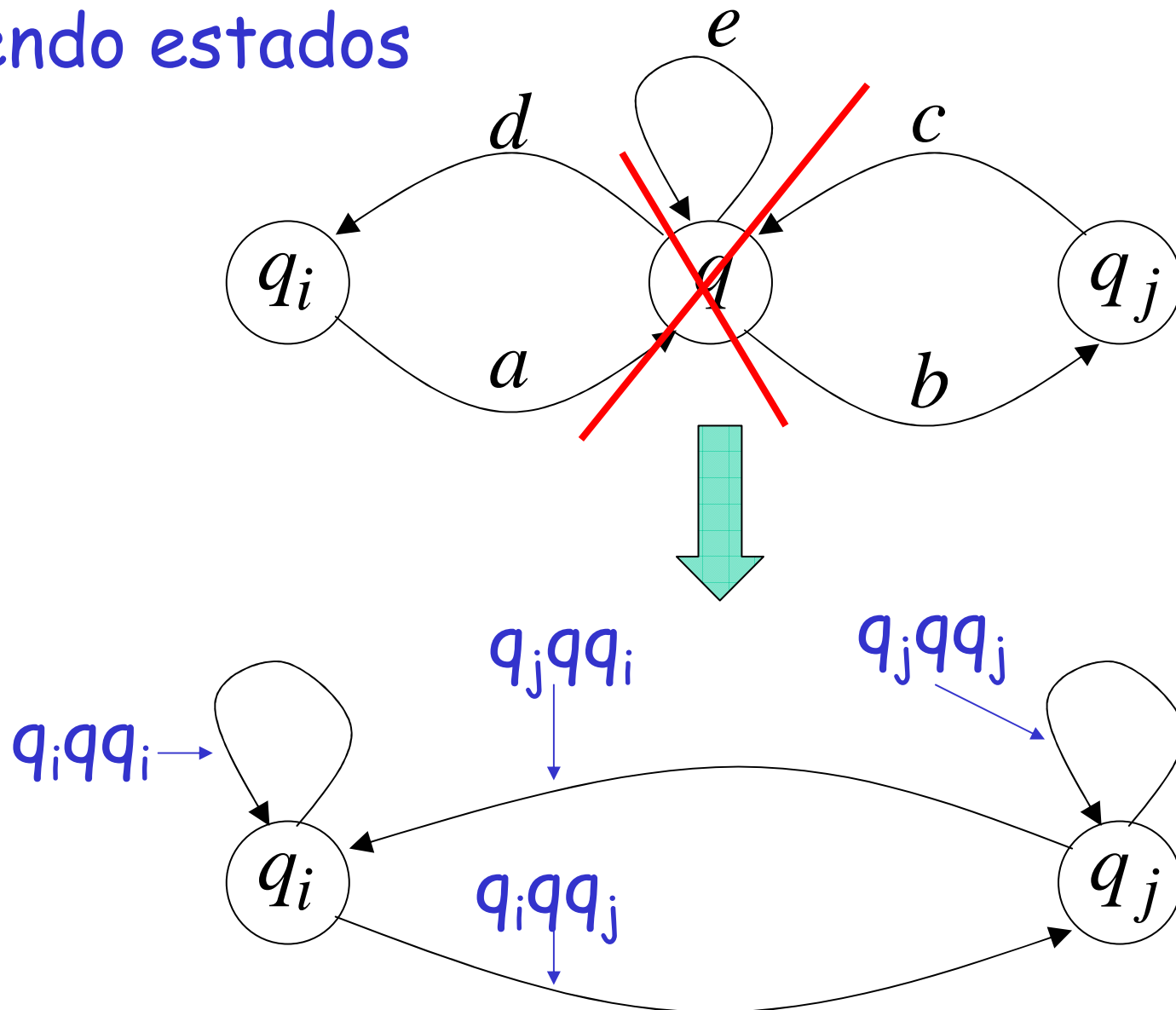
# Em geral

Removendo estados



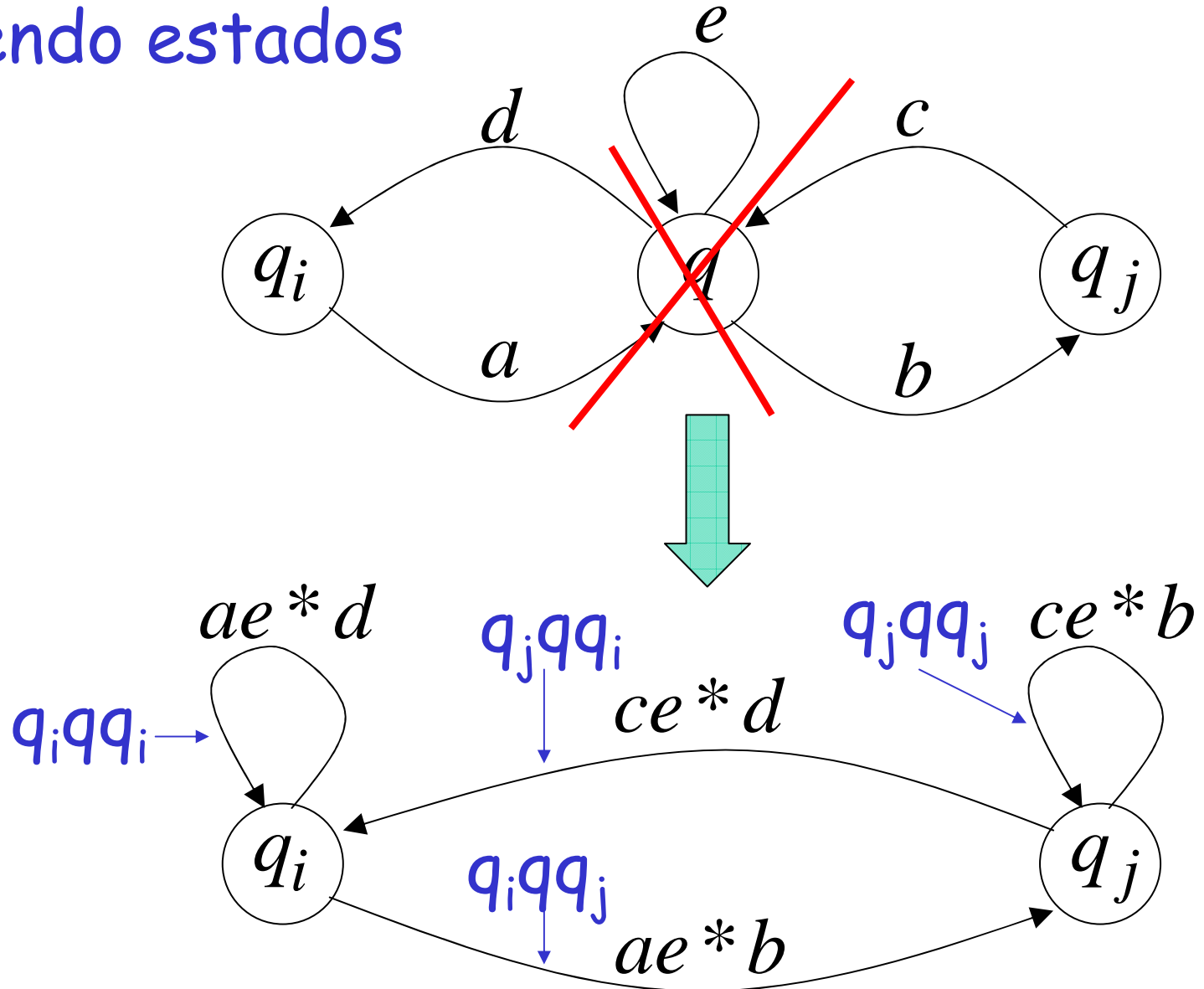
# Em geral

Removendo estados

