CSC2428 – Foundations of XML
Lecture 6

notes taken by Pablo Barceló

1 MSO over Ordered Trees and L_μ

We consider the modal μ-calculus L_μ. This is interpreted over finite *transition systems* $\mathcal{K} = \langle S, (E_r)_{r\in R}, (P_a)_{a\in\Sigma}\rangle$, where R is finite list of binary relation symbols and each E_r is interpreted as a subset of $S\times S$; each P_a is a subset of S. We shall view a Σ-tree $T = \langle D, \prec_{\text{ch}}, \prec_{\text{sb}}, \prec^*_{\text{ch}}, \prec^*_{\text{sb}}, (P_a)_{a\in\Sigma}\rangle$ as a transition system with two binary relations $\prec_{\text{ch}}$ and $\prec_{\text{sb}}$ (their transitive closures $\prec^*_{\text{ch}}$ and $\prec^*_{\text{sb}}$ are expressible in L_μ).

The formulae of L_μ are given by

$$\varphi := a \ (a \in \Sigma) \ | \ X \ | \ \varphi \vee \varphi \ | \ \neg\varphi \ | \ \Diamond(E_r)\varphi \ | \ \mu X \ \varphi(X),$$

where in $\mu X \ \varphi(X)$, the variable X must occur positively in φ.

Let $\mathcal{K}$ be a transition system. For a L_μ formula $\varphi(X, Y_1, \ldots, Y_m)$ and a valuation v that assigns a subset V_i of S to each Y_i, $i \in [1, m]$, we define an operator $F^v_\varphi : 2^S \to 2^S$ as

$$F^v_\varphi(V) \ = \ \{ s \ | \ (\mathcal{K}, s) \models \varphi(V, V_1, \ldots, V_m)\}.$$

It can be shown that the operator F^v_φ is *monotone*, that is, for each $X, Y \subseteq S$, $F^v_\varphi(X) \subseteq F^v_\varphi(Y)$ whenever $X \subseteq Y$ (the proof uses the fact that X only occurs positively on φ). Therefore, for the sequence

$$X^0 = \emptyset, \qquad X^{j+1} = F^v_\varphi(X_j)$$

it is the case that for each $j \geq 0$, $X^j \subseteq X^{j+1}$, and since S is finite, there is $\ell \geq 0$ for which $X^\ell = X^{\ell+1}$. Furthermore, the fact that F^v_φ is monotone implies $X^\ell = X^m$ for each $m > \ell$. We denote such X^ℓ by $\texttt{lfp}(F^v_\varphi)$ (intuitively, $\texttt{lfp}(F^v_\varphi)$ is the *least fixpoint* of the operator F^v_φ).

Given $\mathcal{K}$, $s \in S$, and a valuation v for free variables (such that each $v(X)$ is a subset of S), we define the semantics (omitting the rules for propositional letters and Boolean connectives) by

- $(\mathcal{K}, v, s) \models a$ iff $s \in P_a$.
- $(\mathcal{K}, v, s) \models X$ iff $s \in v(X)$.
- $(\mathcal{K}, v, s) \models \neg\varphi$ iff $(\mathcal{K}, v, s) \not\models \varphi$.
- $(\mathcal{K}, v, s) \models \varphi \vee \varphi$ iff $(\mathcal{K}, v, s) \models \varphi$ or $(\mathcal{K}, v, s) \models \varphi'$.
- $(\mathcal{K}, v, s) \models \Diamond(E_r)\varphi$ iff $(\mathcal{K}, v, s') \models \varphi$ for some s' with $(s, s') \in E_r$.
- $(\mathcal{K}, v, s) \models \mu X \ \varphi(X)$ iff $s \in \texttt{lfp}(F^v_\varphi)$.

When we refer to L_μ being equivalent to a logic on trees, we mean L_μ formulae without free variables (which are then evaluated in an element of $\mathcal{K}$). As usual, we define $\Box(E_r)\varphi$ as $\neg\Diamond(E_r)\neg\varphi$. If we list explicitly binary relations E_i's, we write $L_\mu[E_1, \ldots, E_k]$ to refer L_μ formulae that only use those relations. For example, $L_\mu[\prec_{\mathrm{ch}}, \prec_{\mathrm{sb}}]$ refers to L_μ formulae that use both $\prec_{\mathrm{ch}}$ and $\prec_{\mathrm{sb}}$ relations.

