Monadic Second order logic on trees

(n-ary ordered) Tree domain T is a subset of $\{0,1,2\ldots n-1\}^*$ that is

- prefix closed.
- $\forall u \in \{0,\ldots n-1\}^*$, $i,j \in \{0,\ldots n-1\}$ and $i<j$ if $uj \in T$ then $ui \in T$.

Σ-labeled n-ary trees are $T = (dom_t, lab_t)$ such that dom_t is a tree domain and $lab_t : dom_t \to \Sigma$ is a labeling function.

Tree Automaton $\mathcal{A} = (Q, \Sigma, \delta, F)$

- $\delta = \bigcup_{i=0}^{n-1} \delta_i$

$$\delta_i : Q^i \times \Sigma \to 2^Q$$

Process labeled trees starting at the leaf and going up the tree until it reaches the root.

An input tree t is accepted if $\mathcal{A}$ has a run where the root is labeled by a final state.

Tree Regular Languages A set of Σ-labeled (n-ary) trees such that the set is recognized by some tree automaton.

Properties of Tree Regular Languages

- Deterministic Tree automata is as powerful as a nondeterministic tree automaton.
- Tree regular languages are closed under union, intersection and complementation.
- Other classical closure properties of word regular languages extend to tree languages.
- There are non-regular tree languages.

$$L = \{ A(t_1, t_2) \mid t_1 = t_2 \}$$

Tree with root labeled A and has 2 children which are trees t_1 and t_2

Left & right subtrees of root are isomorphic.

L is not a regular tree language.

Proof Suppose L is recognized by a deterministic tree automaton B with k states.

There must be trees $t_1 \neq t_2$ such that in the run of B on t_1 and t_2 the state labeling the root is the same.

$A(t_1, t_1) \in L$ (and $A(t_2, t_2) \in L$)
but $A(t_1, t_2) \notin L$.

The run of B on $A(t_1, t_1)$ and $A(t_1, t_2)$ will be in the same state at the Root.

Proposition Given a tree automaton $\mathcal{A}$ the problem of determing if $L(\mathcal{A}) = \emptyset$ and the problem of determing if $L(\mathcal{A})$ is "universal" are ~~decidable.~~

If $\mathcal{A}$ is deterministic decidable in linear time.

Σ-labeled Trees as structures

Signature $(\underbrace{<, S_0, S_1 \ldots S_{n+1}}_{\text{binary}}, \{Q_a\}_{a \in \Sigma})$

$<$: ancestor/ relation

$S_0, S_1 \ldots S_{n+1}$: parent - child relation

$\{Q_a\}$: unary; labels on nodes

Tree as a structure $t = (dom_t, lab_t)$ is a structure $\mathcal{T} = (T, <^T, S_i^T, Q_a^T)$

- $T = dom_t$
- $(u, v) \in <^T$ if u is a strict prefix of v
- $S_i = \{(u, ui) \mid u, ui \in T\}$

$dom_t = \{\epsilon, 0, 1, 10, 100, 101\}$

$- Q_a = \{ u \in T \mid lab_t(u) = a \}$.

$T = \{ \epsilon, 0, 1, 10, 100, 101 \}$

$<^T = \{ (\epsilon, 0), (\epsilon, 1), (\epsilon, 10) \ldots \}$

$S_0 = \{ (\epsilon, 0), (1, 10), (10, 100) \}$

$S_1 = \{ (\epsilon, 1), (10, 101) \}$

$Q_\wedge = \{ \epsilon, 10 \}$. $Q_\neg = \{ 1 \}$ . . .

Monadic Second Order Logic on Trees.

Second order logic formulas over the signature $(<, S_0 \ldots S_{n-1}, \{Q_a\}_{a \in \Sigma})$.
Such that all relational variables have arity 1.

Terms : $t ::= x$ — variables.

Formulas $\varphi ::= t_1 = t_2 \mid <(t_1, t_2) \mid S_i(t_1, t_2)$

$Q_a t \mid X t$

Relational variable

$\neg\varphi \mid \varphi \vee \varphi$

arity 1

$\exists x . \varphi \mid \exists X \varphi$

For a sentence φ

$[\![\varphi]\!] = \{ T \text{ a tree} \mid T \models \varphi \}$.

Trees defined by φ.

Doner, Thatcher-Wright A tree language

L is regular iff it is definable in MSO

i.e $\exists$ MSO sentence φ s.t

$$L = [\![\varphi]\!]$$

Proof ($\Longrightarrow$) Let L that is recognized by tree automaton $\mathcal{A} = (\{0, \dots k-1\}, \Sigma, \delta, F)$.

Sentence defing L is going say that the automaton $\mathcal{A}$ has an accepting run on input tree.

Run is a tree labeled by $\{0, \dots k-1\}$

- For each state p we will identify the vertices that have label p in the accepting run.

vertices in state 1 in the accepting run.

$$\varphi_{\mathcal{A}} = \exists X_0 \exists X_1 \dots \exists X_{k-1}$$

$$\forall x \left(\bigvee_{i=0}^{k-1} X_i x \wedge \bigwedge_{i \neq j} \neg (X_i x \wedge X_j x) \right)$$

every vertex has a unique state label.

$$\wedge \ \forall x \left(root(x) \longrightarrow \bigvee_{i \in F} X_i x \right)$$

root is labeled by a final state.

$\rightarrow \mathrm{root}(x) = \forall y \bigwedge_{i=0} \neg S_i(y, x)$

$$\wedge \bigwedge_{i=0}^{n-1} \forall x \forall y_0 \ldots y_{i-1}$$

$$\mathrm{children}_i(x, y_0 \ldots y_{i-1}) \rightarrow$$

$$\bigvee_{p \in \delta(p_0 \ldots p_{i-1}, a)} \left(Q_a(x) \wedge \bigwedge_{j=0}^{i-1} X_{p_j} y_j \wedge X_p x \right)$$

Labeling of vertices by states is consistent with the transition fn.

defined.

This a formula of the form

$$\exists X \ \forall y \ \varphi.$$