Reduction of Tree Automata

LC

June 22, 2020

Abstract

1 Introduction

2 Preliminaries

Let $\mathbb{N}$ be the set of non-negative integers. A *node* is an element $u \in \mathbb{N}^*$. A node u is a *child* of a node v if $u = v \cdot i$ for some $i \in \mathbb{N}$. A *tree domain* is a non-empty prefix-closed set of nodes $D \subseteq \mathbb{N}^*$ s.t., if $u \cdot (i+1) \in D$, then $u \cdot i \in D$ for every $i \in \mathbb{N}$. A *leaf* is a node u in D without children. A *ranked alphabet* is a family of symbols $\Sigma = (\Sigma_k)_{k \in \mathbb{N}}$, where symbols in Σ_k have rank k. A Σ-*tree* is a function $t : D \to \Sigma$, where D is a tree domain, s.t., for every node u in D labelled with a symbol $t(u) \in \Sigma_k$ of rank k, u has precisely k children.

A (finite, nondeterministic, top-down) *tree automaton* is a tuple $\mathcal{A} = \langle \Sigma, Q, I, \longrightarrow \rangle$ where Σ is a ranked alphabet, $Q = (Q_k)_{k \in \mathbb{N}}$ is a finite family of states, of which those in $I \subseteq Q$ are called *initial states*, and $\longrightarrow = (\xrightarrow{a})_{a \in \Sigma}$ is a finite family of transitions s.t. $\xrightarrow{a} \subseteq Q_k \times Q^k$ whenever $a \in \Sigma_k$. Thus, a state $q \in Q_k$ only reads symbols of rank k. Let $q \in Q$ be a state, and let $t : D \to \Sigma$ be an input tree. A *run tree from q on t* is a Q-tree $r : D \to Q$ over the same tree domain D s.t. $r(\varepsilon) = q$ and, for every node $u \in D$, if $t(u) \in \Sigma_k$, then $r(u) \xrightarrow{t(u)} r(u \cdot 0) \cdots r(u \cdot (k-1))$. Whenever such a run tree exists, we write $q \xrightarrow{t}$. The *downward language* recognized by a state q, denoted $\mathcal{L}^{\downarrow}(q)$, is the set of trees t s.t $q \xrightarrow{t}$. The language recognizes by an automaton is $\mathcal{L}(\mathcal{A}) = \bigcup_{q \in I} \mathcal{L}^{\downarrow}(q)$.

A *spine* is a string $w \in (\bigcup_k \Sigma_k \times \{0, \ldots, k-1\} \times Q^{k-1})^*$. We define $p \xrightarrow{\varepsilon} p$ and $p \xrightarrow{(a,i,\bar{q}) \cdot w} r$ with $\bar{q} = q_0 \cdots q_{i-1} q_{i+1} \cdots q_{k-1}$ whenever there exists a transition $p \xrightarrow{a} q_0 \cdots q_{k-1}$ s.t. $q_i \xrightarrow{w} r$.

3 Language inclusion

We study preorders which can be used to establish language inclusion.

4 Saturation

We study preorders which can be used to add transitions while preserving the language. This can also be used to show that the induced equivalences are GFQ.

Downward containment transformer $\mathcal{T}^{\downarrow} : 2^{Q\times Q} \mapsto 2^{Q\times Q}$ takes a binary relation on states U, and yields another binary relation on states $\mathcal{T}^{\downarrow}(U)$ s.t., for every states $p,q \in Q$, $p\ \mathcal{T}^{\downarrow}(U)\ q$ holds if, and only if, for every Σ-tree t and every U-jumping t-run rooted at p, there exists a U-jumping t-run rooted at q.

Upward containment transformer $\mathcal{T}^{\uparrow} : 2^{Q\times Q} \times 2^{Q\times Q} \mapsto 2^{Q\times Q}$ takes two binary relation on states U and D, and yields another binary relation on states $\mathcal{T}^{\uparrow}(U,D)$ s.t., for every states $p,q \in Q$, $p\ \mathcal{T}^{\uparrow}(U,D)\ q$ holds if, and only if, for every spine $w = (a_0,i_0,\bar{p}_0)\cdots(a_n,i_n,\bar{p}_n)$ and initial state p' s.t. $p' \xrightarrow{w} p$, there exists a spine $w' = (a_0,i_0,\bar{q}_0)\cdots(a_n,i_n,\bar{q}_n)$ and an initial state q' s.t. $q' \xrightarrow{w'} q$ and $\bar{p}_j\ (D \cap U^{-1})\ \bar{q}_j$ for every $0 \le j \le n$.

Let $\subseteq_0^{\downarrow} = \subseteq_0^{\uparrow} = id$, and, for every $i \geq 0$, let

$$\subseteq_{i+1}^{\downarrow} = \mathcal{T}^{\downarrow}(\subseteq_i^{\uparrow})$$
$$\subseteq_{i+1}^{\uparrow} = \mathcal{T}^{\uparrow}(\subseteq_i^{\uparrow}, \subseteq_i^{\downarrow})$$

Finally, define $\subseteq^{\downarrow} = \bigcup_i \subseteq_i^{\downarrow}$ and $\subseteq^{\uparrow} = \bigcup_i \subseteq_i^{\uparrow}$.