Automata and Logic

Winter Semester 2018 / 2019

Exercise Sheet 12

23rd January 2019

Automata on Finite Trees

PD Dr.-Ing. habil. Anni-Yasmin Turhan, Dipl.-Math. Francesco Kriegel

Exercise 12.1 Let $\Sigma := \Sigma_0 \cup \Sigma_1 \cup \Sigma_2$ be a finite alphabet with arity function where $\Sigma_0 := \{x, y, z\}$, $\Sigma_1 := \{\neg\}$, and $\Sigma_2 := \{\wedge, \vee\}$. For each of the following sets of trees, define a tree automaton (either LR or RL) recognizing it.

(a) $\{$trees containing the symbol $\vee$ exactly once$\}$

(b) $\{$trees containing the symbol $\neg$ at least once on every path of the tree$\}$

(c) $\{$trees describing *satisfiable* propositional formulae$\}$

Exercise 12.2 Let $\mathcal{A} := (Q, \Sigma, I, \Delta, F)$ be the non-deterministic LR-tree automaton with the following components.

$$Q := \{0, 1\}$$

$\Sigma := \{f, x, y\}$ with $\nu(f) := 2$ and $\nu(x) := \nu(y) := 0$

$$I(x) := \{0, 1\}$$
$$I(y) := \{0\}$$

$$\Delta_f(0, 0) := \{0\}$$
$$\Delta_f(0, 1) := \{1, 0\}$$
$$\Delta_f(1, 0) := \{1, 0\}$$
$$\Delta_f(1, 1) := \{1\}$$

$$F := \{1\}$$

(a) Adapt the standard power-set construction from finite automata on words to LR-tree automata and use it to construct a deterministic LR-tree automaton $\mathcal{A}'$ such that $L(\mathcal{A}) = L(\mathcal{A}')$ holds true.

(b) Try to apply a similar construction to the RL-tree automaton from Example 6.10. Explain why this method fails for RL-tree automata.

Exercise 12.3 Let Σ be a finite alphabet with arity function, $x, y \in \Sigma_0$, and $U, V, W \subseteq \mathbf{T}_\Sigma$. Prove or refute each of the following claims.

(a) $U \cdot^x (V \cup W) = (U \cdot^x V) \cup (U \cdot^x W)$

(b) $(U \cdot^x V) \cdot^y W = U \cdot^x (V \cdot^y W)$

(c) $(U^{*,x})^{*,y} = (U^{*,y})^{*,x}$

Exercise 12.4 Example 6.10 shows that deterministic RL-tree automata recognize a smaller class of languages than non-deterministic ones. We call an RL-tree automaton $\mathcal{A} := (Q, \Sigma, I, \Delta, F)$ *quasi-deterministic* if Δ is a deterministic transition assignment, that is,

$$|\Delta_a(q)| \leq 1$$

holds true for each $a \in \Sigma_n$ where $n > 0$ and for any $q \in Q$. In particular, we can then treat each Δ_a as a partial function of type $Q \to Q^n$. Note that $I \subseteq Q$ may still be a *set*, i.e., more than one initial state is allowed.

Prove or refute each of the following statements.

(a) If $L \subseteq \mathbf{T}_\Sigma$ is a *finite* tree language, then there exists a quasi-deterministic tree automaton recognizing L.

(b) If $L \subseteq \mathbf{T}_\Sigma$ is a *recognizable* tree language, then there exists a quasi-deterministic tree automaton recognizing L.

Exercise 12.5 Let $\mathcal{A} := (Q, \Sigma, I, \Delta, F)$ be an RL-tree automaton that has the following components.

$$Q := \{1, \ldots, 4\}$$

$\Sigma := \{g, n, a, b\}$ with $\nu(g) := 2$, $\nu(n) := 1$, and $\nu(a) := \nu(b) := 0$

$$I := \{1\}$$

$$\begin{aligned}
\Delta_g(1) &:= \{(1,1), (1,2), (3,4), (4,1)\} \\
\Delta_g(2) &:= \emptyset \\
\Delta_g(3) &:= Q \times Q \\
\Delta_g(4) &:= \{(1,2), (1,4), (2,4), (2,2)\}
\end{aligned}$$

$$\begin{aligned}
\Delta_n(1) &:= \{1\} \\
\Delta_n(2) &:= \{3\} \\
\Delta_n(3) &:= \{1,2\} \\
\Delta_n(4) &:= \{1,3\}
\end{aligned}$$

$$\begin{aligned}
F(a) &:= \{2\} \\
F(b) &:= \{2,3\}
\end{aligned}$$

Decide whether $L(\mathcal{A})$ is empty.

Exercise 12.6 Devise a polynomial time algorithm that decides the emptiness problem for LR-tree automata.