Acta Cybernetica, Vol. 10, Nr. 4, Szeged, 1992

The Boolean Closure of DR-Recognizable Tree Languages

E. Jurvanen*

Abstract

The family *DRec* of tree languages recognized by deterministic root-to-frontier (top-down) tree automata is not closed under unions or complements. Hence, it is not a variety of tree languages in the sense of Steinby. However, we show that the Boolean closure of *DRec* is a variety which is properly included in the variety *Rec* of all recognizable tree languages. This Boolean closure is also compared with some other tree language varieties.

1 Introduction

Finite tree recognizers are divided into four types according to whether they are deterministic or not, and whether they read trees from root to frontier or from frontier to root. The nondeterministic tree automata and the deterministic frontier-to-root tree automata recognize the same class of tree languages. This is the class of recognizable tree languages which is here denoted by *Rec*. However, the deterministic root-to-frontier tree automata recognize a proper subclass of *Rec* called here *DRec*. These tree automata types were defined and the connections between the languages they recognize were established in the late sixties by Thatcher and Wright [TW68], Rabin [Rab69], Doner [Don70], Magidor and Moran [MM69].

The class *DRec* has been studied relatively little. Courcelle [Cou78a,Cou78b] and Virágh [Vir80] gave a characterization using a path closure operator. Gécseg and Steinby [GS78] presented an algorithm for minimizing deterministic root-to-frontier tree automata.

In this paper we study the Boolean closure of *DRec* denoted here by $\mathcal{B}(DRec)$. It is shown to form a variety in the sense of Steinby [Ste79,Ste92]. Since also *Rec* is a variety, the next question is, whether variety $\mathcal{B}(DRec)$ is properly included in variety *Rec*. In connection with his studies of logic characterizations of tree language families, Thomas [Tho84] answered this question positively; $\mathcal{B}(DRec)$ is a proper subclass of the chain definable tree languages which form a proper subclass of *Rec*. In this work we also prove the proper inclusion of $\mathcal{B}(DRec)$ in *Rec*, but directly using only the pidgeon hole principle. After that $\mathcal{B}(DRec)$ is compared with respect to the inclusion relation with the varieties *Nil*, *D*, *RD*, *GD* and *Loc*, where *Nil* is the Boolean closure of the family of finite tree languages, and the others consist of the definite, the reverse definite, the generalized definite and the local tree languages, respectively. Some of the definitions of these tree families were

*Department of Mathematics, University of Turku, SF-20500 Turku, Finland, E-mail: jurvanen@utu.fi

given by Heuter [Heu88,Heu89a,Heu89b] and they were shown to be varieties by Steinby [Ste92].

The notation is mostly from [GS84].

2 Preliminaries

For a set A, we denote by pA the power set of A, that is the set of all subsets of A, and by $|A|$ the cardinality of A. If $A \subseteq B$, but $A \neq B$, then we write $A \subset B$.

Let N be the set of natural numbers, $N = \{0, 1, \ldots\}$. A *ranked alphabet* Σ is a finite set of operation symbols each of which has been assigned a unique rank from N. For $m \in N$, the set of m-ary operation symbols of Σ form a set denoted by Σ_m. Thus $\Sigma = \bigcup_{m \in N} \Sigma_m$. Two special cases are the *trivial* ranked alphabets, for which $\Sigma = \Sigma_0$, and the *unary* ranked alphabets satisfying $\Sigma = \Sigma_0 \cup \Sigma_1$.

In a Σ-*algebra* $\mathcal{A} = (A, \Sigma)$, A is a nonempty set, Σ is a set of operation symbols and every operation symbol $\sigma \in \Sigma_m$, where $m \geq 1$, is interpreted as a mapping

$$\sigma^{\mathcal{A}} : A^m \longrightarrow A,$$

and every nullary symbol $\sigma \in \Sigma_0$ is interpreted as an element $\sigma^{\mathcal{A}}$ of A. If $\mathcal{A} = (A, \Sigma)$ and $\mathcal{B} = (B, \Sigma)$ are two Σ-algebras, then a *homomorphism* from $\mathcal{A}$ to $\mathcal{B}$ is a mapping $\phi : A \longrightarrow B$ such that

$$\sigma^{\mathcal{A}}(a_1, \ldots, a_m)\phi = \sigma^{\mathcal{B}}(a_1\phi, \ldots, a_m\phi)$$

holds for all $m \geq 0$, $\sigma \in \Sigma_m$ and $a_1, \ldots, a_m \in A$. In particular, if $\sigma \in \Sigma_0$ and ϕ is a homomorphism, then $\sigma^{\mathcal{A}}\phi = \sigma^{\mathcal{B}}$. An equivalence relation θ on A is a *congruence* of $\mathcal{A}$, if for all $m \geq 0, \sigma \in \Sigma_m$ and $a_1, \ldots, a_m, b_1, \ldots, b_m \in A$,

$$a_1\theta b_1, \ldots, a_m\theta b_m \quad \text{implies} \quad \sigma^{\mathcal{A}}(a_1, \ldots, a_m)\theta\sigma^{\mathcal{A}}(b_1, \ldots, b_m).$$

