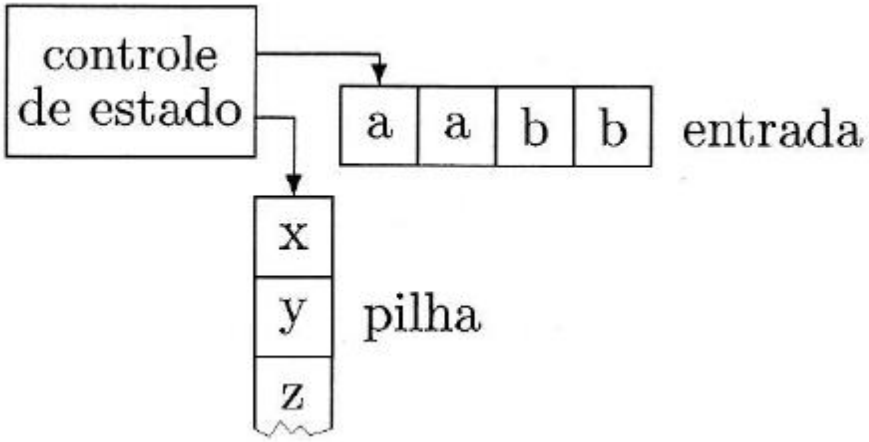
ACH2043
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

Aula 10

Cap. 2.2 – Autômato com pilha (cont.)

Profa. Ariane Machado Lima
ariane.machado@usp.br

Cap 2.2 – Autômato com pilha (AP)



Exemplo

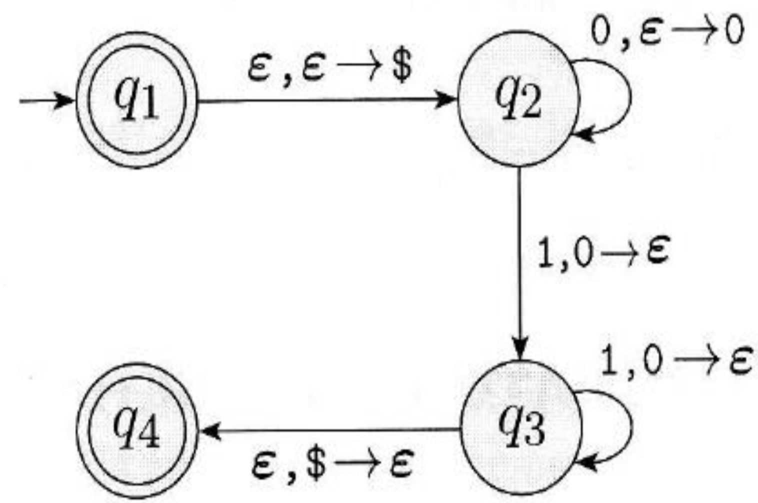
$\{0^n 1^n | n \geq 0\}$. Suponha que M_1 seja $(Q, \Sigma, \Gamma, \delta, q_1, F)$

$Q = \{q_1, q_2, q_3, q_4\}$,

$\Sigma = \{0,1\}$,

$\Gamma = \{0, \$\}$,

$F = \{q_1, q_4\}$, e



δ é dada pela tabela abaixo, na qual entradas em branco significam $\emptyset$.

Entrada:	0			1			ε		
Pilha:	0	\$	ε	0	\$	ε	0	\$	ε
q_1									$\{(q_2, \$)\}$
q_2			$\{(q_2, 0)\}$	$\{(q_3, \varepsilon)\}$					
q_3				$\{(q_3, \varepsilon)\}$				$\{(q_4, \varepsilon)\}$	
q_4									

Equivalência entre APN e GLC

TEOREMA 2.20 ....

Uma linguagem é livre-do-contexto se e somente se algum autômato com pilha a reconhece.

Autômato com pilha NÃO DETERMINÍSTICO!!!

Equivalência entre APN e GLC

LEMA 2.21 ....

Se uma linguagem é livre-do-contexto, então algum autômato com pilha a reconhece.

