Automates d’arbre

TD n°6 : Two-way automata & PDL

Exercise 1 :

Let \( A \) be a NFTA on an alphabet \( F = F' \cup \{ \} \) where \( F' \) contains a constant. Construct a two-way automata in the Horn clause formalism which recognizes the set of terms that can be deduced from this set of rules :

\[
\begin{align*}
\frac{t}{\text{if } t \in L(A)} & \quad \frac{u \circ v}{\{u\}_v} \quad \text{encrypt} & \quad \frac{\{u\}_v \circ v}{u} \quad \text{decrypt}
\end{align*}
\]

Explain why it is difficult to do the same construction with the other definition of two-way automata.

Exercise 2 :

We have a naive translation from two-way automata (in the 'classic' sense) to two-way automata (in the Horn clause formalism) which does not work : let us assume that \( F \) only contains unary symbols and constants and that you have an automaton \( A \) which does not have non-deterministic or alternating rules. So \( A \) only has rules of the form \( \delta(q,a) = (q',d) \) with \( d \in \{-1,1\} \) or \( \delta(q,a) = \text{true} \). For each transition \( \delta(q,a) = (q',1) \), we consider the clause \( q'(x) \rightarrow q(a(x)) \), for each transition \( \delta(q,a) = (q',-1) \), we consider the clause \( q'(f(a(x))) \rightarrow q(a(x)) \) for all \( f \in F \) and for each transition \( \delta(q,a) = \text{true} \), we consider the clause \( \rightarrow q(a(x)) \). Let us call \( C(A) \) this set of clauses.

1) Which clauses of \( C(A) \) are not of the shape of the Horn clause formalism of a two-way automaton?
2) Give a two-way automaton in the Horn clause formalism which recognizes the interpretation of \( q_f \) in the smallest Herbrand model of \( C(A) \).
3) Give an example of an automaton \( A \) such that the automaton obtained at the question 2) does not recognize the same language.

Definition (PDL).

The syntax is the following :

\[
\begin{align*}
\phi & ::= a \mid \top \mid \neg \phi \mid \phi \lor \phi \mid (\pi) \phi & \quad \text{(position formulae)}
\end{align*}
\]

\[
\begin{align*}
\pi & ::= \downarrow \mid \rightarrow \mid \pi^{-1} \mid \pi \mid \pi + \pi \mid \pi^* \mid \phi? & \quad \text{(path formulae)}
\end{align*}
\]

The semantic is defined this way : let \( t \) be a tree, we define \( [\phi]_t \) (resp. \( [\pi]_t \)) as a set of positions of \( t \) (resp. a relation on positions of \( t \)) by induction on the size of \( \phi \) (resp. \( \pi \)) :

\[
\begin{align*}
[a]_t &= \{ w \in \text{Pos}(t) \mid t(w) = a \} & [\downarrow]_t &= \{ (w, w.i) \mid w, w.i \in \text{Pos}(t) \}
\end{align*}
\]

\[
\begin{align*}
[\top]_t &= \text{Pos}(t) & [\rightarrow]_t &= \{ (w, i, w.(i+1)) \mid w, i, w.(i+1) \in \text{Pos}(t) \}
\end{align*}
\]

\[
\begin{align*}
[\neg \phi]_t &= \text{Pos}(t) \setminus [\phi]_t & [\pi^{-1}]_t &= [\pi]_t^{-1}
\end{align*}
\]

\[
\begin{align*}
[\phi_1 \lor \phi_2]_t &= [\phi_1]_t \cup [\phi_2]_t & [\pi_1; \pi_2]_t &= [\pi_2]_t \circ [\pi_1]_t
\end{align*}
\]

\[
\begin{align*}
[(\pi) \phi]_t &= [\pi]_t^{-1} ([\phi]_t) & [\pi_1 + \pi_2]_t &= [\pi_1]_t \cup [\pi_2]_t
\end{align*}
\]

\[
\begin{align*}
[\pi^*]_t &= [\pi]_t & [\phi?]_t &= \Delta_{[\phi]}_t = \{ (w, w) \mid w \in [\phi]_t \}
\end{align*}
\]

Let \( t \) be a tree and \( w, w' \in \text{Pos}(t) \). We note :
$t, w \models \phi$ if $w \in \llbracket \phi \rrbracket_t$
$t \models \phi$ if $t, \epsilon \models \phi$ and we say that $t$ satisfies $\phi$
$t, w, w' \models \pi$ if $(w, w') \in \llbracket \pi \rrbracket_t$

**Exercise 3:**

Let $t$ be the tree:

```
    a
   / \  \\
  b   a   c
   \  /   /  \\
    \ b  b
   /  /   \\
  a  /  \\
```

Which formulae are satisfied by $t$?

1. $\phi_1 = \neg a \lor \langle \downarrow \rangle (\neg \langle \leftarrow \rangle \top \land b \land (\rightarrow^*) (c \land \neg \langle \rightarrow \rangle \top))$
2. $\phi_2 = \neg a \lor \langle \downarrow \rangle (\neg \langle \leftarrow \rangle \top \land b \land ((\rightarrow ; c?)^*) (\neg \langle \rightarrow \rangle \top))$
3. $\phi_3 = \langle (a? ; \downarrow)^* \rangle (a \land \neg \langle \downarrow \rangle \top)$

**Exercise 4:**

Give a translation of PDL in MSO which preserves models. That is, given a position formula $\phi$ (resp. a path formula $\pi$), construct a MSO formula $\tilde{\phi}$ (resp. $\tilde{\pi}$) whose set of free variable is $\{X_a \mid a \in F\} \cup \{x\}$ (resp. $\{X_a \mid a \in F\} \cup \{x, y\}$) such that $t, w \models \phi$ iff $(P_a(t))_{a \in F}, w \models \tilde{\phi}$ (resp. $t, w, w' \models \pi$ iff $(P_a(t))_{a \in F}, w, w' \models \tilde{\pi}$) where $P_a(t) = \{ w \in Pos(t) \mid t(w) = a \}$.

**Exercise 5:**

Fix an alphabet $F$. Give a PDL formula $\pi$ such that:
- for all tree $t$ and all position $p$ of $t$, there exists exactly one position $q$ of $t$ such that $(p, q) \in \llbracket \pi \rrbracket_t$ ($\pi$ defines a function on positions).
- for all tree $t$ and position $p$ of $t$, $(p, q) \in \llbracket \pi^* \rrbracket_t$ iff $q$ is a position of $t$ such that $t(q) = t(p)$. 