An equivalence class of a congruence is called a *congruence class* and the congruence class of $a \in A$ is denoted by $a\theta$. A congruence of $\mathcal{A}$ is said to *saturate* a subset $L \subseteq A$, if $L = L\theta$. This means that L is the union of some congruence classes of θ. If a congruence has finitely many congruence classes, then the congruence is *finite*.

Let X be an alphabet, that is a finite set of letters, such that $\Sigma \cap X = \emptyset$. We assume also that $X \cup \Sigma_0 \neq \emptyset$. The set of all ΣX-*trees* is the smallest set containing every $x \in X, \sigma \in \Sigma_0$ and $\sigma(t_1, \ldots, t_m)$, where $m \geq 1, \sigma \in \Sigma_m$ and $t_1, \ldots, t_m$ are ΣX-trees. A set of ΣX-trees is called a ΣX-*forest* or a ΣX-*tree language*. The set of all ΣX-trees is denoted by $F_\Sigma(X)$. The *complement* of a ΣX- forest T is $T^C = F_\Sigma(X) \setminus T$. The height, root, subtrees and leaves of a tree t are denoted by $\mathrm{hg}(t)$, $\mathrm{root}(t)$, $\mathrm{sub}(t)$ and $\mathrm{leaf}(t)$ respectively. As usual, if $t \in X \cup \Sigma_0$, then $\mathrm{hg}(t) = 0$, $\mathrm{root}(t) = t$ and $\mathrm{sub}(t) = \{t\}$. For $t = \sigma(t_1, \ldots, t_m)$, where $m > 0$, $\mathrm{hg}(t) = 1 + \max_{1 \leq i \leq m} \mathrm{hg}(t_i)$, $\mathrm{root}(t) = \sigma$ and $\mathrm{sub}(t) = \{t\} \cup \bigcup_{1 \leq i \leq m} \mathrm{sub}(t_i)$. The leaves of any tree are its subtrees of height 0.

Let ξ be a letter not in $X \cup \Sigma$. A tree $p \in F_\Sigma(X \cup \{\xi\})$ is a *special tree*, if ξ occurs in it exactly once. The set of all special trees is denoted by $Sp_\Sigma(X)$. The *product* of a special tree $p \in F_\Sigma(X \cup \{\xi\})$ and a tree $t \in F_\Sigma(X)$ is a tree $t \cdot_\xi p \in F_\Sigma(X)$, which is formed from p by substituting t for its leaf ξ. When $p \in Sp_\Sigma(X)$ and $T \subseteq F_\Sigma(X)$, the p-*translation* of T is

$$p(T) = \{t \cdot_\xi p \mid t \in T\}$$

and the *inverse p-translation* of T is

$$p^{-1}(T) = \{t \in F_\Sigma(X) \mid t \cdot_\xi p \in T\}.$$

The ΣX-trees form a Σ-algebra $\mathcal{F}_\Sigma(X) = (F_\Sigma(X), \Sigma)$ with operations from Σ defined as

$$\sigma^{\mathcal{F}_\Sigma(X)}(t_1, \ldots, t_m) = \sigma(t_1, \ldots, t_m),$$

where $m \geq 0, t_1, \ldots, t_m \in F_\Sigma(X)$ and $\sigma \in \Sigma_m$. This Σ-algebra is called the ΣX-*term algebra.*

A set of trees which can be recognized by a frontier-to-root or a nondeterministic root-to-frontier recognizer [GS84] is called *recognizable*. The set of all recognizable ΣX-tree languages we denote by $Rec(\Sigma, X)$.

A *deterministic root-to-frontier* Σ-*algebra* (a DR Σ-algebra) is a pair $\mathcal{A} = (A, \Sigma)$, where A is a nonempty set and every operation symbol $\sigma \in \Sigma_m$ with $m > 0$ is interpreted as a mapping

$$\sigma^{\mathcal{A}} : A \to A^m.$$

If $\sigma \in \Sigma_0$, then it defines a singleton $\sigma^{\mathcal{A}}$ in A. An algebra $\mathcal{A} = (A, \Sigma)$ is called *finite,* if the set A is finite.

