Autômato de Pilha e sua Equivalência com GLC

Prof^a Jerusa Marchi

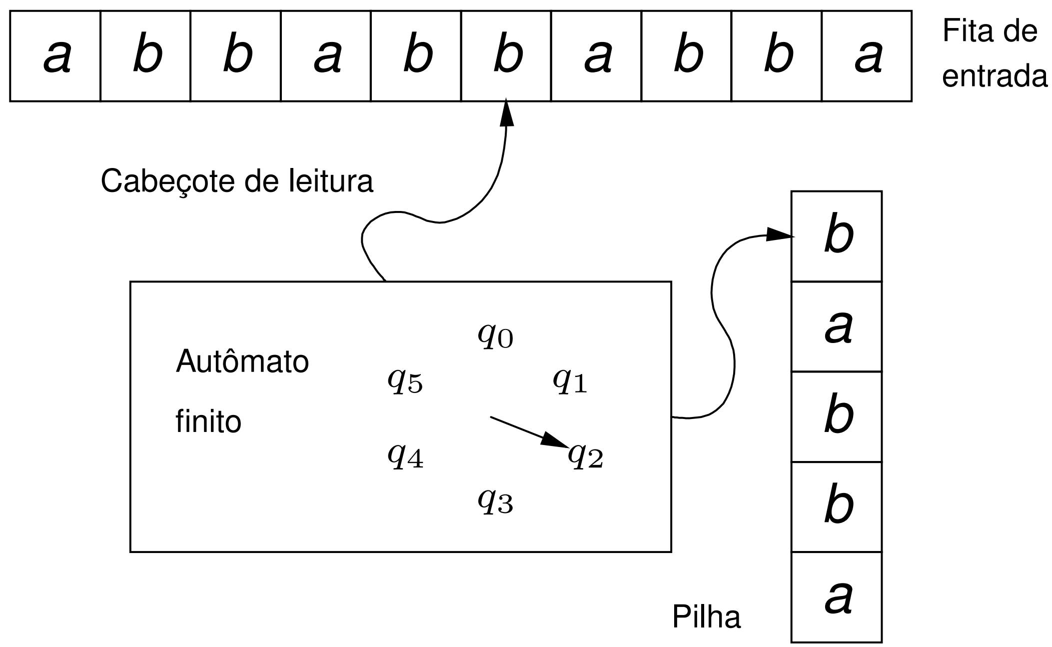
Departamento de Informática e Estatística

Universidade Federal de Santa Catarina

e-mail: jerusa.marchi@ufsc.br

Autômatos de Pilha

- Linguagens Livres de Contexto são reconhecidas por Autômatos de Pilha
  - Autômato finito não determinístico com ε-transições com uma memória auxiliar (*pilha*)
  - A manipulação (ler, push, pop) é permitida somente no topo da pilha (LIFO)



Autômatos de Pilha

- AP podem ser *determinísticos*:
  - Aceitam todas as linguagens regulares
  - Aceitam um subconjunto próprio das Linguagens Livres de Contexto

Equivalência AP - GLC

Teorema Uma Linguagem é Livre de Contexto se e somente se algum Autômato de Pilha a reconhece.

- Idéia da Prova:
  - ($\rightarrow$) Demonstrar como um Autômato de Pilha pode usar a GLC para reconhecer a palavra de entrada
  - ($\leftarrow$) Demonstrar como um AP pode produzir uma GLC que gera a linguaguem

GLC $\Rightarrow$ AP

Lema 1: Se uma linguagem é livre de contexto, então algum autômato de pilha a reconhece.

Ideia da Prova: Seja A uma LLC. A é gerada por uma GLC G. A idéia é construir um AP P que aceita a entrada w se G gera essa entrada, determinando se existe uma derivação para w. O processo de derivação gera cadeias intermediárias que são formas sentenciais. Projetamos P para determinar se alguma série de substituições, usando as regras de G pode levar da variável inicial à w. O AP P é não determinístico, o que assegura que todas as possíveis derivações serão testadas.

GLC $\Rightarrow$ AP

P opera da seguinte forma:

1. Coloque o símbolo marcador $\$$ e a variável inicial de G na pilha
2. Repita
   (a) Se o topo da pilha é o símbolo de variável A, não deterministicamente selecione uma das regras de G para A e substitua A pelo corpo da produção.
   (b) Se o topo da pilha é o símbolo terminal a, verifique se o símbolo da entrada é a. Se for, desempilhe, avance na entrada e repita. Se não, rejeite esse ramo de Computação.
   (c) Se o topo da pilha é o símbolo $\$$, então entre no estado de aceitação. A entrada é aceita se tiver sido totalmente consumida.

GLC $\Leftarrow$ AP

Lema 2: Se um Autômato de pilha reconhece alguma linguagem, então ela é livre de contexto.

Ideia da Prova: Temos um AP P e desejamos montar uma GLC G que gere todas as cadeias que P aceita. Para isto, para cada par de estados p e q em P, a gramática terá uma variável A_{pq}. Essa variável gera todas as cadeias que podem levar P de p com pilha vazia a q com pilha vazia, ou ainda mantendo a pilha inalterada (o que havia no estado p permanece em q).

Para tal modifica-se P para que possua as seguintes características:

1. Ele tem um único estado de aceitação, q_f
2. Ele esvazia a pilha antes de aceitar.
3. Cada transição ou empilha um símbolo ou desempilha um símbolo, mas não ambos ao mesmo tempo.

GLC $\Leftarrow$ AP

- Dar a P as características 1 e 2 é fácil. Já para a característica 3, substituímos cada transição que simultaneamente desempilha e empilha, por uma sequência de duas transições que passa por um novo estado e substituímos as transições que não desempilham, nem empilham por uma sequência de duas transições, uma que empilha um símbolo arbitrário e outra que o desempilha.

GLC $\Leftarrow$ AP

- Para que A_{pq} gere todas as cadeias que levam P de p a q iniciando e terminando com pilha vazia, precisamos observar que há duas possíbilidades: Ou o símbolo desempilhado no final é o que foi empilhado no inicio, ou não.
- Se for, a pilha está vazia somente no inicio e no final da computação de P sobre w. Essa primeira possibilidade é simulada pela regra $A_{pq} \rightarrow aA_{rs}b$ onde a é a entrada lida no primeiro movimento, b é a entrada lida no último movimento, r é o estado seguinte a p e s é o estado anterior a q.
- A segunda possibilidade é simulada pela regra $A_{pq} \rightarrow A_{pr}A_{rq}$, onde r é o estado no qual a pilha fica vazia.

GLC $\Leftarrow$ AP

- Para construir G então são criados símbolos não-terminais A_{pq} para todo $p, q \in K$, a variável inicial será $A_{q_0 q_a}$. Para cada p, q, r, $s \in K$, $t \in \Gamma$ 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 $p, q, r \in K$, ponha a regra $A_{pq} \to A_{pr}A_{rq}$ em G.
- Para cada $p \in K$, ponha a regra $A_{pp} \to \varepsilon$ em G.