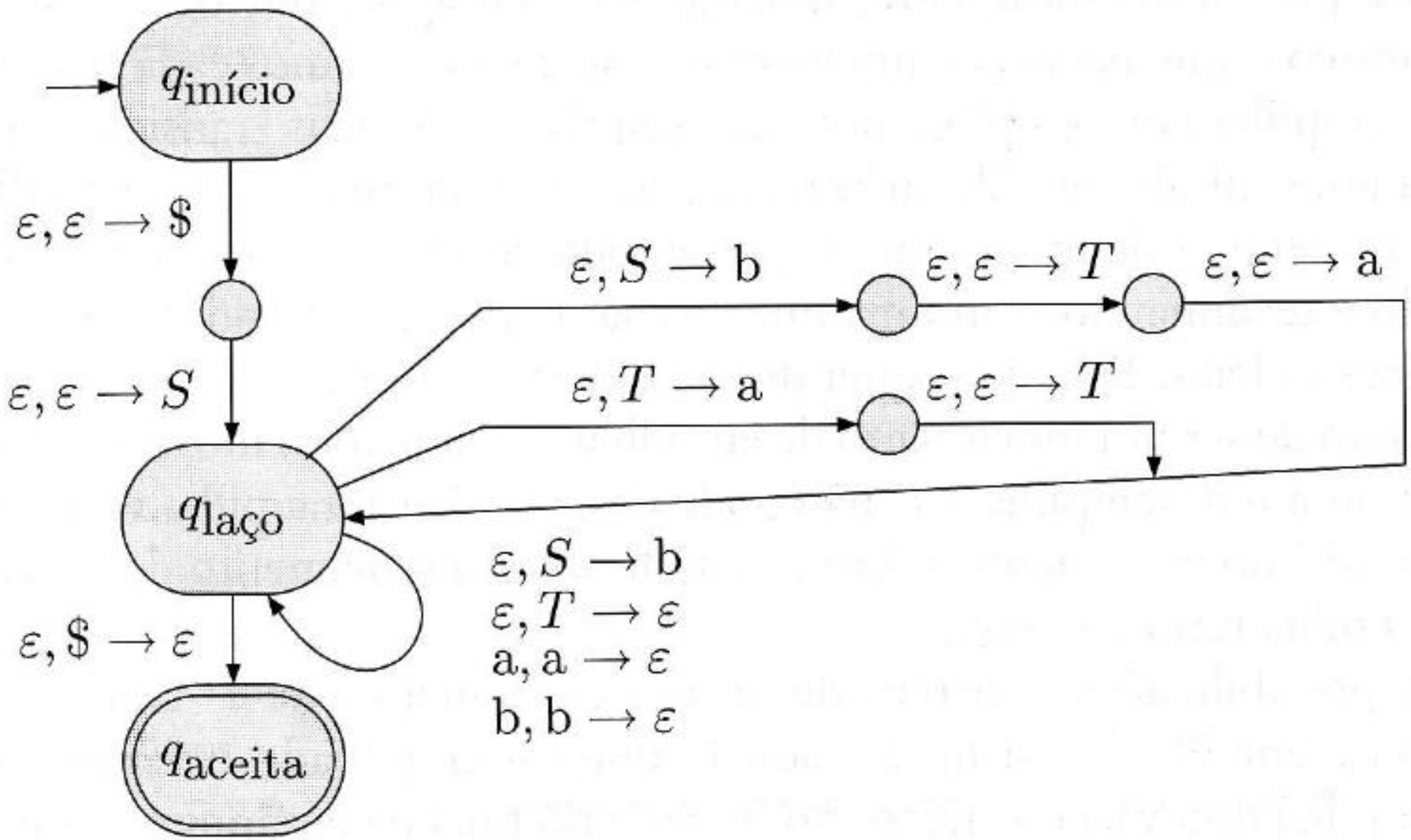
Ideia da prova:

- Uma LLC é gerada por uma GLC
- Mostrar como converter uma GLC em um APN equivalente

Conversão GLC em APN (ideia)

