MAC 4722 - Linguagens, Autômatos e Computabilidade

Rodrigo Augusto Dias Faria - NUSP 9374992
Departamento de Ciência da Computação - IME/USP

10 de maio de 2016

Lista 4

L4.1 Complete a demonstração formal do lema 2.21, a primeira parte do teorema 2.20. A saber, primeiro demonstre que, para toda palavra w derivada pela gramática A, uma computação que aceite a palavra w no autômato construído P pode conduzir do estado q_{inicio} para o estado q_{aceita}. Em seguida, demonstre que toda palavra w aceita por uma computação de P admite uma derivação pela gramática A.

Resposta: TODO

L4.2 (Sipser 2.9) Dê uma gramatica livre-do-contexto que gere a linguagem

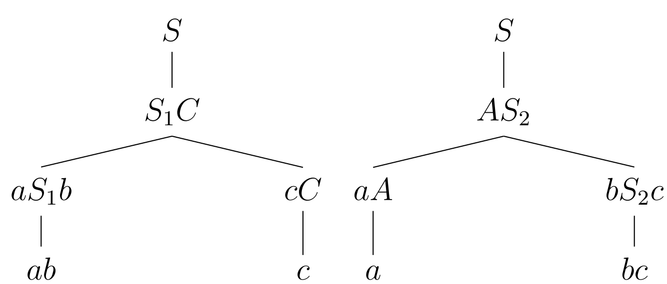
$$A = \{a^i b^j c^k \mid i = j \text{ ou } j = k \text{ onde } i, j, k \geq 0\}$$

Resposta: A GLC que gera a linguagem A é $G = (\{S, S_1, S_2, A, C\}, \{a, b, c\}, R, S)$, onde S é a variável inicial e R é o conjunto de regras:

$$\begin{aligned}
S &\to AS_2 \mid S_1C \\
S_1 &\to aS_1b \mid \epsilon \\
S_2 &\to bS_2c \mid \epsilon \\
A &\to aA \mid \epsilon \\
C &\to cC \mid \epsilon
\end{aligned}$$

Sua gramática é ambígua? Por que ou por que não?

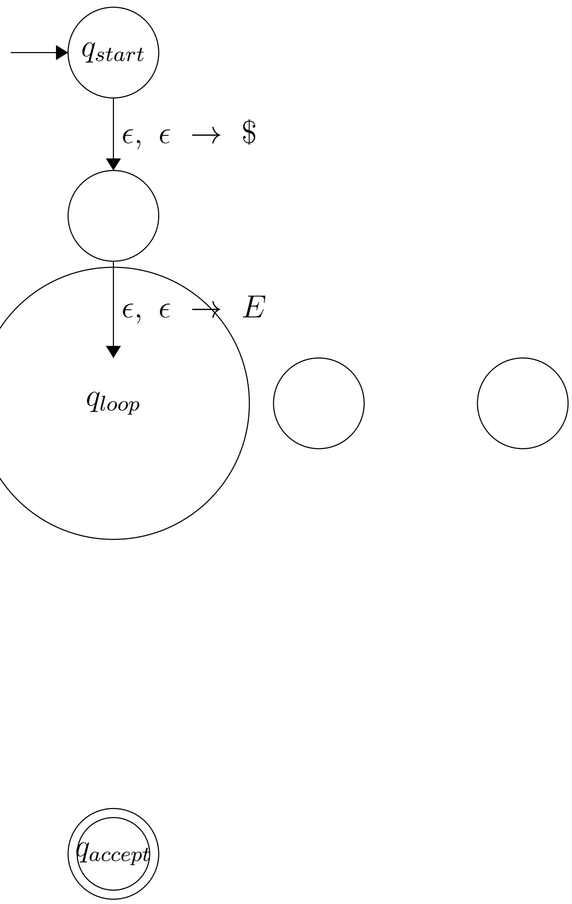
Sim, ela é ambígua, pois G gera uma mesma cadeia, digamos w, ambiguamente, ou seja, w tem duas árvores sintáticas distintas. A derivação da cadeia $w = abc$, por exemplo, produz duas árvores sintáticas diferentes.



L4.3 (Sipser 2.11) Converta a GLC G_4 do exercício 2.1 para um AP equivalente, usando o teorema 2.20.

$$E \to E + T \mid T$$
$$T \to T \times F \mid F$$
$$F \to (E) \mid a$$

Resposta: TODO



L4.4 (Sipser 2.14) Converta a seguinte GLC numa GLC equivalente na forma normal de Chomsky, usando o procedimento dado no Teorema 2.9.

$$A \to BAB \mid B \mid \epsilon$$
$$B \to 00 \mid \epsilon$$

Resposta: Seguem os passos de acordo com o teorema.

