Tree Automata and their Applications

TD n°3 : Logic and Hedges

2021-2022

Exercise 1 : MSO on finite trees

We consider trees with maximum arity 2. Give MSO formulae which express the following :

1. X is closed under predecessors
2. $x \subseteq y$ (with $\subseteq$ the prefix relation on positions)
3. 'a' occurs twice on the same path
4. 'a' occurs twice not on the same path
5. There exists a sub tree with only a's
6. The frontier word contains the chain 'ab'

Exercise 2 : The power of Wsks

Produce formulae of WSkS for the following predicates :

- the set X has exactly two elements.
- the set X contains at least one string beginning with a 1.
- $x \leq_{lex} y$ where $\leq_{lex}$ is the lexicographic order on $\{1, ...k\}^*$.
- given a formula of WSkS ϕ with one free first-order variable, produce a formula of WSkS expressing that there is an infinity of words on $\{1, ..., k\}^*$ satisfying ϕ.

Exercise 3 : The limit of Wsks

Prove that the predicate $x = 1y$ is not definable in WSkS.

Exercise 4 : Alternating Word Automata

Definition 1 *If $\mathcal{X}$ is a set of propositional variables, let $\mathbb{B}(\mathcal{X})$ be the set of positive propositional formulae on $\mathcal{X}$, i.e., formulae generated by the grammar $\phi ::= \bot \mid \top \mid \phi \vee \phi \mid \phi \wedge \phi$.*

Definition 2 *A AWA (Alternating Word Automata) is a tuple $\mathcal{A} = (Q, \Sigma, Q_0, Q_f, \delta)$ where Σ is a finite set (alphabet), Q is a finite set (of states), $Q_0 \subset Q$ (initial states), $Q_f \subseteq Q$ (final states) and δ is a function from $Q \times \Sigma$ to $\mathbb{B}(Q)$ (transition function). A run of $\mathcal{A} = (Q, \Sigma, Q_0, Q_f, \delta)$ on a word w is a tree t labelled by Q such that :*

— *if $w = \varepsilon$, then $t = q_0$ with $q_0 \in Q_0$.*

— *if $w = a.w'$, then $t = q_0(t_1, \ldots, t_n)$ $q_0 \in Q_0$ and such that for all i, t_i is a run of w' on $(Q, \Sigma, q_i, Q_f, \delta)$ and $\{q_1, \ldots, q_n\} \models \delta(q_0, a)$.*

Definition 3 *We say that a run is accepting if every leaf of the form q satisfies that $q \in Q_f$.*

1. Let $\Sigma = \{0,1\}$ and $\mathcal{A} = (Q, \Sigma, q_0, Q_f, \delta)$ the AWA such that $Q = \{q_0, q_1, q_2, q_3, q_4, q_1', q_2'\}$, $Q_f = \{q_0, q_1, q_2, q_3, q_4\}$ and :

$$\delta = \{ \begin{array}{ll} q_0 0 \longrightarrow (q_0 \wedge q_1) \vee q_1' & q_0 1 \longrightarrow q_0 \\ q_1 0 \longrightarrow q_2 & q_1 1 \longrightarrow \top \\ q_2 0 \longrightarrow q_3 & q_2 1 \longrightarrow q_3 \\ q_3 0 \longrightarrow q_4 & q_3 1 \longrightarrow q_4 \\ q_4 0 \longrightarrow \top & q_4 1 \longrightarrow \top \\ q_1' 0 \longrightarrow q_1' & q_1' 1 \longrightarrow q_2' \\ q_2' 0 \longrightarrow q_2' & q_2' 1 \longrightarrow q_1' \} \end{array}$$

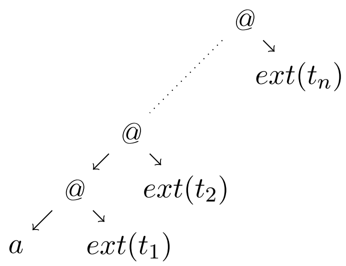
   Give an example of an accepting computation of $\mathcal{A}$ on $w = 00101$ and an example of a non accepting computation of $\mathcal{A}$ on w.
2. Prove that for all AWA, we can compute in exponential time a non-deterministic automaton which accepts the same language.
3. Show how to reduce the emptiness problem for an AWA on a one letter alphabet $\{a\}$ with formalas that are in positive disjunctive normal form to the emptiness problem of a tree automaton .
4. Show how to reduce the emptiness problem for a tree automaton to the emptiness problem of an AWA on a one letter alphabet $\{a\}$. Conclude on the complexity of the emptiness problem for an AWA on a one letter alphabet.

Exercise 5 : Extensions

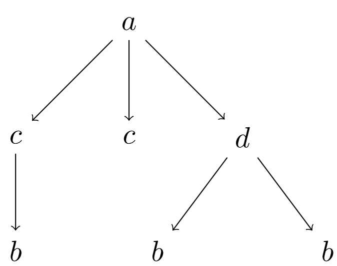
Definition 4 (extension encoding)
Let t be an unranked tree on Σ. Let $\mathcal{F}_{ext}^{\Sigma} = \{@(2)\} \cup \{a(0) \mid a \in \Sigma\}$. We define the ranked tree $ext(t)$ by induction on the size of t by :

- *for $a \in \Sigma$, $ext(a) = a$*
- *if $t = a(t_1, ..., t_n)$ with $n \geq 1$, $ext(t) = @(ext(a(t_1, ..., t_{n-1})), ext(t_n))$*

that is $ext(a(t_1, ..., t_n))$ is equal to :



Give the extension encoding of :



Exercise 6 : The soundess of the extension
Let L be a language of unranked trees. Prove that L is recognizable by a NFHA iff $ext(L)$ is recognizable by a NFTA.

⋆⋆⋆

Homework for next time : To the infinity...

Let $\Sigma = \{a, b\}$. Define a DFHA $\mathcal{A}$ such that $L(\mathcal{A})$ is the set of all trees such that "for every leaf labeled with a, there is an ancestor from which there is a path whose nodes are labeled with b". Here "ancestor" means strict ancestor and "from which there is a path" means that there is a path from a son of this ancestor to a leaf.

⋆⋆⋆

Homework for next time : Membership

1. Recall the complexity of the uniform membership problem for DFTAs, NFTAs and NFHAs.
2. Prove that **(AlternatingUMembership)** :
   Instance : an AWA $\mathcal{A}$ and a word w
   Question : $w \in L(\mathcal{A})$?
   is in PTIME.
3. Prove that **(HarderUMembership)** :
   Instance : an NFHA $\mathcal{A}$ where the horizontal languages are given by AWA (and not finite automata) and a word w
   Question : $w \in L(\mathcal{A})$?
   is in NP.
4. Let Φ be a propositional formula in CNF with n variables $x_1, ..., x_n$. Construct, in polynomial time, an AWA $\mathcal{A}_\Phi$ whose language is $\{w \in \{0,1\}^n \mid w \models \Phi \text{ i.e. } \Phi_{[x_i \leftarrow w_i]} = \top\}$.
5. Deduce that membership for NFHA where horizontal languages are given by AWA is NP-complete.

Note : You can send the homework by mail to asuresh@lsv.fr, or hand it to me in person on 6th January, 2022 8.30 am. ***This is a slightly longer homework assignment and will be equivalent to two regular homework assignments.***