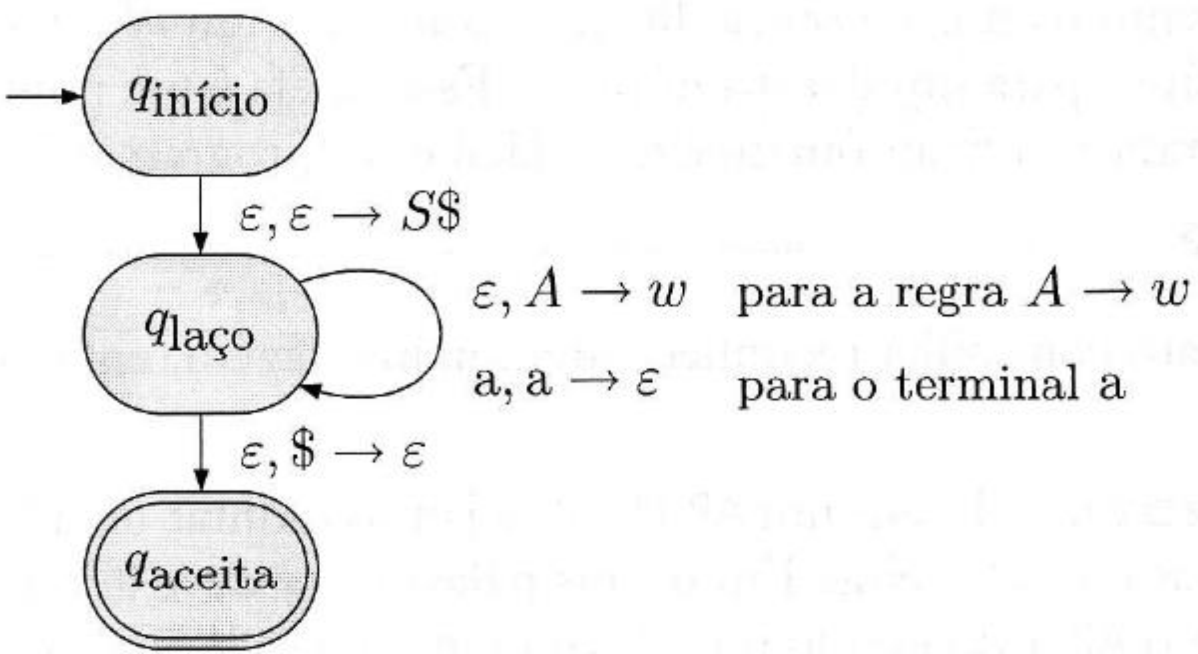
- Exemplo

$$S \to \mathtt{a}T\mathtt{b} \mid \mathtt{b}$$
$$T \to T\mathtt{a} \mid \varepsilon$$



Conversão GLC em APN (ideia)

- Caso Geral:



Equivalência entre APN e GLC

TEOREMA 2.20

Uma linguagem é livre-do-contexto se e somente se algum autômato com pilha a reconhece.

LEMA 2.21

Se uma linguagem é livre-do-contexto, então algum autômato com pilha a reconhece.

LEMA 2.27

Se um autômato com pilha reconhece alguma linguagem, então ela é livre-do-contexto.

Conversão APN em GLC (ideia)

- Para facilitar, vamos considerar que o APN possui as seguintes características:

1. Ele tem um único estado de aceitação, q_{aceita}.
2. Ele esvazia sua pilha antes de aceitar.
3. Cada transição ou empilha um símbolo (um movimento de *empilha*) ou desempilha um símbolo (um movimento de *desempilha*), mas não faz ambas as coisas ao mesmo tempo.

Conversão APN em GLC (ideia)

