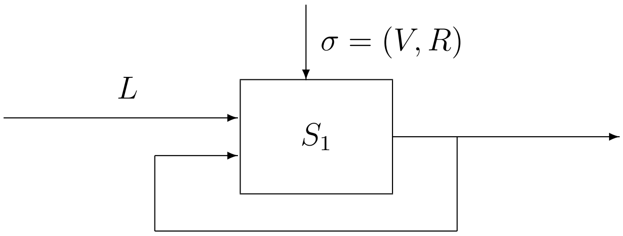
Prof. Dr. Hans-Jörg Kreowski **Juli 2007**

Formale Sprachen: DNA Computing

4. Übungsblatt

Das ist bezüglich Aufgabe 5 eine revidierte Fassung, weil die vorige eine falsche Aussage enthielt, die sich dann natürlich nicht beweisen lässt. Ich hoffe, dass es jetzt stimmt.

Da man bei Eingabe von $L \subseteq V^*$ in die 1-Splicing-Maschine S_1 als Ausgabe ebenfalls eine Sprache $\sigma_1(L) \subseteq V^*$ erhält, kann man 1-Splicing für ein gegebenes Regelsystem $\sigma = (V, R)$ beliebig wiederholen:



Das definiert das iterierte 1-Splicing, das für L die Sprache $\sigma_1^*(L)$ liefert, die definiert ist durch:

$$\sigma_1^*(L) = \bigcup_{i \in \mathbb{N}} \sigma_1^i(L)$$

mit $\sigma_1^0(L) = L$ und $\sigma_1^{i+1}(L) = \sigma_1^i(L) \cup \sigma_1(\sigma_1^i(L))$ für $i \in \mathbb{N}$. Iteriertes 1-Splicing lässt sich wie 1-Splicing als Sprachklassengenerator auffassen:

$$H_1(FL_1, FL_2) = \{\sigma_1^*(L) \mid L \in FL_1, \sigma = (V, R), R \in FL_2\}.$$

Dazu einige Aufgaben:

1. Betrachte $\sigma(V) = (V, \{a\#\$\#b \mid a, b \in V\})$ und zeige, dass $\sigma_1^*(L)$ unendlich ist, falls $V \neq \emptyset$ und $\emptyset \neq L \neq \{\lambda\}$.

2. Zeige $FL_1 \subseteq H_1(FL_1, FL_2)$ für alle Sprachfamilien FL_1 und FL_2, falls FL_2 alle endlichen Sprachen enthält.

3. Begründe $H_1(RE, RE) \subseteq RE$.

4. Sei $k \geq 1$ fest gewählt. Betrachte

$$\sigma(k) = (\{a, b\}, \{a^{k+1}\#\ a^k\ \$\ a^{k+1}\ \#\ a^k\}) \text{ und } \mathrm{L}'_\mathrm{k} = \{\mathrm{a}^{2\mathrm{k}}\mathrm{b}^{2\mathrm{k}}\mathrm{a}^{2\mathrm{k}+2}\mathrm{b}^{2\mathrm{k}}\mathrm{a}^{2\mathrm{k}}\}$$

und zeige:

$$\sigma(k)_1^*(L'_k) = L_k = \{a^{2k}b^{2k}a^nb^{2k}a^{2k} \mid n \geq 2k+1\}.$$

5. Betrachte die rechtslineare Grammatik

$$G = (\{S, A, B\}, T = \{a, b, c\}, \{S ::= cA, A ::= aB, B ::= bA, B ::= c\}, S)$$

sowie die endliche Sprache

$$A_G = \{S, ZcA, ZaB, ZbA, ZZc\}$$

und die endliche Menge von Splicing-Regeln

$$R_G = \{\#S\$Z\#cA, \#A\$Z\#aB, \#B\$Z\#bA, \#B\$ZZ\#c\}.$$

Die Frage ist, was die Grammatik G mit der regulären Sprache $L(G)$ einerseits und die iterierte 1-Splicing-Sprache $\sigma(G)_1^*(A_G)$ für $\sigma(G) = (\{S, A, B, Z, a, b, c\}, R_G)$ andererseits miteinander zu tun haben.

Beweise dazu mit vollständiger Induktion über $k \in I\!N$:

(a) $S \xrightarrow{k} uX$ für $u \in T^*$ und $X \in \{S, A, B, \lambda\}$ impliziert $uX \in \sigma(G)_1^k(A_G)$.

(b) $w \in \sigma(G)_1^k(A_G)$ impliziert $w = ZZc$ oder $w = ZuY$ für ein $u \in T^*$ und $Y \in \{S, A, B\}$ oder $w = uX$ für ein $u \in T^*$ und $X \in \{S, A, B, \lambda\}$ sowie $S \xrightarrow{k} uX$.

Zeige mit Hilfe dieser beiden Beobachtungen:

(c) $L(G) = \sigma(G)_1^*(A_G) \cap T^*$.

Abgabe bis 20. Juli 2007 in mein Postfach 6. Ebene MZH