The *full* μ-calculus L_μ^{full} (cf. [7]) adds backward modalities $\Diamond(E^-)\varphi$ with the semantics $(\mathcal{K}, s) \models \Diamond(E^-)\varphi$ iff $(\mathcal{K}, s') \models \varphi$ for some s' such that $(s', s) \in E$ (and $\Box(E^-)\varphi$ is again $\neg\Diamond(E^-)\neg\varphi$). Then,

Theorem 1 *[1] The equivalence*

$$\mathrm{MSO}[\prec_{\mathrm{ch}}, \prec_{\mathrm{sb}}] \;=\; L_\mu^{\mathrm{full}}[\prec_{\mathrm{ch}}, \prec_{\mathrm{sb}}]$$

hold for unary queries over unranked trees. Furthermore, all translations between these formalisms are effective.

From the proof it is possible to obtain the following corollary. We assume that relation $\prec_{\mathrm{fc}}$ is interpreted in a tree with domain D as follows: $s \prec_{\mathrm{fc}} s'$ iff $s.s' \in D$ and $s' = s \cdot 0$, that is, s' is the first child of s in the sibling order.

Corollary 1 *[1, 6] The following equivalences hold for Boolean queries over unranked trees,*

$$\mathrm{MSO}[\prec_{\mathrm{ch}}, \prec_{\mathrm{sb}}] \;=\; L_\mu^{\mathrm{full}}[\prec_{\mathrm{ch}}, \prec_{\mathrm{sb}}] \;=\; L_\mu[\prec_{\mathrm{fc}}, \prec_{\mathrm{ch}}, \prec_{\mathrm{sb}}],$$

and translations between these formalisms are effective.

We do not know much about model checking of L_μ and L_μ^{full} over trees at this point, except for the following which is a corollary of [4]:

Proposition 1 *The model-checking of a $L_\mu[\prec_{\mathrm{fc}}, \prec_{\mathrm{sb}}]$ formula over trees can be done in time $O(||\varphi||^2 \cdot ||T||)$.*

Binary trees A *binary tree* is a tree where each element s has exactly two children $s \cdot 0$ and $s \cdot 1$, unless s is a leaf. Therefore, binary trees can be represented as structures

$$T \;=\; (D, \prec, <_0, <_1, (P_a)_{a \in \Sigma})\,,$$

where $\prec$ is the usual prefix relation on strings, $s <_0 s'$ iff $s' = s \cdot 0$, and $s <_1 s'$ iff $s' = s \cdot 1$. Then,

Theorem 2 *[6] The following holds for Boolean queries over binary trees,*

$$L_\mu[<_0, <_1] \;=\; \mathrm{FO}[\prec, <_0, <_1]\,.$$

2 FO over Ordered Trees and Conditional XPath

Core XPath is defined by the following grammar:

$$\begin{array}{rcl}
\text{axes} & := & \text{self} \mid \text{child} \mid \text{desc} \mid \text{nextsib} \mid \text{sib} \\
\text{node tests } \alpha & := & a \ (a \in \Sigma) \mid * \mid \neg\alpha \mid \alpha \vee \alpha \mid [\beta] \\
\text{location paths } \beta & := & \text{axes} \mid \text{axes}^- \mid ?\alpha \mid \beta \vee \beta \mid \beta \circ \beta
\end{array}$$

Semantics for node tests is given in nodes of the tree as follows (without easy Boolean combinations):

- $(T, s) \models a$ iff s is labeled a in T.
- $(T, s) \models *$ for any s.
- $(T, s) \models [\beta]$ iff there is s' such that $(T, s, s') \models \beta$

Semantics for location paths is given by pairs. For instance, $(T, s, s') \models \text{child}^-$ iff $s' \prec_{\text{ch}} s$, $(T, s, s') \models \text{sib}$ iff $s \prec_{\text{sb}}^* s'$, and $(T, s, s') \models \text{self}$ iff $s = s'$. Furthermore,

- $(T, s, s') \models ?\alpha$ iff $(T, s) \models \alpha$ and $s = s'$.
- $(T, s, s') \models \beta \vee \beta'$ iff $(T, s, s') \models \beta$ or $(T, s, s') \models \beta'$
- $(T, s, s') \models \beta \circ \beta'$ iff there is s'' such that $(T, s, s'') \models \beta$ and $(T, s'', s') \models \beta'$

A node test α defines the query Q_α such that $Q_\alpha(T) = \{s \mid (T, s) \models \alpha\}$, while a location path β defines the query Q_β such that $Q_\beta(T) = \{(s, s') \mid (T, s, s') \models \beta\}$.