Let X be an alphabet. A *deterministic root-to-frontier* ΣX-*recognizer* (a DR ΣX-recognizer) is a triple $\mathbf{A} = (\mathcal{A}, a_0, \alpha)$, where

(1) $\mathcal{A}$ is a finite DR Σ-algebra $\mathcal{A} = (A, \Sigma)$,
(2) $a_0 \in A$ is the *initial state* and
(3) $\alpha : X \to pA$ is the *final assignment.*

The recognizer is also denoted by $\mathbf{A} = (A, \Sigma, X, a_0, \alpha)$. The elements of the set A are called the *states* of the recognizer.

Next we define the language which a DR ΣX-recognizer $\mathbf{A} = (\mathcal{A}, a_0, \alpha)$ accepts. We need the mapping $\tilde{\alpha} : F_\Sigma(X) \to pA$, which is defined as follows:

(1) If $x \in X$, then $x\tilde{\alpha} = x\alpha$.
(2) If $\sigma \in \Sigma_0$, then $\sigma\tilde{\alpha} = \{\sigma^{\mathcal{A}}\}$.
(3) If $t = \sigma(t_1, \ldots, t_m)$, where $m \geq 1$, then

$$t\tilde{\alpha} = \{a \in A | \sigma^{\mathcal{A}}(a) \in (t_1\tilde{\alpha} \times \ldots \times t_m\tilde{\alpha})\}.$$

Now the forest recognized by $\mathbf{A}$ is the set

$$T(\mathbf{A}) = \{t \in F_\Sigma(X) | a_0 \in t\tilde{\alpha}\}.$$

A forest that can be recognized by a DR ΣX-recognizer is called *DR-recognizable* or simply a *DRec-language*. The set of all DR-recognizable ΣX-tree languages is $DRec(\Sigma, X)$.

Because a deterministic recognizer can always be regarded as nondeterministic, a DR-recognizable language is also recognizable. Thus $DRec(\Sigma, X) \subseteq Rec(\Sigma, X)$.

Lemma 2.1 *If* $\Sigma \neq \Sigma_0 \cup \Sigma_1$, *then* $DRec(\Sigma, X)$ *is properly included in* $Rec(\Sigma, X)$.

Proof. We generalize a tree language originally due to Magidor and Moran [MM69] and simplified by Thatcher [Tha73]. Let $x \in X \cup \Sigma_0$ and $\sigma \in \Sigma_m$ for $m \geq 2$. Then the forest $\{\sigma(\sigma(x, \ldots, x), x, \ldots, x), \sigma(x, \ldots, x, \sigma(x, \ldots, x))\}$ belongs to $Rec(\Sigma, X)$, but it is not DR-recognizable. □

Next we define the paths of a tree and the path closure of a forest. Using the path closure concept we can distinguish $DRec$-languages among Rec-languages. Then we can easily see that any intersection of finitely many $DRec$-languages is also a $DRec$-language, but that the Boolean closure of $DRec$-languages properly contains the $DRec$-languages themselves.

Let Σ be a ranked alphabet. For every operation symbol $\sigma \in \Sigma_m$ $(m > 0)$, we define a set of new unary operations $\Gamma(\sigma) = \{\sigma_1, \ldots, \sigma_m\}$ so that if $\sigma \neq \tau$, then $\Gamma(\sigma) \cap \Gamma(\tau) = \emptyset$. Then we form a new alphabet $\Gamma = \Gamma_0 \cup \Gamma_1$, where

(1) $\Gamma_0 = \Sigma_0$ and
(2) $\Gamma_1 = \bigcup\{\Gamma(\sigma) | \sigma \in \Sigma_m, m \geq 1\}$.

The *paths* of a tree $t \in F_\Sigma(X)$ form the set $\delta(t) \subseteq F_\Gamma(X)$ defined as follows:

(1) For $x \in X$, let $\delta(x) = \{x\}$.
(2) For $\sigma \in \Sigma_0$, let $\delta(\sigma) = \{\sigma\}$.
(3) For $t = \sigma(t_1, \ldots, t_m)$, let

$$\delta(t) = \bigcup_{i=1}^{m} \sigma_i(\delta(t_i)).$$

Now the set of the paths of a forest $T \subseteq F_\Sigma(X)$ is

$$\delta(T) = \bigcup\{\delta(t) | t \in T\}$$

and its *path closure* $\Delta(T)$ is the forest

$$\Delta(T) = \{t \in F_\Sigma(X) | \delta(t) \subseteq \delta(T)\}.$$

For example, the path set of the tree $t = \sigma(z, \sigma(\omega(y), x))$ contains the elements $\sigma_1(z), \sigma_2(\sigma_1(w_1(y)))$ and $\sigma_2(\sigma_2(x))$, and the path closure of the forest $T = \{\sigma(x, y), \sigma(y, x)\}$ is $\Delta(T) = T \cup \{\sigma(x, x), \sigma(y, y)\}$.

