Automata and Grammars

SS 2018

Assignment 11: Solutions to Selected Problems

Problem 11.1. [Emptiness and Finiteness]

Determine the cardinality of the language $L(G_i)$ for the following context-free grammars G_i $(i = 1, 2, 3)$:

(a) $G_1 = (\{S, A, B, C, D\}, \{a, b\}, P_1, S)$, where P_1 is defined as follows:

$$\begin{array}{rcl} P_1 & = & \{S \to aS, S \to AB, S \to CD, A \to aDb, A \to AD, A \to BC, \\ & & B \to bSb, B \to BB, C \to BA, C \to ASb, D \to ABCD, D \to \varepsilon\}. \end{array}$$

(b) $G_2 = (\{S, A, B, C, D\}, \{a, b\}, P_2, S)$, where P_2 is defined as follows:

$$\begin{array}{rcl} P_2 & = & \{S \to AB, S \to BC, S \to CD, A \to BC, A \to BD, \\ & & B \to BC, B \to DD, B \to b, C \to AC, C \to BC, D \to a\}. \end{array}$$

(c) $G_3 = (\{S, A, B, C, D\}, \{a, b\}, P_3, S)$, where P_3 is defined as follows:

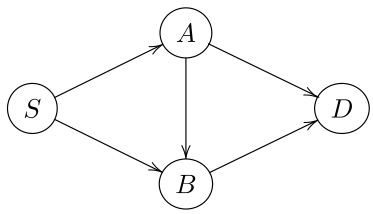
$$\begin{array}{rcl} P_3 & = & \{S \to AB, S \to BC, S \to CD, A \to BC, A \to BD, \\ & & B \to CC, B \to DD, B \to b, C \to AS, D \to AC, D \to a\}. \end{array}$$

Solution. (a) $V_{\text{term}} = \{ X \in N \mid L(G_1, X) \neq \emptyset \} = \{D, A\}$, and hence, $L(G_1) = \emptyset$.

(b) $V_{\text{term}} = \{D, B, A, S\}$. As $S \in V_{\text{term}}$, we see that $L(G_2) \neq \emptyset$. Further, $V_{\text{reach}} = \{S, A, B, D\}$, that is, the nonterminal C is useless. Thus, we can delete it together with all productions that contain it, which yields the grammar $G_2' = (\{S, A, B, D\}, \{a, b\}, P_2', S)$, where P_2' is defined as follows:

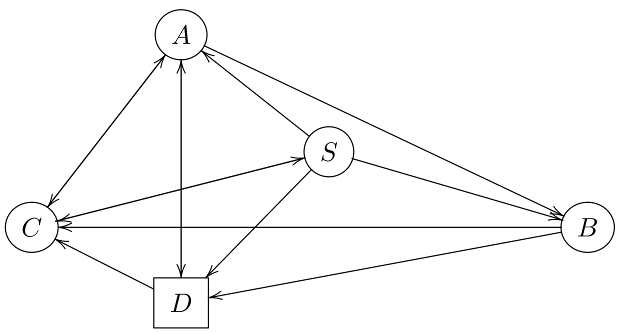
$$P_2' = \{S \to AB, A \to BD, B \to DD, B \to b, D \to a\}.$$

From G_2' we obtain the following graph:



As this graph does not contain any cycle, we see that $L(G_2') = L(G_2)$ is finite. In fact, $L(G_2) = \{bab, a^3b, ba^3, a^5\}$, that is, it has cardinality 4.

(c) $V_{\text{term}} = \{D, B, A, S, C\}$, that is, $L(G_3) \neq \emptyset$. As $V_{\text{reach}} = \{S, A, B, C, D\}$, we see that G_3 is a proper grammar without ε-rules that is in CNF. From G_3 we obtain the following graph:



As this graph contains cycles, e.g., $S \to D \to C \to S$, we see that $L(G_3)$ is infinite. □

Problem 11.2. [CKY-Algorithm]

Apply the CYK-algorithm from the proof of Theorem 3.30 to the following context-free grammar

$$G = (\{A, B, C, R, S, T, U, V, X, Y, Z\}, \{a, b, c\}, P, S)$$

and the input words $w_1 = bcaaacbb$ and $w_2 = abcbaabc$, where P is defined as follows:

$$\begin{aligned} P_1 \;=\; & \{S \to TU, T \to BA, T \to BX, T \to BY, X \to TA, \\ & Y \to CA, Y \to RA, U \to AB, U \to AV, U \to AZ, \\ & Z \to UB, V \to RB, R \to CR, R \to c, A \to a, B \to b, C \to c\}. \end{aligned}$$