We say that a language $\mathcal{L}$ is *closed under path complementation*, if for every location path β in $\mathcal{L}$ there is a location path β' in $\mathcal{L}$ such that for every tree T,

$$Q_{\beta'}(T) \; = \; \{(s, s') \mid (s, s') \in T, \; (s, s') \notin Q_\beta(T)\}\,.$$

Theorem 3 *[5] The following hold:*

1. *Core XPath is not closed under path complementation.*

2. *Any extension of Core XPath that is closed under path complementation has exactly the expressive power of unary* $\text{FO}[\prec_{\text{ch}}^*, \prec_{\text{sb}}^*]$ *on node tests, and exactly the expressive power of binary* $\text{FO}[\prec_{\text{ch}}^*, \prec_{\text{sb}}^*]$ *on location paths.*

Of course, adding formula $\neg\beta$ to Core XPath would make the language closed under path complementation. However, Marx in [5] suggests that this is not a very intuitive operator. A more intuitive one in his oppinion are *conditional axis* operators as follows.

Conditional XPath is the extension of Core XPath with the location path formula $(\text{axis}/?\alpha)^+$, such that:

- $(T, s, s') \models (\text{axis}/?\alpha)^+$ iff there is a path $s_0, s_1, \ldots, s_k$ such that $s = s_0$, $s' = s_k$, $(T, s_i, s_{i+1}) \models$ axis for each $i < k$, and $(T, s') \models \alpha$.

This is enough, as

Theorem 4 *[5] Conditional XPath is closed under path complementation.*

From Theorem 3 we obtain that the node tests of Conditional XPath have precisely the power of unary $\text{FO}[\prec_{\text{ch}}^*, \prec_{\text{sb}}^*]$, and the location paths of Conditional XPath have precisely the power of binary $\text{FO}[\prec_{\text{ch}}^*, \prec_{\text{sb}}^*]$.

3 FO over Ordered Trees and $\mathrm{CTL}^\star_{\mathrm{past}}$

In this section we look at the full vocabulary containing both $\prec^*_{\mathrm{ch}}$ and $\prec^*_{\mathrm{sb}}$. We show that these are captured by $\mathrm{CTL}^\star$ if we add the ability to reason about the past.

We define $\mathrm{CTL}^\star$ in a way that is convenient when we have several binary relations, say $E_1, \ldots, E_m$. Then $\mathrm{CTL}^\star[E_1, \ldots, E_m]$ has *state* formulae α, and E_i*-path* formulae β_i, $i \le m$, defined by the grammars below:

$$\begin{array}{rcl} \alpha & := & a\ (a \in \Sigma)\ \mid\ \neg\alpha\ \mid\ \alpha \vee \alpha\ \mid\ \mathbf{E}\beta_i,\ i \le m \\ \beta_i & := & \alpha\ \mid \neg\beta_i\ \mid\ \beta_i \vee \beta_i\ \mid\ \mathbf{X}_{E_i}\beta_i\ \mid\ \beta_i\, \mathbf{U}_{E_i}\, \beta_i \end{array}$$

Of course for the case of just one binary relation this is the standard definition of $\mathrm{CTL}^\star$. A state formula is evaluated on a state of the transition system, while an E_i-path formula is evaluated on an E_i-path of the transition system.

An E_i-path π is a sequence $s_1 s_2 \ldots$ of nodes such that $(s_j, s_{j+1}) \in E_i$ for every s_j, s_{j+1} in π, and such that if the set $\{s \mid (s_j, s) \in E_i\}$ is nonempty, then one of the elements of this set is s_{j+1}. (Note that in trees, both $\prec_{\mathrm{ch}}$-paths and $\prec_{\mathrm{sb}}$-paths will be finite, although typically in transition systems one considers infinite paths. This is not a problem, however, since we can make all paths infinite by adding a child labeled $\perp \notin \Sigma$ to each leaf, and a $\prec_{\mathrm{ch}}$ loop for that element, and likewise for the youngest sibling on each sibling path.)