1. Nova variável inicial

$$S_0 \to A$$
$$A \to BAB \mid B \mid \epsilon$$
$$B \to 00 \mid \epsilon$$

2. Removendo a regra $A \to \epsilon$

$$\begin{aligned} S_0 &\to A \mid \epsilon \\ A &\to BAB \mid B \mid BB \\ B &\to 00 \mid \epsilon \end{aligned}$$

3. Removendo a regra $B \to \epsilon$

$$\begin{aligned} S_0 &\to A \mid \epsilon \\ A &\to BAB \mid B \mid BB \mid AB \mid BA \\ B &\to 00 \end{aligned}$$

4. Removendo a regra unitária $A \to B$

$$\begin{aligned} S_0 &\to A \mid \epsilon \\ A &\to BAB \mid 00 \mid BB \mid AB \mid BA \\ B &\to 00 \end{aligned}$$

5. Removendo a regra unitária $S_0 \to a$

$$\begin{aligned} S_0 &\to BAB \mid 00 \mid BB \mid AB \mid BA \mid \epsilon \\ A &\to BAB \mid 00 \mid BB \mid AB \mid BA \\ B &\to 00 \end{aligned}$$

6. Simplificando, tomando $X \to AB$ e $Y \to 0$

$$\begin{aligned} S_0 &\to BX \mid YY \mid BB \mid AB \mid BA \mid \epsilon \\ A &\to BX \mid YY \mid BB \mid AB \mid BA \\ B &\to YY \\ X &\to AB \\ Y &\to 0 \end{aligned}$$

L4.5 (Sipser 2.25) Para qualquer linguagem A, seja $SUFIXO(A) = \{v \mid uv \in A$ para alguma cadeia $u\}$. Mostre que a classe de linguagens livres-do-contexto é fechada sob a operação $SUFIXO$.

Resposta: Seja A uma linguagem livre de contexto. Existe um autômato a pilha (AP) $P = (Q_1, \Sigma, \Gamma, \delta_1, q_1, F_1)$ que aceita as cadeias de A. Vamos construir um novo autômato a pilha $M = (Q, \Sigma, \Gamma, \delta, q, F)$ que reconhece a linguagem $SUFIXO(A)$. Para tanto, precisamos, inicialmente, criar uma cópia de P, digamos, $P' = (Q_{1'}, \Sigma, \Gamma, \delta_{1'}, q_{1'}, F_{1'})$. Devemos, também, alterar as entradas das transições de δ_1 para vazio, ou seja, para cada transição $a, b \to c$ de δ_1, teremos $\epsilon, b \to c$.

Logo, podemos escrever M formalmente como:

1. $Q = Q_1 \cup Q_{1'}$,
2. Σ é o alfabeto de entrada,
3. Γ é o alfabeto de pilha,
4. $\delta = \delta_1 \cup \delta_{1'}$ e $(q_{i'}, \epsilon) \in \delta(q_i, \epsilon, \epsilon)$ para todo i, tal que $q_i \in Q_1$ e $q_{i'} \in Q_{1'}$,
5. $q_0 = q_1$
6. $F = F_{1'}$

Demonstração. Para demonstrar que a construção está correta, devemos provar que $\forall w \in \Sigma^*$, $w \in SUFIXO(A) \iff w \in L(M)$, onde $L(M)$ é a linguagem do autômato M.

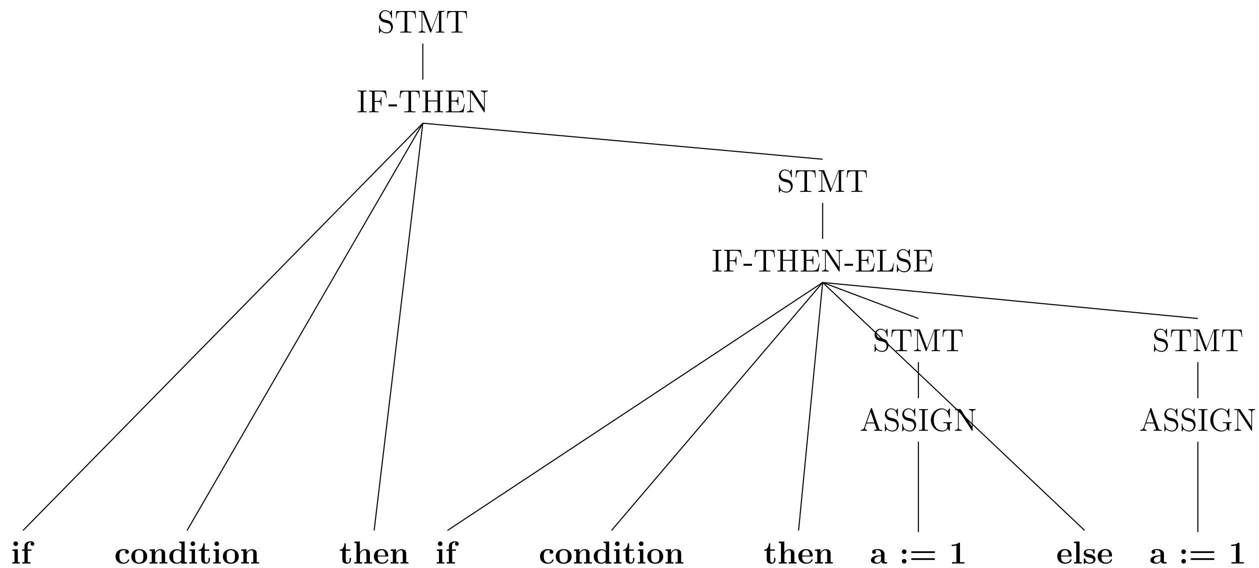
$\Rightarrow$ Se $w \in SUFIXO(A)$, então $w \in L(M)$ Seja $v = x_1x_2 \dots x_n$ uma cadeia em $SUFIXO(A)$. Logo, existe uma cadeia $uv \in A$ e um passeio $X = q_1q_2 \dots q_m$ em P (autômato que reconhece as cadeias de A) tal que q_1 é o estado inicial de P e q_m é um estado final de P. Seja $X' = q_jq_{j+1} \dots q_m$ o trecho de X que consome v. No autômato M alteramos as transições de A para que estas não consumam a cadeia, mas atualizem apropriadamente a pilha. Logo, ao ler a cadeia v, o autômato M simulará o passeio de $q_1q_2 \dots q_j$ sem consumir a cadeia e, então, através de uma transição ϵ irá para o estado $q_{j'}$ de A que consumirá e percorrerá o passeio $q_{j'}q_{j+1'} \dots q_{m'}$ equivalente ao passeio $X' = q_jq_{j+1'} \dots q_m$, levando a $q_{m'}$ que é um estado final de M.