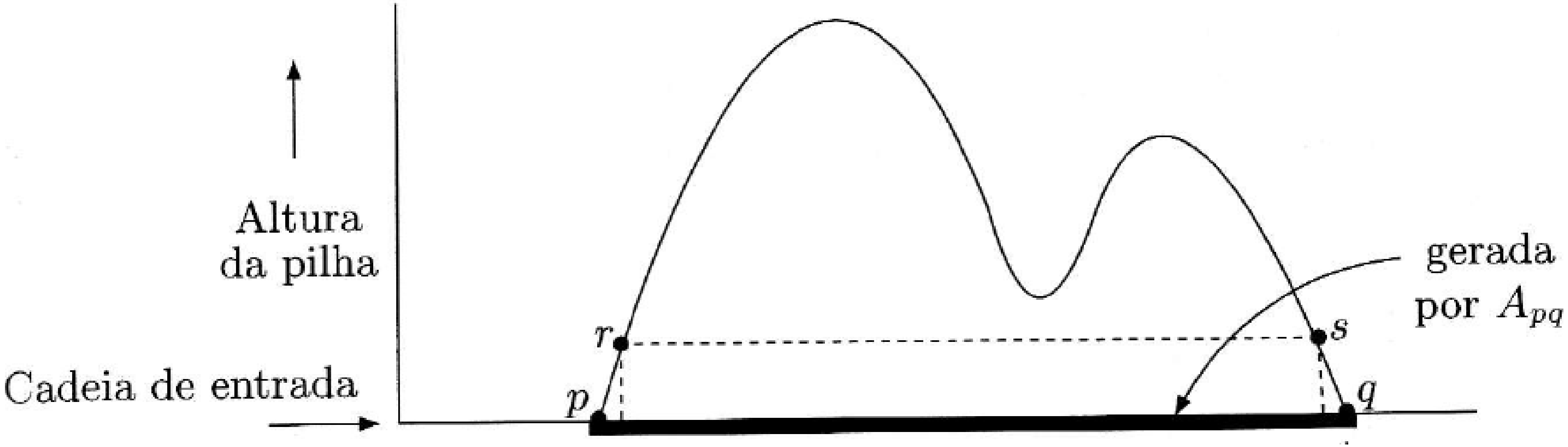
- G deve gerar uma cadeia x se x faz o APN ir do estado inicial ao estado de aceitação.
- Para cada par de estados (p, q), criamos uma variável A_{pq} que gere todas as cadeias x que levam o APN do estado p (com uma pilha vazia) ao estado q (com uma pilha vazia).
- Neste APN:
  - no estado p (com pilha vazia), o primeiro movimento é de EMPILHA.
  - O último movimento é de DESEMPILHA (chegando no estado q, com pilha vazia)

Conversão APN em GLC (ideia)

- No caminho de p a q (reconhecendo x), 2 situações:
  - A pilha só se torna vazia novamente quando chega em q
  - A pilha se torna vazia em algum ponto do caminho, antes de chegar em q

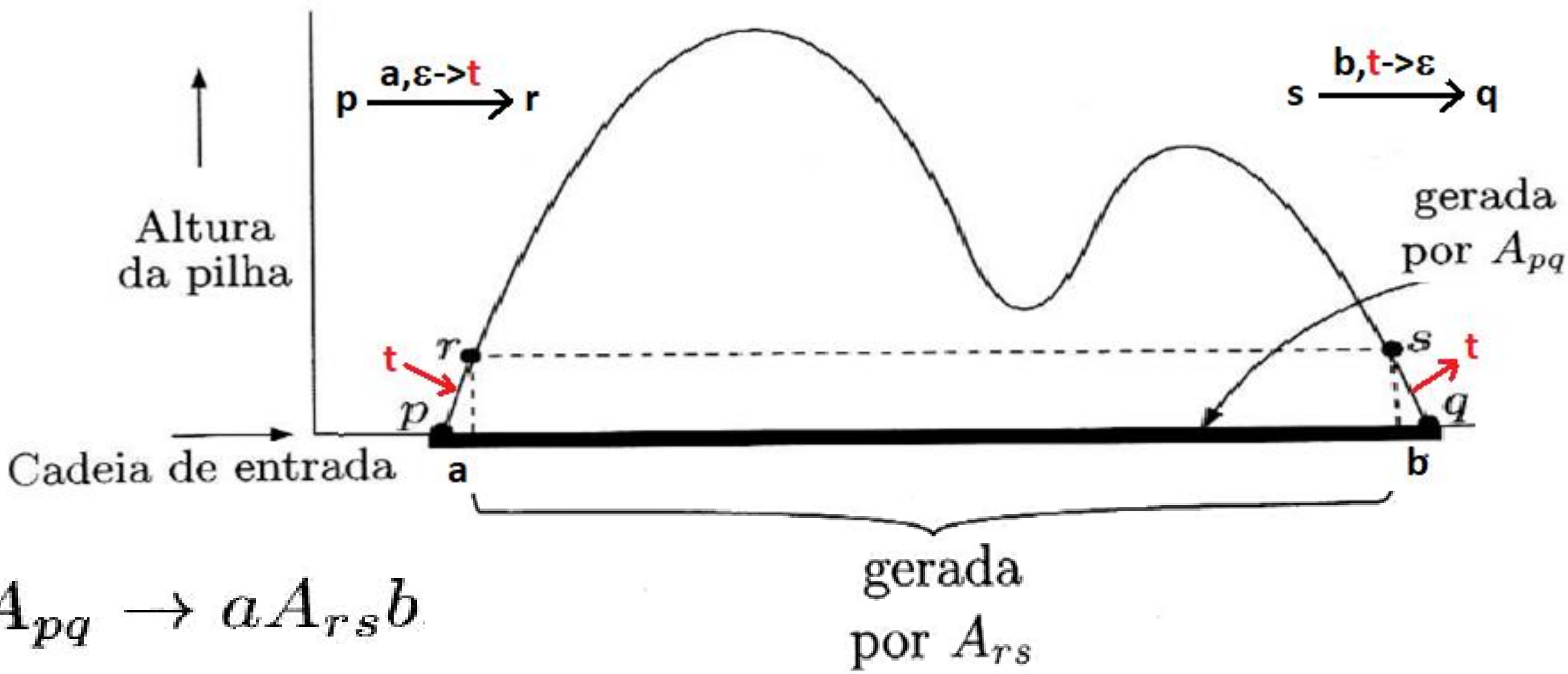
Conversão APN em GLC (ideia)