Solution. $w_{1,1} = bcaaacbb$:

x/j	b	c	a	a	a	c	b	b
1	B	C, R	A	A	A	C, R	B	B
2	–	Y	–	–	–	V	–	–
3	T	–	–	–	U	–	–	–
4	X	–	–	–	Z	–	–	–
5	–	–	–	U	–	–	–	–
6	–	–	–	–	–	–	–	–
7	–	–	–	–	–	–	–	–
8	S	–	–	–	–	–	–	–

As $S \in V_{1,8}$, we see that $w_{1,1} \in L(G_1)$. We can even reconstruct a derivation for $w_{1,1}$ from the table above:

$$\begin{aligned} S & \to_{P_1} TU && \to_{P_1} BYU && \to_{P_1} bYU && \to_{P_1} bRAU \\ & \to_{P_1} bcAU && \to_{P_1} bcaU && \to_{P_1} bcaAZ && \to_{P_1} bcaaZ \\ & \to_{P_1} bcaaUB && \to_{P_1} bcaaAVB && \to_{P_1} bcaaaVB && \to_{P_1} bcaaaRBB \\ & \to^3_{P_1} bcaaacbb. \end{aligned}$$

$w_{1,2} = abcbaabc$:

x/j	a	b	c	b	a	a	b	c
1	A	B	C, R	B	A	A	B	C, R
2	U	–	V	T	–	U	–	–
3	–	–	–	X	–	–	–	–
4	–	–	–	S	–	–	–	–
5	–	–	–	–	–	–	–	–
6	–	–	–	–	–	–	–	–
7	–	–	–	–	–	–	–	–
8	–	–	–	–	–	–	–	–

As $S \notin V_{1,8}$, we see that $w_{1,2} \notin L(G_1)$. □

Problem 11.3. [DPDA]

In Theorem 3.17 we have seen that, for each PDA M_1, there exists a PDA M_2 such that $N(M_2) = L(M_1)$, while for deterministic PDAs, a corresponding result does not hold. Why doesn't the proof of Theorem 3.17 carry over to DPDAs?

Solution. Let $M_1 = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be a PDA, and let $L = L(M_1)$, that is,

$$L = \{\, w \in \Sigma^* \mid (q_0, Z_0, w) \vdash_{M_1}^{*} (p, \gamma, \varepsilon) \text{ for some } p \in F \text{ and } \gamma \in \Gamma^* \,\}.$$

The PDA M_2 that simulates the PDA M_1 step by step solves the following two problems:

1. M_2 must be able to recognize when M_1 empties its pushdown without being in a final state, as in this situation, M_2 must not accept.
2. M_2 must empty its pushdown when M_1 accepts.

Accordingly, M_2 is defined as

$$M_2 = (Q \cup \{q_\ell, q_0'\}, \Sigma, \Gamma \cup \{X_0\}, \delta', q_0', X_0, \emptyset),$$

where the transition relation δ' is defined as follows:

(1) $\delta'(q_0', \varepsilon, X_0) = \{(q_0, X_0 Z_0)\}$,

(2) $\delta'(q, a, Z) \supseteq \delta(q, a, Z)$ for all $q \in Q$, $a \in \Sigma \cup \{\varepsilon\}$, and $Z \in \Gamma$,

(3) $\delta'(q, \varepsilon, Z) \ni (q_\ell, Z)$ for all $q \in F$ and $Z \in \Gamma \cup \{X_0\}$,

(4) $\delta'(q_\ell, \varepsilon, Z) = \{(q_\ell, \varepsilon)\}$ for all $Z \in \Gamma \cup \{X_0\}$.

- By (1) M_2 enters the initial configuration of M_1 with the symbol X_0 below the bottom marker of M_1.
- By (2) M_2 simulates the computation of M_1 step by step.
- If and when M_1 reaches a final state, then M_2 can empty its pushdown using (3) and (4).

Even if M_1 is deterministic, M_2 is not. This stems from the problem that M_2 cannot detect when the input has been read completely. Hence, whenever it enters a final state of M_1, then it has the option of continuing the simulation of M_1 using (2) or of emptying its pushdown using (3) (and then (4)). □

Problem 11.4. [DPDA]

Consider the DPDA $M = (\{q_0, q_1, q_2\}, \{a, b\}, \{\#, Z\}, \delta, q_0, \#, \{q_2\})$, where the transition function δ is defined as follows:

$$\begin{aligned}\delta(q_0, a, \#) &= (q_1, \#Z),\\ \delta(q_1, a, Z) &= (q_1, ZZ),\\ \delta(q_1, b, Z) &= (q_2, \varepsilon),\\ \delta(q_2, b, Z) &= (q_2, \varepsilon).\end{aligned}$$

(a) Give a nonempty input word $w \in \{a, b\}^*$ that M does not read completely.