Given an E_i-path $\pi = s_1 s_2 \ldots$, we let π^k be the path starting at s_k. Then (we only list the essential rules):

- $(\mathcal{K}, s) \models \mathbf{E}\beta_i$ iff there exists an E_i-path $\pi = s\ldots$ such that $(\mathcal{K}, \pi) \models \beta_i$;
- $(\mathcal{K}, \pi) \models \alpha$ iff $(\mathcal{K}, s_1) \models \alpha$;
- $(\mathcal{K}, \pi) \models \mathbf{X}_{E_i}\beta_i$ iff $\pi = s_1 s_2 \ldots$ is an E_i-path and and $(\mathcal{K}, \pi^2) \models \beta_i$; and
- $(\mathcal{K}, \pi) \models \beta_i\ \mathbf{U}_{E_i}\ \beta'_i$ iff π is an E_i-path and there is a number k such that $(\mathcal{K}, \pi^k) \models \beta'_i$ and $(\mathcal{K}, \pi^l) \models \beta_i$ for all $l < k$.

To capture FO over $\prec^*_{\mathrm{ch}}$ and $\prec^*_{\mathrm{sb}}$, we shall use an extension $\mathrm{CTL}^\star_{\mathrm{past}}$ of $\mathrm{CTL}^\star$ that allows reasoning about the past. Normally such a logic is defined by allowing in addition to $\mathbf{X}$ and $\mathbf{U}$ their "inverses" usually called $\mathbf{Y}$ (yesterday) and $\mathbf{S}$ (since) [3]. We shall use the notation $\mathbf{X}_{\prec^-_{\mathrm{ch}}}$ and $\mathbf{X}_{\prec^-_{\mathrm{sb}}}$ instead of $\mathbf{Y}$, referring to them as next with respect to inverses of $\prec^-_{\mathrm{ch}}$ and $\prec^-_{\mathrm{sb}}$ of $\prec_{\mathrm{ch}}$ and $\prec_{\mathrm{sb}}$. In general, the semantics of path formulae of $\mathrm{CTL}^\star_{\mathrm{past}}$ refers to a path and a position in a path; that is, one defines the notion $(\mathcal{K}, \pi, \ell) \models \beta$. A path therefore includes not only the future but also the past, and we require that paths include the entire past. That is, all $\prec_{\mathrm{ch}}$-paths start in the root, and all $\prec_{\mathrm{sb}}$-paths start in the oldest child.

The semantics of $\mathrm{CTL}^\star_{\mathrm{past}}[E_1, \ldots, E_r]$ is as follows [3] (again, listing only the essential rules):

- $(\mathcal{K}, s) \models \mathbf{E}\beta_i$ if there is an E_i-path $\pi = s_1 s_2 \ldots$ and $\ell \ge 1$ such that $s = s_\ell$ and $(\mathcal{K}, \pi, \ell) \models \beta_i$;
- $(\mathcal{K}, \pi, \ell) \models \mathbf{X}_{E_i^-}\beta_i$ iff π is an E_i-path, $\ell > 1$ and $(\mathcal{K}, \pi, \ell - 1) \models \beta_i$;
- $(\mathcal{K}, \pi, \ell) \models \beta_i\ \mathbf{S}_{E_i}\ \beta'_i$ iff π is an E_i-path and there exists $k < \ell$ such that $(\mathcal{K}, \pi, k) \models \beta'_i$ and $(\mathcal{K}, \pi, j) \models \beta_i$ for all $k < j \le \ell$.

One can also define a version of $\mathrm{CTL}^\star_{\mathrm{past}}$ with one "until" and "since" operator and several "next" and "previous" operators. We shall denote this logic by $\mathrm{CTL}^\star_{\mathrm{past}}[\prec_{\mathrm{ch}} \cup \prec_{\mathrm{sb}}]$. The semantics naturally combines the semantics of past and the the semantics of $\mathrm{CTL}^\star[\bigcup_i E_i]$ (that is, we have

$\prec_{\text{ch}} \cup \prec_{\text{sb}}$ paths). For example, $(T, \pi, \ell) \models \mathbf{X}_{\prec_{\text{ch}}}\varphi$ if $(T, \pi, \ell + 1) \models \varphi$ and from position ℓ to position $\ell + 1$ one goes by the child relation.