- No camilho de p a q (reconhecendo x), 2 situações:
  - A pilha só se torna vazia novamente quando chega em q
  - A pilha se torna vazia em algum ponto do caminho, antes de chegar em q



Conversão APN em GLC (ideia)

- No camilho de p a q (reconhecendo x), 2 situações:
  - A pilha só se torna vazia novamente quando chega em q
  - A pilha se torna vazia em algum ponto do caminho, antes de chegar em q



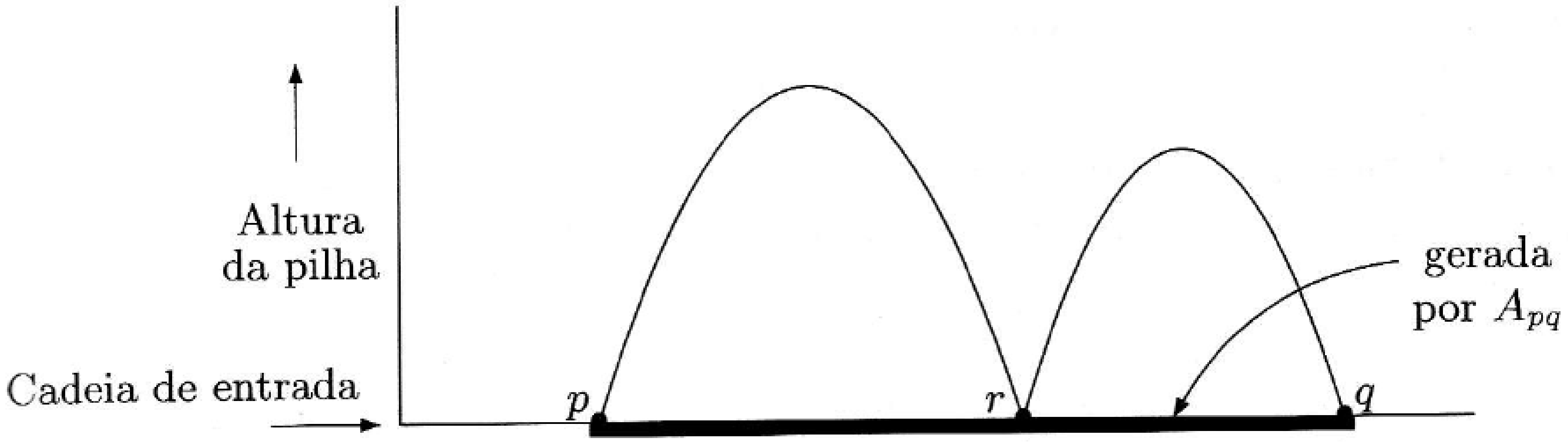
$$A_{pq} \to aA_{rs}b$$

Conversão APN em GLC (ideia)

