Fundamentele Informatica 3

voorjaar 2012

http://www.liacs.nl/home/rvvliet/fi3/

Rudy van Vliet
kamer 124 Snellius, tel. 071-527 5777
rvvliet(at)liacs.nl

college 5, 5 maart 2012

6. Context-Free and Non-Context-Free Languages

6.2. Intersections and Complements of CFLs

6.3. Decision Problems Involving Context-Free Languages

Huiswerkopgave 1,
inleverdatum 6 maart 2012, 13:45 uur

Volgende week publicatie Huiswerkopgave 2

6.2. Intersections and Complements of CFLs

Closure properties:

	union	concat.	Kleene *	intersect.	complem.
Reg. languages	×	×	×	×	×
CF languages	×	×	×		

Example 6.10. Two CFLs Whose Intersection Is Not a CFL

$$\begin{aligned} AnBnCn &= \{a^ib^ic^i \mid i \geq 0\} \\ &= \{a^ib^ic^k \mid i,k \geq 0\} \cap \{a^ib^jc^j \mid i,j \geq 0\} \end{aligned}$$

Example 6.11. A CFL Whose Complement Is Not a CFL

$$L = XX' = \{a,b\}^* - XX$$

(hence $L' = XX = \{xx \mid x \in \{a,b\}^*\}$)

$$\begin{aligned} L &= \{y \in \{a,b\}^* \mid \ |y| \text{ is odd}\} \\ &\cup \{x_1x_2 \in \{a,b\}^* \mid \ |x_1| = |x_2| \text{ and } \geq 1 \text{ mismatch}\} \end{aligned}$$

Example 6.12. Another Example with Intersections and Complements

Let

$$\begin{aligned} L_1 &= \{a^ib^jc^k \mid 0 \leq i \leq j\} \\ L_2 &= \{a^ib^jc^k \mid 0 \leq j \leq k\} \\ L_3 &= \{a^ib^jc^k \mid 0 \leq k \leq i\} \end{aligned}$$

Then

$$AnBnCn = L_1 \cap L_2 \cap L_3$$

Example 6.12. Another Example . . . (continued)

Let

$$\begin{aligned} L_1 &= \{a^ib^jc^k \mid i \leq j\} \\ L_2 &= \{a^ib^jc^k \mid j \leq k\} \\ L_3 &= \{a^ib^jc^k \mid k \leq i\} \end{aligned}$$

$$R = \{a\}^*\{b\}^*\{c\}^*$$

Then

$$\begin{aligned} L_1' &= R' \cup \{a^ib^jc^k \mid i > j\} \\ L_2' &= R' \cup \{a^ib^jc^k \mid j > k\} \\ L_3' &= R' \cup \{a^ib^jc^k \mid k > i\} \end{aligned}$$

$$\begin{aligned} AnBnCn &= \left(L_1' \cup L_2' \cup L_3'\right)' \\ &= L_1 \cap L_2 \cap L_3 \end{aligned}$$

Back to regular languages. . .

Theorem 2.15
Suppose $M_1 = (Q_1, \Sigma, q_1, A_1, \delta_1)$ and $M_2 = (Q_2, \Sigma, q_2, A_2, \delta_2)$ are finite automata accepting L_1 and L_2, respectively.
Let M be the FA $(Q, \Sigma, q_0, A, \delta)$, where
$Q = Q_1 \times Q_2$
$q_0 = (q_1, q_2)$
and the transition function δ is defined by the formula
$\delta((p,q),\sigma) = (\delta_1(p,\sigma), \delta_2(q,\sigma))$
for every $p \in Q_1$, every $q \in Q_2$, and every $\sigma \in \Sigma$.

Then
1. If $A = \{(p,q) | \; p \in A_1$ or $q \in A_2\}$, M accepts the language $L_1 \cup L_2$.
2. If $A = \{(p,q) | \; p \in A_1$ and $q \in A_2\}$, M accepts the language $L_1 \cap L_2$.
3. If $A = \{(p,q) | \; p \in A_1$ and $q \notin A_2\}$, M accepts the language $L_1 - L_2$.

Theorem 6.13.

If L_1 is a context-free language and L_2 is a regular language, then $L_1 \cap L_2$ is a CFL.

Sketch of Proof. . .

Let $M_1 = (Q_1, \Sigma, \Gamma, q_1, Z_0, A_1, \delta_1)$ be a PDA accepting L_1.
Let $M_2 = (Q_2, \Sigma, q_2, A_2, \delta_2)$ be an FA accepting L_2.

Then PDA $M = (Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ with

$$Q = Q_1 \times Q_2 \qquad q_0 = (q_1, q_2) \qquad A = A_1 \times A_2$$

And δ . . . ?