It is known that over a Kripke structure (or a transition system with a single binary relation), $\text{CTL}^\star_{\text{past}} = \text{CTL}^\star$ if each state has a unique path that leads from an initial state to it [3]. But unranked trees are modeled as transition systems with two binary relations, and over the union of these relations, there may be more than one history. In fact one can easily show that over unranked trees, $\text{CTL}^\star \subsetneq \text{CTL}^\star_{\text{past}}$ (for example, a formula saying that the root's oldest child is labeled b and one other child is labeled a is not expressible in $\text{CTL}^\star$ as can be shown by a simple game argument).

Theorem 5 *[1] The equivalences*

$$\text{FO}[\prec_{\text{ch}}, \prec_{\text{sb}}] \;=\; \text{CTL}^\star_{\text{past}}[\prec_{\text{ch}}, \prec_{\text{sb}}] \;= \text{CTL}^\star_{\text{past}}[\prec_{\text{ch}} \cup \prec_{\text{sb}}] \;=\; \text{CTL}^\star[\prec_{\text{fc}}, \prec_{\text{ch}}, \prec_{\text{sb}}]$$

hold for unary queries over unranked trees. Furthermore, all translations between these formalisms are effective.

By inspecting the proof of the previous theorem, we can get the following characterization of Boolean FO over ordered trees.

Theorem 6 *[1] For Boolean queries,*

$$\text{FO}[\prec^*_{\text{ch}}, \prec^*_{\text{sb}}] \;=\; \text{CTL}^\star_{\text{past}}[\prec_{\text{ch}}, \prec_{\text{sb}}] \;=\; \text{CTL}^\star_{\text{past}}[\prec_{\text{ch}} \cup \prec_{\text{sb}}] \;=\; \text{CTL}^\star[\prec_{\text{fc}} \cup \prec_{\text{sb}}]$$

hold over unranked trees. Moreover, every formula of $\text{CTL}^\star_{\text{past}}[\prec_{\text{ch}}, \prec_{\text{sb}}]$ *is equivalent to a formula that does not use past operators* $\mathbf{S}_{\prec_{\text{ch}}}$ *and* $\mathbf{X}_{\prec^-_{\text{ch}}}$ *but uses* $\mathbf{X}_{\prec^-_{\text{sb}}}$ *and* $\mathbf{S}_{\prec_{\text{sb}}}$*, and the translations between these logics are effective.*

By combining results in the previous section with the ones presented here, we can see a very interesting fact: over unranked trees languages originally designed for querying XML data (XPath) are essentially the same than languages designed in a totally different context, that of verification of formal systems ($\text{CTL}^\star$).

Regarding model checking of these logics we know the following. First, the model checking of either $\text{CTL}^\star_{\text{past}}[\prec_{\text{ch}}, \prec_{\text{sb}}]$ or $\text{CTL}^\star[\prec_{\text{fc}}, \prec_{\text{ch}}, \prec_{\text{sb}}]$ is polynomial in both the size of the tree and size of the formula. However, exact bounds still have to be determined. The model checking of $\text{CTL}^\star_{\text{past}}[\prec_{\text{ch}} \cup \prec_{\text{sb}}]$ is $O(2^{||\varphi||} \cdot ||T||)$, which matches the complexity of $\text{CTL}^\star$ over arbitrary transition systems.

Binary trees From [2] we know the following,

Theorem 7 *[2] For Boolean queries over binary trees,*

$$\text{CTL}^\star[<_0 \cup <_1] \;=\; \text{FO}[\prec, <_0, <_1]\,.$$

References

[1] P. Barceló, and L. Libkin. Temporal Logics over Unranked Trees. In *LICS'05*, pages 31–40.

[2] T. Hafer, W. Thomas. Computation tree logic CTL* and path quantifiers in the monadic theory of the binary tree. In *ICALP'87*, 269–279, 1987.

[3] O. Kupferman, and A. Pnueli. Once and for all. In *LICS'95*, pages 25–35.

[4] R. Mateescu. Local model-checking of modal mu-calculus on acyclic labeled transition systems. In *TACAS 2002*, pages 281–295.

[5] M. Marx. Conditional XPath. TODS, to appear.

[6] D. Niwinski. Fixed points vs. infinite generation. In *LICS 1988*, pages 402–409.

[7] M. Y. Vardi. Reasoning about the past with two-way automata. In *ICALP 1998*, pages 628–641.