(b) Use the construction from the proof of Lemma 3.33 to extend M to an equivalent DPDA M' that always reads its input words completely.

Solution. (a) On input $w = abb$, M executes the following computation:

$$(q_0, \#, abb) \vdash_M (q_1, \#Z, bb) \vdash_M (q_2, \#, b).$$

(b) Let $M' = (\{q_0', q_0, q_1, q_2, d\}, \{a, b\}, \{\#, Z, X_0\}, \delta', q_0, \#, \{q_2\}$, where δ' is defined by the following table:

δ'	q_0'	q_0	q_1	q_2	d
$(\varepsilon, \#)$	$(q_0, X_0\#)$	$-$	$-$	$-$	$-$
(a, X_0)	$-$	(d, X_0)	(d, X_0)	(d, X_0)	(d, X_0)
$(a, \#)$	$-$	$(q_1, \#Z)$	$(d, \#)$	$(d, \#)$	$(d, \#)$
(a, Z)	$-$	(d, Z)	(q_1, ZZ)	(d, Z)	(d, Z)
(b, X_0)	$-$	(d, X_0)	(d, X_0)	(d, X_0)	(d, X_0)
$(b, \#)$	$-$	$(d, \#)$	$(d, \#)$	$(d, \#)$	$(d, \#)$
(b, Z)	$-$	(d, Z)	(q_2, ε)	(q_2, ε)	(d, Z)

Then

$$\begin{aligned}(q_0', \#, abb) &\vdash_{M'} (q_0, X_0\#, abb) \vdash_{M'} (q_1, X_0\#Z, bb) \vdash_{M'} (q_2, X_0\#, b)\\ &\vdash_{M'} (d, X_0\#, \varepsilon),\end{aligned}$$

that is, M' reads the input abb completely. □

Problem 11.5. [Ogden's Lemma for DCFL]

Prove that the language $L = \{ a^n b^n c, a^n b^{2n} d \mid n \geq 0 \}$ is not deterministic context-free by applying Ogden's Lemma for DCFL (Theorem 3.40) to L.

Solution. Assume that L is deterministic context-free. Let k be the corresponding constant from Thm. 3.40, and let $p := k!$.
Consider the word $z := a^p b^p c \in L$, where we mark all occurrences of b. Then $z = a^p b^p c$ has a factorization $z = a^p b^p c = uvwxy$ that satisfies conditions (1) to (5) of the theorem, that is,

(1) $v \neq \varepsilon$,

(2) $uv^i wx^i y \in L$ for all $i \geq 0$,

(3) u, v and w or w, x and y contain marked positions,

(4) $\delta(vwx) \leq k$,

(5) if $y \neq \varepsilon$, then the following equivalence holds for all $m, n \geq 0$ and all $\alpha \in \Sigma^*$: $uv^{m+n}wx^n\alpha \in L$ iff $uv^m w\alpha \in L$.

As by (2) $z_0 = uwy \in L$ and $z_2 = uv^2wx^2y \in L$, we see that $v = a^i$ and $x = b^i$ for some $i \in \{1, 2, \ldots, k\}$. Hence, u does not contain a marked letter, and so by (3), w, x, and y contain marked positions, that is,

$$u = a^{n_1},\ v = a^i,\ w = a^{p-n_1-i}b^j,\ x = b^i \text{ and}$$
$$y = b^{p-i-j}c \text{ for some } j \in \{1, 2, \ldots, k\}.$$

In fact, we have $i + j < k$. In particular, we have $y \neq \varepsilon$.
Now we choose $m = 1$ and $n = 1$, and we take $\alpha = b^{p-j+p+i}d$. Then

$$uv^{n+m}wx^n\alpha = a^{n_1}a^{i\cdot 2}a^{p-n_1-i}b^j b^i b^{p-j+p+i}d = a^{p+i}b^{p+i+p+i}d \in L,$$

but $uv^m w\alpha = a^{n_1}a^i a^{p-n_1-i}b^j b^{p-j+p+i}d = a^p b^{p+p+i}d = a^p b^{2p+i}d \notin L$, a contradiction! Thus, it follows that $L \notin$ DCFL. □