Tutorial 6: Tree automata

(1) Let $\mathcal{M}$ be a bottom-up deterministic tree automata. Describe an algorithm for the following problem.

MEMBERSHIP$_{\mathcal{M}}$
Input: A tree T
Task: Output "Yes," if $T \in L(\mathcal{M})$. Otherwise, output "No."

What if $\mathcal{M}$ is non-deterministic?

(2) Describe an algorithm for the following problem.

MEMBERSHIP
Input: A tree T and a deterministic bottom-up tree automata $\mathcal{M}$.
Task: Output "Yes," if $T \in L(\mathcal{M})$. Otherwise, output "No."

What if $\mathcal{M}$ is non-deterministic?

Appendix

Let $\Sigma = \{a_1, \ldots, a_m\}$. A Σ tree $T = (r, I, F, \ell ab)$ can be viewed as a first-order structure:

$$\mathcal{A}_T \quad = \quad (I, \text{succ}_0, \text{succ}_1, P_{a_1}, \ldots, P_{a_m})$$

where

- $\text{succ}_0(u, v)$ holds, if and only if v is the left child of u;
- $\text{succ}_1(u, v)$ holds, if and only if v is the right child of u;
- for every $a_i \in \Sigma$, for every $u \in I$, $P_{a_i}(u)$ holds if and only if $\ell ab(u) = a_i$.

First-order logic for trees can be defined with relations $\text{succ}_0, \text{succ}_1, P_{a_1}, \ldots, P_{a_m}$. Monadic Second Order (MSO) logic is defined by allowing quantification over *subsets* of the domain:

$$\forall X \ \varphi \quad \text{or} \quad \exists X \ \varphi$$

where X is a "new" unary relation symbol that can be used inside φ. Intuitively, they mean "for every possible set X, φ hold" and "there is a set X such that φ holds." We can write $L(\varphi)$ to be the set of all trees for which φ holds.

It is known that for every (bottom-up) tree automata $\mathcal{M}$, there is an MSO sentence φ such that $L(\mathcal{M}) = L(\varphi)$. Vice versa, for every MSO sentence φ, there is a (bottom-up) tree-automata $\mathcal{M}$ such that $L(\mathcal{M}) = L(\varphi)$. For more details, please consult [1, 2].

References

[1] H. Comon, M. Dauchet, R. Gilleron, C. Löding, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. Tree automata techniques and applications. Available on: `http://www.grappa.univ-lille3.fr/tata`, 2007. release October, 12th 2007.

[2] L. Libkin. *Elements of Finite Model Theory.* Texts in Theoretical Computer Science. An EATCS Series. Springer, 2004.