

1 AUTOMATA ON FINITE TREES

2 REGULAR LANGUAGES OF FINITE TREES

RANKED TREES

- rank: maximal arity of nodes (number of children)
- binary trees: rank ≤ 2 .
- **Finite** binary tree over alphabet Σ : partial mapping

$$t : \{0, 1\}^* \rightarrow \Sigma$$

such that $\text{dom}(t)$ is finite, prefix-closed and $x1 \in \text{dom}(t)$ implies $x0 \in \text{dom}(t)$, for all x .

ϵ : root, 0, 1: children of the root, etc.

BOTTOM-UP TREE AUTOMATA (NTA, TA)

An NTA (non-deterministic tree automaton) $\mathcal{A} = \langle S, \Sigma, \delta, F \rangle$ consists of

- a finite set of states S ,
- a finite alphabet Σ ,
- a transition relation $\delta \subseteq S^{\leq 2} \times \Sigma \times S$ (we assume binary trees),
- a set of final states F .

An NTA is **deterministic** (denoted TA), if δ is a function $\delta : S^{\leq 2} \times \Sigma \rightarrow S$.

NTA ACCEPTANCE

- Let $\mathcal{A} = \langle S, \Sigma, \delta, F \rangle$ be an NTA. A run of \mathcal{A} on a binary tree t is a (partial) mapping

$$\rho : \{0, 1\}^* \rightarrow S$$

with $\text{dom}(\rho) = \text{dom}(t)$ and

- 1 if $x \in \text{dom}(t)$ of rank 0 (leaf) then $(t(x), \rho(x)) \in \delta$,
 - 2 if $x \in \text{dom}(t)$ of rank 1 then $(\rho(x0), t(x), \rho(x)) \in \delta$,
 - 3 if $x \in \text{dom}(t)$ of rank 2 then $(\rho(x0), \rho(x1), t(x), \rho(x)) \in \delta$.
- A run ρ of \mathcal{A} on t is **accepted** if $\rho(\epsilon) \in F$.
- Accepted language** $L(\mathcal{A}) =$ set of all finite binary trees having an accepting run of \mathcal{A} .

REGULAR LANGUAGES

A language of binary trees is **regular** if it is accepted by an NTA.

PROPERTIES OF NTA

- NTA can be determinized.
- The family of regular languages of (binary) finite trees is a Boolean algebra.

TOP-DOWN TREE AUTOMATA

- A top-down tree automaton $\mathcal{A} = \langle S, \Sigma, I, \delta \rangle$ consists of a finite set of states S , a finite alphabet Σ , a set of initial states I and a transition relation

$$\delta \subseteq S \times \Sigma \times S^{\leq 2}$$

- Accepting run ρ : $\rho(\epsilon) \in I$ and $(\rho(x), t(x)) \in \delta$ for each leaf x .
- Deterministic top-down tree automaton: $\delta : S \times \Sigma \rightarrow S^{\leq 2}$.
- Non-deterministic tree top-down automata accept precisely regular tree languages.
- Top-down deterministic tree automata are strictly weaker than non-deterministic ones. Example: $L = \{c(a, b), c(b, a)\}$.

1 AUTOMATA ON FINITE TREES

2 REGULAR LANGUAGES OF FINITE TREES

PRODUCT, ITERATION

- Let L, K be tree languages and $a \in \Sigma$. We define $L \cdot_a K$ as the set of trees of L where every a -labeled leaf is replaced by some element of K .
- Let L be a tree language and $a \in \Sigma$. We define inductively (for $k \geq 0$) :

$$L^{0,a} = \{a\}, \quad L^{k+1,a} = L^{k,a} \cdot_a L$$

- $L^{*,a} = \cup_{k \geq 0} L^{k,a}$

REGULAR EXPRESSIONS

The set of **regular expressions** over trees is defined as the least set containing \emptyset and t (for every tree t), which is closed under union, \cdot_a and *,a (for all $a \in \Sigma$).

KLEENE-LIKE THEOREM

A language of trees is regular if and only if it is defined by a regular expression.

THATCHER & WRIGHT, DONER

A language of finite trees is regular if and only if it is definable in MSOL.

ANTICHAIN LOGIC (POTTHOFF, THOMAS)

- A subset of nodes in a tree is an anti-chain if its elements are pairwise incomparable w.r.t. the descendant order.
- Anti-chain MSOL is the restriction of MSOL to set quantification over anti-chains.
- A language of proper binary, finite trees (without unary branching) is regular if and only if it is definable in anti-chain MSOL.