- No camilho de p a q (reconhecendo x), 2 situações:
  - A pilha só se torna vazia novamente quando chega em q
  - A pilha se torna vazia em algum ponto do caminho, antes de chegar em q

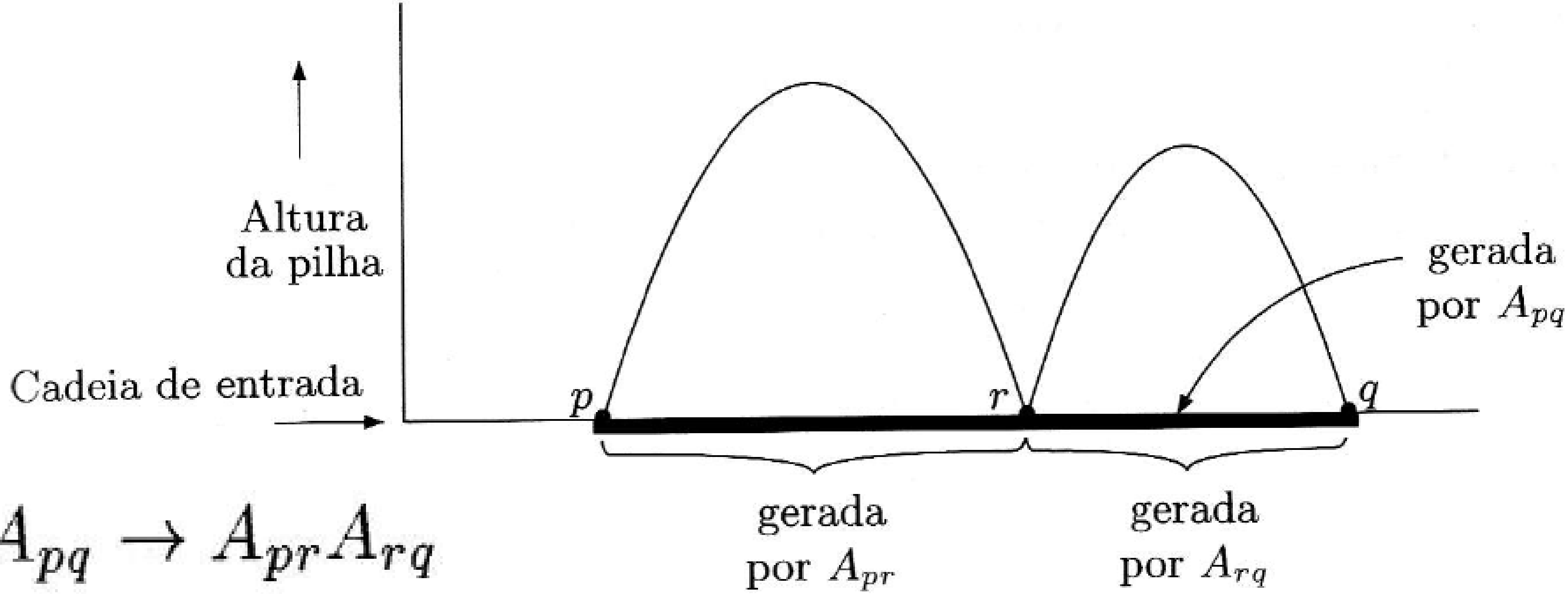
Conversão APN em GLC (ideia)

- No camilho de p a q (reconhecendo x), 2 situações:
  - A pilha só se torna vazia novamente quando chega em q
  - A pilha se torna vazia em algum ponto do caminho, antes de chegar em q



Conversão APN em GLC (ideia)

- No camilho de p a q (reconhecendo x), 2 situações:
  - A pilha só se torna vazia novamente quando chega em q
  - A pilha se torna vazia em algum ponto do caminho, antes de chegar em q



$$A_{pq} \rightarrow A_{pr}A_{rq}$$

Conversão APN em GLC (PROVA)

PROVA Digamos que $P = (Q, \Sigma, \Gamma, \delta, q_0, \{q_{\text{aceita}}\})$ e vamos construir G. As variáveis de G são $\{A_{pq} | \, p, q \in Q\}$. A variável inicial é $A_{q_0, q_{\text{aceita}}}$. Agora descrevemos as regras de G.

