Automates d'arbre

TD n°4 : Logic and Hedges

Charlie Jacomme

October 03, 2017

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'

Solution:

1. $closed(X) := \forall y \forall z (y \in X \wedge (z \downarrow_1 y) \vee z \downarrow_2 y)) \Rightarrow z \in X)$
2. $x \subseteq y := \forall X (y \in X \wedge closed(X) \Rightarrow X(x))$
3. $\exists x \exists y (\neg(x = y) \wedge x \subseteq y \wedge P_a(x) \wedge P_a(y))$
4. $\exists x \exists y (\neg(y \subseteq x) \wedge \neg(x \subseteq y) \wedge Pa(x) \wedge P_a(y))$
5. $\exists x \forall y (x \subseteq y \Rightarrow P_a(y))$
6. We first implement a way to say that a leaf is next to another one :

$$x \prec y := \exists x_0 \exists y_0 \exists z (z \downarrow_1 x_0) \wedge (z \downarrow_2 y_0) \wedge x_0 \subseteq x \wedge y_0 \subseteq y)$$

   And with this :

$$\exists x \exists y (Fr(x) \wedge Fr(y) \wedge P_a(x) \wedge P_b(y) \wedge x \prec y \wedge \neg \exists z (Fr(z) \wedge x \prec z \wedge z \prec y))$$

Exercise 2 : From formulaes to automaton

Give tree automatons recognizing the languages on trees of maximum arity 2 defined by the formulae :

1. $(x \in S \wedge (x \downarrow_1 y \Rightarrow y \in S)) \wedge (z \in S \Rightarrow P_f(z))$
2. $\exists S.(x \in S \wedge (x \downarrow_1 y \Rightarrow y \in S)) \wedge (z \in S \Rightarrow P_f(z))$

Solution:

1. We construct an NFTA $\mathcal{A}_1$ on $\Sigma \times \{0,1\}^2$, which recognizes $x \in S$. The idea is to reject if we can witness a $x \notin S$, and we accept otherwise. So, for all $f \in \mathcal{F}$:
   — $(f, 1, 0)(q_1, q_2) \longrightarrow \perp \ \ if \ \forall i, q_i \neq \perp$
   — $(f, \_, \_)(q_1, q_2) \longrightarrow \top \ if \ \forall i, q_i \neq \perp$
   We construct an NFTA $\mathcal{A}_2$ on $\Sigma \times \{0,1\}^3$, which recognizes $(x \downarrow_1 y \Rightarrow y \in S)$). If we witness a $y \notin S$, we go into a specific state to check if it is not the son of x, thus failing the formula.

— $(f,,1,0)(q_1,q_2) \longrightarrow q_{y\notin S}$ if $\forall i, q_i \neq \bot$
— $(f,,1,0)(q_{y\notin S},q_2) \longrightarrow \bot$
— $(f,\_,\_)(q_1,q_2) \longrightarrow \top$ if $\forall i, q_i \neq \bot$
We construct an NFTA $\mathcal{A}_3$ on $\Sigma \times \{0,1\}^2$, which recognizes $(z \in S \Rightarrow P_h(z))$.
— $(f,1,1)(q_1,q_2) \longrightarrow \bot$ if $\forall i, q_i \neq \bot, \forall f \neq h \in \mathcal{F}$
— $(f,\_,\_)(q_1,q_2) \longrightarrow \top$ if $\forall i, q_i \neq \bot, \forall f \in \mathcal{F}$
Then, with the correct inversed projections, we can transform A_i into A'_i on $\Sigma \times \{0,1\}^4$ with ordering (x,y,z,S), and $\bigcap A'_i$ is the desired automaton.

2. We project $\bigcap A'_i$ on $\Sigma \times \{0,1\}^3$, and we obtain the result.

Exercise 3 : 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.

Solution:

- $$|X| \leq 2 \doteq \forall Y.\, Y \subseteq X \Rightarrow (Y = \varnothing \vee Sing(Y) \vee Y = X)$$
  $$|X| \geq 2 \doteq \exists x,y.\, x \neq y \wedge x \in X \wedge y \in X$$
  $$|X| = 2 \doteq |X| \leq 2 \wedge |X| \geq 2$$
- $$X \cap 1.\Sigma^* \neq \varnothing \doteq \exists x.\, x \in X \wedge 1 \leq x$$
- $$x \leq_{lex} y \doteq x \leq y \vee (\exists z.\, \bigvee_{i<j\leq k} (z.i \leq x \wedge z.j \leq y))$$
- $$X \models \phi \doteq \forall x, x \in X \Rightarrow \phi(x)$$
  $$\phi \text{ satisfied by an infinity of words } \doteq \forall X,\, X \models \phi \Rightarrow \exists Y,\, X \subsetneq Y \wedge Y \models \phi$$

Exercise 4 : The limit of Wsks

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

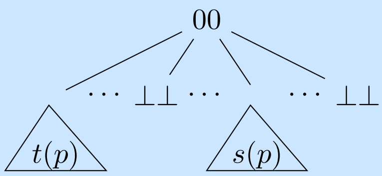
Solution:

We use the equivalence with recognizable tree languages. So we have to prove that $L = \{tra(x,y) \mid x = 1.y\}$ is not recognizable. Using the translation, we see that

$$L \cap \{t_i\sigma \mid t_i = 00(i\bot(x_1,...,x_k), y_2,...,y_k), i \in \{0,1\}, \sigma \text{ closed substitution}\}$$

$$= \{tra(x,y) \mid x = 1.y \wedge y \in \{2,...,k\}.\{1,...,k\}^*\} = L'$$

So it is enough to prove that L' is not recognizable. Now elements of L' are of the form :



with $p \in \{2, ..., k\}.\{1, ..., k\}^*$, t and s injective and the height of t and s strictly increasing with p. You can reason by contradiction using the pumping lemma : for p large enough, using the pumping lemma, you can iterate a piece of $t(p)$ without touching $s(p)$ (or vice versa) while staying in L' which is absurd by injectivity.

⋆ ⋆ ⋆

Homework for next week : 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.