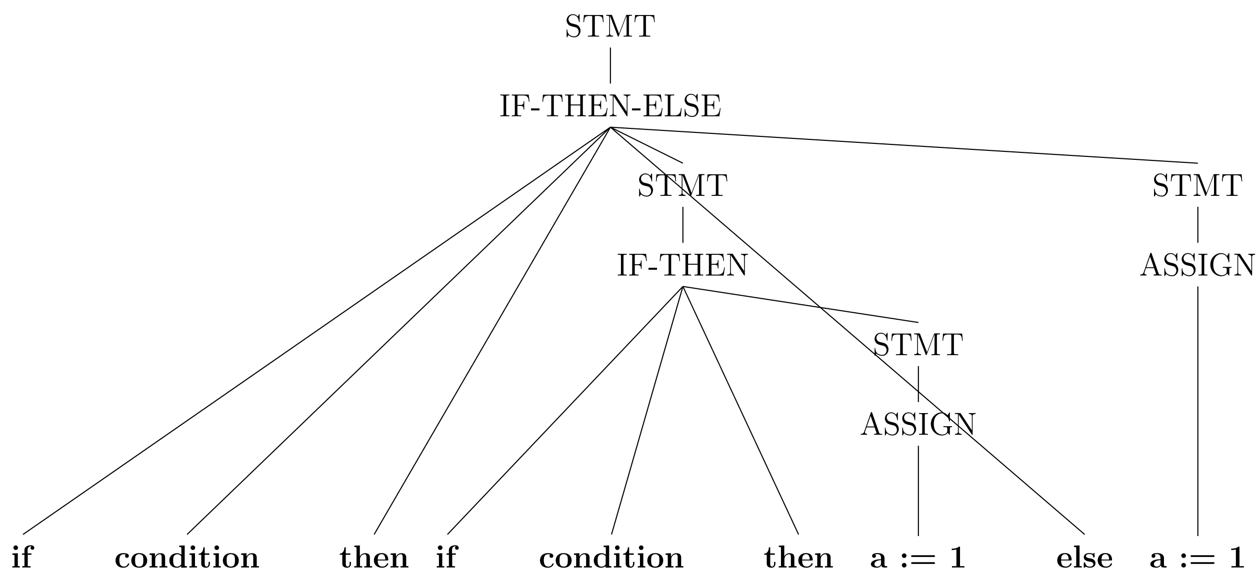
$\Leftarrow$ Se $w \in L(M)$, então $w \in SUFIXO(A)$ Seja $w = x_1x_2 \dots x_n$ uma cadeia reconhecida pelo autômato M. Logo, existe um passeio $X = q_1q_2 \dots q_m$ que reconhece w em M, tal que q_1 é o estado inicial de P e q_m é um estado final de P'. A porção de M proveniente de P não consome a cadeia de entrada e, portanto, ela é consumida na porção de M proveniente de P'. Para que w esteja em $SUFIXO(A)$, precisamos encontrar uma cadeia u tal que $uw \in A$. to be completed ■

L4.6 Sipser(2.27)

a. Mostre que G é uma gramática ambígua.

Resposta: Podemos mostrar que G é ambígua quando em uma derivação ela produz duas árvores sintáticas distintas para uma mesma cadeia. Seja $w =$ **if condition then if condition then a := 1 else a := 1**.


STMT
IF-THEN
STMT
IF-THEN-ELSE
STMT
STMT
ASSIGN
ASSIGN
if
condition
then
if
condition
then
a := 1
else
a := 1


STMT
IF-THEN-ELSE
STMT
STMT
IF-THEN
ASSIGN
STMT
ASSIGN
if
condition
then
if
condition
then
a := 1
else
a := 1