- Para cada $p, q, r, s \in Q$, $t \in \Gamma$ e $a, b \in \Sigma_\varepsilon$, se $\delta(p, a, \varepsilon)$ contém (r, t) e $\delta(s, b, t)$ contém (q, ε), ponha a regra $A_{pq} \to aA_{rs}b$ em G.
- Para cada $p, q, r \in Q$, ponha a regra $A_{pq} \to A_{pr}A_{rq}$ em G.
- Finalmente, para cada $p \in Q$, ponha a regra $A_{pp} \to \varepsilon$ em G.

Conversão APN em GLC (PROVA)

Temos que provar que essa construção funciona, ou seja, que A_{pq} gera x se e somente se x pode levar o APN de p (com pilha vazia) a q (com pilha vazia).

AFIRMATIVA 2.30 ....

Se A_{pq} gera x, então x pode levar P de p com pilha vazia a q com pilha vazia.

AFIRMATIVA 2.31 ....

Se x pode levar P de p com pilha vazia para q com pilha vazia, A_{pq} gera x.

Conversão APN em GLC (PROVA)

AFIRMATIVA 2.30

Se A_{pq} gera x, então x pode levar P de p com pilha vazia a q com pilha vazia.

Provamos essa afirmação por indução sobre o número de passos na derivação de x a partir de A_{pq}.

Base: A derivação tem 1 passo.
Uma derivação com um único passo tem de usar uma regra cujo lado direito não contém variáveis. As únicas regras em G onde nenhuma variável ocorre no lado direito são $A_{pp} \to \varepsilon$. Claramente, a entrada ε leva P de p com pilha vazia a p com pilha vazia e, portanto, a base está provada.

AFIRMATIVA 2.30

Se A_{pq} gera x, então x pode levar P de p com pilha vazia a q com pilha vazia.

Passo da Indução: Assuma verdadeiro para derivações de comprimento no máximo k, onde $k \geq 1$, e prove verdadeiro para derivações de comprimento $k+1$. Suponha que $A_{pq} \stackrel{*}{\Rightarrow} x$ com $k+1$ passos. O primeiro passo nessa derivação é ou $A_{pq} \Rightarrow aA_{rs}b$ ou $A_{pq} \Rightarrow A_{pr}A_{rq}$. Lidamos com esses dois casos separadamente.

No primeiro caso, considere a parte y de x que A_{rs} gera, de forma que $x = ayb$. Em razão do fato de que $A_{rs} \stackrel{*}{\Rightarrow} y$ com k passos, a hipótese da indução nos diz que P pode ir de r com pilha vazia para s com pilha vazia. Em razão do fato de que $A_{pq} \to aA_{rs}b$ é uma regra de G, $\delta(p, a, \varepsilon)$ contém (r, t) e $\delta(s, b, t)$ contém (q, ε), para algum símbolo de pilha t. Logo, se P começa em p com uma pilha vazia, após ler a ele pode ir para o estado r e empilhar t. Então, ler a cadeia y pode levá-lo a s e deixar t na pilha. E após ler b ele pode ir para o estado q e desempilhar t. Conseqüentemente, x pode levá-lo de p com pilha vazia para q com pilha vazia.

AFIRMATIVA 2.30

Se A_{pq} gera x, então x pode levar P de p com pilha vazia a q com pilha vazia.

Passo da Indução: Assuma verdadeiro para derivações de comprimento no máximo k, onde $k \geq 1$, e prove verdadeiro para derivações de comprimento $k+1$. Suponha que $A_{pq} \overset{*}{\Rightarrow} x$ com $k+1$ passos. O primeiro passo nessa derivação é ou $A_{pq} \Rightarrow aA_{rs}b$ ou $A_{pq} \Rightarrow A_{pr}A_{rq}$. Lidamos com esses dois casos separadamente.

