8. Autômatos com Pilha: lidando com as Gramáticas Livre de Contexto

8.1 Autômatos com Pilha (Apresentação informal)

Um autômato com pilha não determinístico tem os mesmos elementos que um autômato finito não determinístico, além de uma estrutura de dado de tipo pilha (LIFO) de capacidade ilimitada.

A cada passo de execução, o autômato finito consulta parte do topo da pilha e, eventualmente, troca seus símbolos por outra cadeia de símbolos.

8.1.2 Definição

Um autômato com pilha não determinístico é definido como um séptupla $M=(Q,\Sigma,\Gamma,\Delta,Z,S,F)$ onde :

- Q é um conjunto finito de estados discretos,

- Σ é um alfabeto de entrada,

- Γ é um alfabeto de pilha,

- $Z \in \Gamma$ é o símbolo inicial de pilha,

- $s \in Q$ é o estado inicial

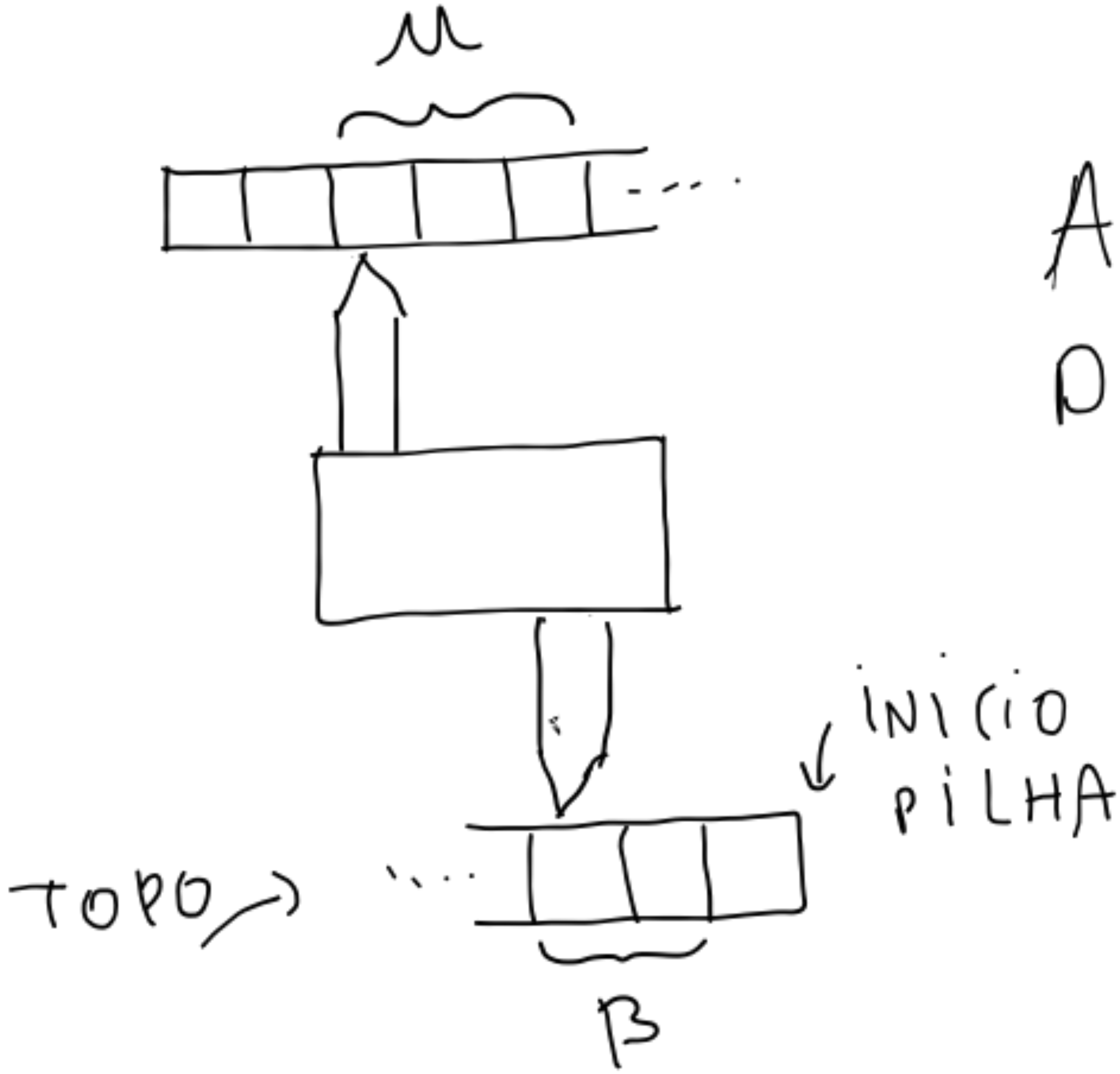
- $F \subseteq Q$ é o conjunto dos estados de aceitação,