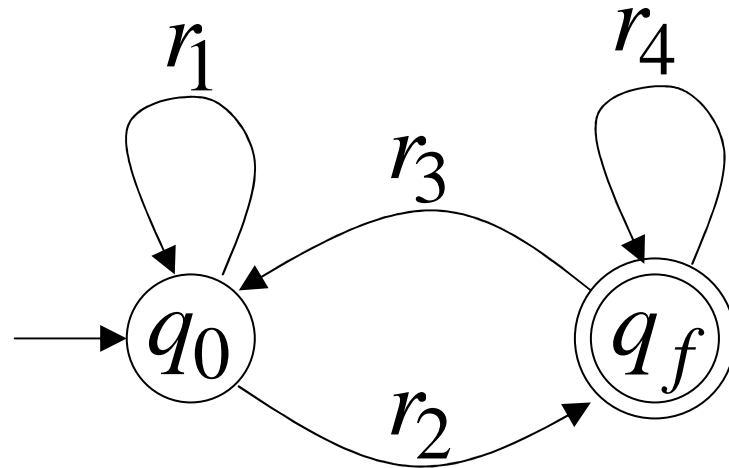


# Em geral

Removendo estados



## Obtendo uma expressão regular final



$$r = r_1^* r_2 (r_4 + r_3 r_1^* r_2)^*$$

$$L(r) = L(M) = L$$