No segundo caso, considere as partes y e z de x que A_{pr} e A_{rq}, respectivamente, geram, de forma que $x = yz$. Como $A_{pr} \overset{*}{\Rightarrow} y$ em no máximo k passos e $A_{rq} \overset{*}{\Rightarrow} z$ em no máximo k passos, a hipótese da indução nos diz que y pode levar P de p para r e z pode levar P de r para q, com pilha vazia no início e no final. Logo, x pode levá-lo de p com pilha vazia para q com pilha vazia. Isso completa o passo da indução.

Conversão APN em GLC (PROVA)

AFIRMATIVA 2.31

Se x pode levar P de p com pilha vazia para q com pilha vazia, A_{pq} gera x.

Provamos essa afirmação por indução sobre o número de passos na computação de P que vai de p para q com pilhas vazias sobre a entrada x.

Base: A computação tem 0 passos.

Se uma computação tem 0 passos, ela começa e termina no mesmo estado — digamos, p. Portanto, temos de mostrar que $A_{pp} \overset{*}{\Rightarrow} x$. Em 0 passos, P só tem tempo de ler a cadeia vazia, portanto $x = \varepsilon$. Por construção, G tem a regra $A_{pp} \to \varepsilon$; portanto, a base está provada.

AFIRMATIVA 2.31

Se x pode levar P de p com pilha vazia para q com pilha vazia, A_{pq} gera x.

Passo da Indução: Assuma verdadeiro para computações de comprimento no máximo k, onde $k \geq 0$, e prove verdadeiro para computações de comprimento $k+1$.
Suponha que P tenha uma computação na qual x leva de p para q com pilhas vazias em $k+1$ passos. Ou a pilha está vazia apenas no início e no final dessa computação, ou ela se torna vazia em algum outro ponto também.

AFIRMATIVA 2.31 ....

Se x pode levar P de p com pilha vazia para q com pilha vazia, A_{pq} gera x.

Passo da Indução: Assuma verdadeiro para computações de comprimento no máximo k, onde $k \geq 0$, e prove verdadeiro para computações de comprimento $k + 1$.

Suponha que P tenha uma computação na qual x leva de p para q com pilhas vazias em $k + 1$ passos. Ou a pilha está vazia apenas no início e no final dessa computação, ou ela se torna vazia em algum outro ponto também.

Caso 1:

- O primeiro símbolo a ser empilhado (t) deve ser o último a ser desempilhado
- Símbolos a e b lidos da entrada no primeiro e último movimento, respectivamente
  - Segundo estado r e penúltimo estado s
- δ(p,a,ε) contém (r,t) e δ(s,b,t) contém (q,ε), logo $A_{pq} \rightarrow aA_{rs}b$ está em G
- x = ayb. y faz o APN ir de r a s, sem tocar em t, em k-1 passos
- Pela hipótese de indução, A_{rs} =>* y. Logo, A_{pq} =>* x

AFIRMATIVA 2.31 ....

Se x pode levar P de p com pilha vazia para q com pilha vazia, A_{pq} gera x.

Passo da Indução: Assuma verdadeiro para computações de comprimento no máximo k, onde $k \geq 0$, e prove verdadeiro para computações de comprimento $k+1$.
Suponha que P tenha uma computação na qual x leva de p para q com pilhas vazias em $k+1$ passos. Ou a pilha está vazia apenas no início e no final dessa computação, ou ela se torna vazia em algum outro ponto também.

Caso 2:

- Estado r onde a pilha fica vazia no meio da computação de x
- Cada caminho (de p a r, e de r a q) tem no máximo k passos, x = yz
- Pela hipótese de indução: A_{pr} =>* y e A_{rq} =>* z.
- Como $A_{pq} \rightarrow A_{pr}A_{rq}$ está em G, A_{pq} =>* x

Equivalência entre APN e GLC

TEOREMA 2.20 ....

Uma linguagem é livre-do-contexto se e somente se algum autômato com pilha a reconhece.

LEMA 2.21 ....

Se uma linguagem é livre-do-contexto, então algum autômato com pilha a reconhece.

LEMA 2.27 ....

Se um autômato com pilha reconhece alguma linguagem, então ela é livre-do-contexto.