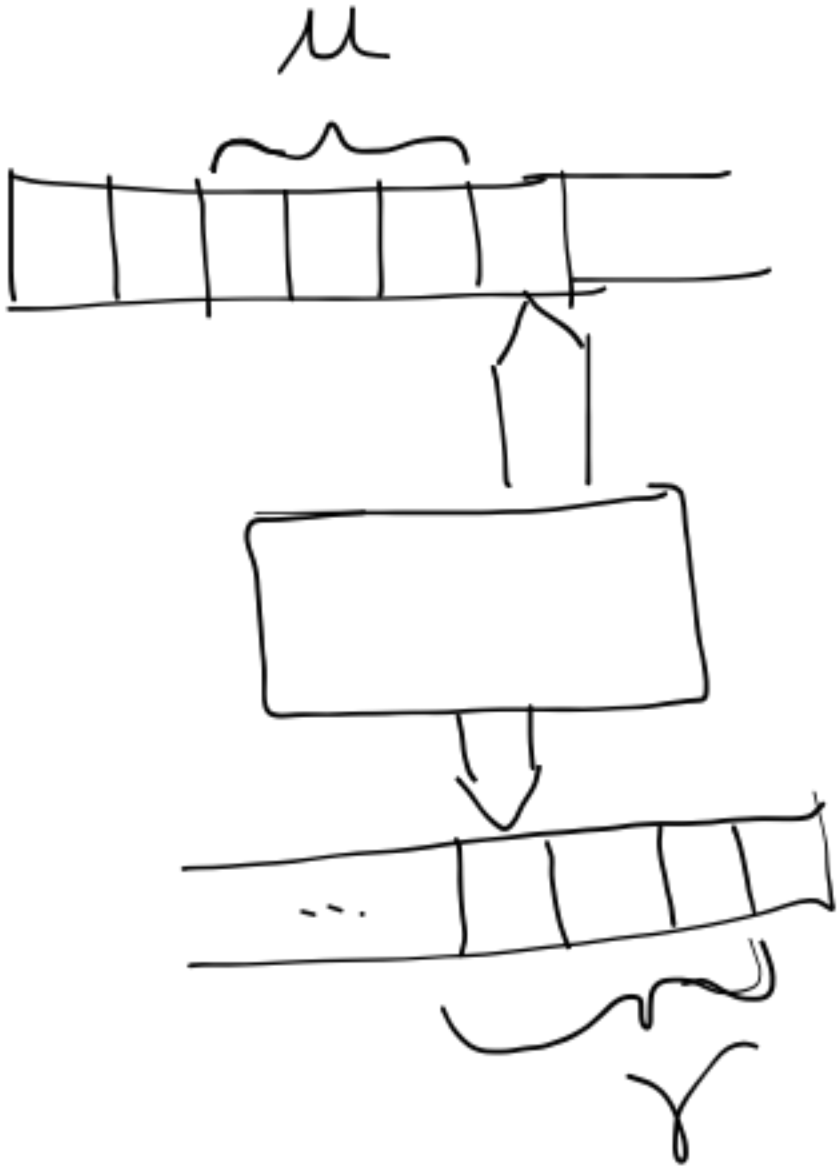
- $\Delta \subset ((Q \times \Sigma^* \times \Gamma^*) \times (Q \times \Gamma^*)$ é uma relação de transição

Uma transição $((p,u,\beta),(q,\Upsilon)) \in \Delta$ significa que o autômato pode passar do estado p para o estado q desde que o prefixo da palavra de entrada seja u e que a cadeia β se encontre no topo da pilha.

Depois do disparo, o autômato movimenta a cabeça para a direita, consumindo o prefixo u, além de substituir β pela cadeia Υ no topo da pilha.