Theorem 6.13.

Sketch of Proof. . . (continued)

1. For every $(p,q) \in Q_1 \times Q_2$, $\sigma \in \Sigma$ and $Z \in \Gamma$,

$$\delta((p,q),\sigma,Z) = \{((p',q'),\alpha) \mid (p',\alpha) \in \delta_1(p,\sigma,Z) \text{ and } q' = \delta_2(q,\sigma)\}$$

Theorem 6.13.

Sketch of Proof. . . (continued)

1. For every $(p,q) \in Q_1 \times Q_2$, $\sigma \in \Sigma$ and $Z \in \Gamma$,

$$\delta((p,q),\sigma,Z) = \{((p',q'),\alpha) \mid (p',\alpha) \in \delta_1(p,\sigma,Z) \text{ and } q' = \delta_2(q,\sigma)\}$$

2. For every $(p,q) \in Q_1 \times Q_2$ and $Z \in \Gamma$,

$$\delta((p,q),\Lambda,Z) = \{((p',q),\alpha) \mid (p',\alpha) \in \delta_1(p,\Lambda,Z)\}$$

Theorem 6.13.

Sketch of Proof. . . (continued)

Now,

1. $(q_1, yz, Z_1) \vdash^n_{M_1} (p, z, \alpha)$ and $\delta^*(q_2, y) = q$

if and only if

2. $((q_1,q_2), yz, Z_1) \vdash^n_{M} ((p,q), z, \alpha)$

The details of the (inductive) proof of this statement do not have to be known for the exam.

6.3. Decision Problems Involving Context-Free Languages

Back to regular languages. . .

Example 2.34. Decision Problems Involving Languages Accepted by Finite Automata

1. Membership problem:
Given an FA M and a string x over the alphabet of M,
is $x \in L(M)$?

1. Membership problem for context-free languages:
Given a context-free grammar G and a string x,
is $x \in L(G)$?

Back to regular languages. . .

Example 2.34. Decision Problems Involving Languages Accepted by Finite Automata

2. Given an FA M, is $L(M)$ nonempty ?

3. Given an FA M, is $L(M)$ infinite ?

2. Given a context-free language L, is L nonempty ?

3. Given a context-free language L, is L infinite ?

Theorem 2.29. The Pumping Lemma for Regular Languages
Suppose L is a language over the alphabet Σ.
If L is accepted by a finite automaton $M = (Q, \Sigma, q_0, A, \delta)$,
and if n is the number of states of M,

then for every $x \in L$ satisfying $|x| \geq n$, there are three strings u, v, and w such that $x = uvw$ and the following three conditions are true:

1. $|uv| \leq n$.
2. $|v| > 0$ (i.e., $v \neq \Lambda$).
3. For every $i \geq 0$, the string uv^iw also belongs to L.

Theorem 6.1. The Pumping Lemma for Context-Free Languages

Suppose L is a context-free language. Then there is an integer n so that for every $u \in L$ with $|u| \geq n$, u can be written as $u = vwxyz$, for some strings v, w, x, y and z satisfying

1. $|wy| \geq 0$
2. $|wxy| \leq n$
3. for every $m \geq 0$, $vw^mxy^mz \in L$

Back to regular languages. . .

4. Given FAs M_1 and M_2, is $L(M_1) \cap L(M_2)$ nonempty ?

5. Given FAs M_1 and M_2, is $L(M_1) \subseteq L(M_2)$?

4. Given CFGs G_1 and G_2, is $L(G_1) \cap L(G_2)$ nonempty ?

5. Given CFGs G_1 and G_2, is $L(G_1) \subseteq L(G_2)$?

Back to regular languages. . .

4. Given FAs M_1 and M_2, is $L(M_1) \cap L(M_2)$ nonempty ?

5. Given FAs M_1 and M_2, is $L(M_1) \subseteq L(M_2)$?

4. Given CFGs G_1 and G_2, is $L(G_1) \cap L(G_2)$ nonempty ?

5. Given CFGs G_1 and G_2, is $L(G_1) \subseteq L(G_2)$?

Undecidable !

Exercise.

Let M be a PDA.

Construct a PDA M' such that $L(M') = L(M)$
and M' has no Λ-transitions from an accepting state.

Exercise.

Let M be a DPDA.

Construct a DPDA M' such that $L(M') = L(M)$
and M' has no Λ-transitions from an accepting state.

Excercise 5.20.

Show that if L is accepted by a DPDA M, then there is a DPDA M' accepting the language $\{x\#y \mid x \in L \text{ and } xy \in L\}$.
(The symbol # is assumed not to be in any of the strings of L.)