ANTES DISPARO





DEPOIS
DISPARO

Uma configuração de um autômato não determinístico com pilha consiste de uma tripla composta pelo estado discreto corrente, pela parte remanescente da palavra de entrada a ser tratada e pelo conteúdo da pilha.

É um elemento de $Q \times \Sigma^* \times \Gamma^*$

Definição: a configuração (q',w',α') é derivável em uma etapa da configuração (q,w,α) pela máquina M se:

- $w=uw'$ (u pertencente a Σ^*é prefixo de w);
- $\alpha=\beta\delta$ (antes da transição, o topo da pilha - lido da esquerda para a direita- contém $\beta \in \Gamma^*$;
- $\alpha'=\Upsilon\delta$ (após a transição, o primeiro símbolo de Υ tornou-se o topo da pilha)

- $((q,u,\beta),(q', \Upsilon)) \in \Delta$.

NOTAÇÃO

$$(q, w, \alpha) \vdash_M (q', w', \alpha')$$

Definição: uma configuração C' é derivável em várias etapas pela maquina M, a partir da configuração C, se existem $k \geq 0$ e configurações intermediárias C0, C1, C2,...,Ck tais que:

- $C=C0$

- $C'=Ck$

- $Ci \vdash_M Ci+1$ para $0 \leq i < k$

NOTAÇÃO:

$$C \vdash_M^* C'$$

Definição: uma execução de um autômato finito não determinístico com pilha numa palavra w é uma seqüência de configurações:

$(s,w,Z) \mid\!- (q1,w1,\alpha 1) \mid\!- \ldots \mid\!- (qn,wn,\alpha n) \mid\!- \ldots$

onde Z representa o símbolo inicial da pilha.

Comportamentos possíveis da seqüência:

- Termina na configuração (p,ε,Υ), onde p é um estado de aceitação;

- Termina em uma configuração a partir da qual nenhuma transição é permitida;

- Produz uma seqüência infinita (sucessivas transição sobre a palavra vazia - possível no caso de autômatos não deterministicos).

Como o autômato é não determinístico, as execuções que aceitam a palavra de entrada são as execuções que chegam a um estado de aceitação após a leitura inteira da palavra de entrada.

Definição: uma palavra w é aceita pelo autômato não determinístico com pilha M se :

$(s,w,Z) \vdash^{*}_{M} (p,\varepsilon,\Upsilon)$ com $p \in F$

> Autômatos deste tipo são denominados autômatos aceitantes sobre estado de aceitação.

O símbolo inicial de pilha Z permite definir transições possíveis somente quando a pilha se encontra no estado inicial.

Não se pode simplesmente iniciar a execução com uma pilha vazia usando uma transição do tipo: $((q,u,\varepsilon),(q,\varepsilon))$, já que tal transição pode ser disparada mesmo sem que a pilha esteja vazia!

Definição alternativa de aceitação de palavras:
(s,w,Z) |-*M (p,ε,ε) , onde p não é, necessariamente, um estado de aceitação (autômatos denominados "aceitantes de palavra vazia")

- As definições de aceite sobre estado de aceitação ou sobre a pilha vazia são equivalentes;
- Conversão de um autômato M1 aceitante sobre um estado de aceitação em um M2 equivalente aceitante sobre pilha vazia: modifica-se M1 de modo que ele possa esvaziar sua pilha quando estiver em um estado de aceitação. O inverso é feito acrescentando-se em M2 um novo estado (de aceitação) que só pode ser atingido quando a pilha estiver vazia.

VER ARQUIVOS EXEMPLOS-1

8.2 As linguagens livre de contexto

8.2.1 Definição

Uma linguagem é livre de contexto se existe uma gramática livre de contexto que produz as palavras da linguagem.

Produções da forma: $A \rightarrow \beta$ com $A \in (V - \Sigma)$ e onde não existem restrições sobre β.

Ex: As palavras da linguagem a(n)b(n) com $n \geq 0$ são produzidas pelas regras de produção:

(1) $S \rightarrow aSb$
(2) $S \rightarrow \varepsilon$

Tal linguagem é então livre de contexto

Ex: A linguagem das palavras da forma ww(R) é produzida por uma gramática cujas regras são:

(1) $S \rightarrow aSa$
(2) $S \rightarrow bSb$
(3) $S \rightarrow \varepsilon$

Tal linguagem é então livre de contexto

Exemplo das linguagens que possuem duplo balanceamento como os blocos bem estruturados das linguagens de programação BEGIN(n)END(n) e as linguagens com parênteses balanceados ((n))(n)

Exercício: encontrar uma gramática livre de contexto que produz expressões aritméticas contendo colchetes balanceados, dois operadores e um operando: exemplo→ [x+x]*x

VER ARQUIVO EXEMPLOS-2

8.2.2 Relação com os autômatos com pilhas

Teorema: uma linguagem é livre de contexto se, e somente se, é aceita por um autômato com pilha.

- Será apresentado a seguir um método que produz o autômato não determinístico com pilha correspondente a uma dada gramática livre de contexto.

Seja uma linguagem livre de contexto L gerada pela gramática $G=(V,\Sigma,R,S)$. Então, o seguinte autômato com pilha M aceita L:

$M=(Q,\Sigma,\Gamma,\Delta,Z,s,F)$, onde:

- $Q=\{p,q,r\}$
- $\Gamma=V \cup \{Z\}$
- $Z \notin V$
- $s=p$
- $F=\{r\}$
- Δ: conjunto formado por:
  - $((p,\varepsilon,\varepsilon),(q,S))$;
  - $((q,\varepsilon,A),(q,x))$, para cada $A \rightarrow x \in R$;
  - $((q,a,a),(q,\varepsilon))$, para cada $a \in \Sigma$;
  - $((q,\varepsilon,Z),(r,\varepsilon))$.

OBS: Note que as regras do autômato devem estar aptas a reconhecer todas as palavras geradas pela gramática que lhe corresponde. Para tanto, elas devem ser tais que possibilitem empilhar e desempilhar todas as expressões da gramática envolvidas na construção de cada uma dessas palavras, de modo que, ao término do desempilhamento, a pilha esteja vazia (note que, de fato, há regras para desempilhar cada símbolo de Γ) e o autômato esteja em um estado de aceitação.

Exercício: construir o autômato com pilha de 3 estados que aceita as palavras a(n)b(n) para n≥0 e produzir a seqüência de configurações para a(2)b(2), a(2)b(3), a(1)b(2)

Exercício: construir o autômato com pilha de 3 estados que aceita as palavras ww(R) e produzir a seqüência de configurações para abba e abb.

VER ARQUIVO EXEMPLOS-3D

Observações: pode-se observar que cada execução do autômato M corresponde a uma derivação da gramática.

Além disso, cada derivação da gramática corresponde a uma execução do autômato. Isso permanece verdadeiro se a cada etapa da derivação é o não terminal à esquerda que é trocado pela parte direita da regra de produção aplicada.

Tal derivação é chamada de derivação à esquerda.

De fato, é possível mostrar que toda palavra de uma gramática livre de contexto pode ser produzida por uma derivação à esquerda, já que é sempre possível permutar as etapas de derivação de uma gramática livre de contexto.

8.2.3 Propriedades das linguagens livre de contexto

Sejam L1 e L2 duas linguagens livre de contexto.

Propriedade 1: a linguagem L1 ∪ L2 é também livre de contexto.

Prova: se G1=(V1,Σ1,R1,S1) e G2=(V2,Σ2,R2,S2) são as gramáticas que produzem as palavras de L1 e L2.

Então a linguagem L1 ∪ L2 é produzida pela gramática livre de contexto G=(V,Σ,R,S) onde:

- $V = V1 \cup V2 \cup \{S\}$ (os não terminais de V1 e V2 estão distintos e S é um novo símbolo)

- $\Sigma = \Sigma1 \cup \Sigma2$

- S é o novo símbolo inicial

- $R = R1 \cup R2 \cup \{S \to S1, S \to S2\}$

Propriedade 2: a linguagem L1.L2 é também livre de contexto.

Prova: se $G1=(V1,\Sigma 1,R1,S1)$ e $G2=(V2,\Sigma 2,R2,S2)$ são as gramáticas que produzem as palavras de L1 e L2, uma gramática livre de contexto que produz a linguagem L1.L2 é $G=(V,\Sigma,R,S)$ onde...

Terminar a prova!

Propriedade 3: a linguagem L1* é também livre de contexto.

Prova: uma gramática livre de contexto que produz a linguagem L1* é $G=(V1,\Sigma 1,R,S1)$ onde R=...

Terminar a prova!