Some of the properties of the path closure are noted in the following lemma.

Lemma 2.2 [Vir80]. *If $T, T_1, T_2 \subseteq F_\Sigma(X)$, then*

(1) $T \subseteq \Delta(T)$,
(2) $\Delta(T) = \Delta(\Delta(T))$ *and*
(3) $T_1 \subseteq T_2$ *implies* $\Delta(T_1) \subseteq \Delta(T_2)$. □

Theorem 2.3 [Cou78b, Vir80]. *Let $T \in Rec(\Sigma, X)$ and $\Sigma_0 = \emptyset$. Then*

$$T \in DRec(\Sigma, X) \quad \text{iff} \quad \Delta(T) = T.$$

□

Corollary 2.4 *Let $S, T \subseteq F_\Sigma(X)$. Then*

$$S, T \in DRec(\Sigma, X) \quad \text{implies} \quad S \cap T \in DRec(\Sigma, X).$$

Proof. We present a short proof in the case $\Sigma_0 = \emptyset$. For a general ranked alphabet, a product of two DR-recognizers can be constructed that accepts the intersection. If $S, T \in DRec(\Sigma, X)$, then $S \cap T \in Rec(\Sigma, X)$, because $Rec(\Sigma, X)$ is closed under Boolean operations.

Assume $t \in \Delta(S \cap T)$. Then $\delta(t) \subseteq \delta(S \cap T) \subseteq \delta(S) \cap \delta(T)$. Now $t \in \Delta(S) \cap \Delta(T) = S \cap T$. Thus $S \cap T \in DRec(\Sigma, X)$. □

Let $\mathcal{F}$ be a family of subsets of a set U. The *Boolean closure* $\mathcal{B}(\mathcal{F})$ of $\mathcal{F}$ is the smallest set $\mathcal{G}$ of subsets of U which contains $\mathcal{F}$ such that $X, Y \in \mathcal{G}$ implies $X \cap Y, X \cup Y$ and $U \backslash X \in \mathcal{G}$. We denote the complement $U \backslash X$ of X by X^C. The following theorem can be found, for instance, in [Sik64].

Theorem 2.5 *Let $\mathcal{F} \subseteq pU$ be a family of subsets of a given set U. A subset T of U is in the Boolean closure of $\mathcal{F}$ iff there exist $k \geq 1, n_1, \ldots, n_k \geq 1$ and $m_1, \ldots, m_k \geq 0$ such that T can be expressed in the form*

$$\begin{aligned} T \;=\; & (F_{11} \cap F_{12} \cap \cdots \cap F_{1m_1} \cap F^C_{1,m_1+1} \cap F^C_{1,m_1+2} \cap \cdots \cap F^C_{1n_1}) \cup \\ & (F_{21} \cap F_{22} \cap \cdots \cap F_{2m_2} \cap F^C_{2,m_2+1} \cap F^C_{2,m_2+2} \cap \cdots \cap F^C_{2n_2}) \cup \\ & \vdots \\ & (F_{k1} \cap F_{k2} \cap \cdots \cap F_{km_k} \cap F^C_{k,m_k+1} \cap F^C_{k,m_k+2} \cap \cdots \cap F^C_{kn_k}), \end{aligned}$$

where $F_{ij} \in \mathcal{F}$ for every $1 \leq i \leq k$ and every $1 \leq j \leq n_i$. □

Corollary 2.4 and Theorem 2.5 give the following result.

Corollary 2.6 *A set $T \subseteq Rec(\Sigma, X)$ belongs to $\mathcal{B}(DRec(\Sigma, X))$ iff there exist $k \geq 1$ and $n_1, \ldots, n_k \geq 1$ such that T can be presented in the form*

$$\begin{aligned} T \;=\; & (T_{11} \cap T^C_{12} \cap T^C_{13} \cap \cdots \cap T^C_{1n_1}) \cup \\ & (T_{21} \cap T^C_{22} \cap T^C_{23} \cap \cdots \cap T^C_{2n_2}) \cup \\ & \vdots \\ & (T_{k1} \cap T^C_{k2} \cap T^C_{k3} \cap \cdots \cap T^C_{kn_k}), \end{aligned}$$

where for all $1 \leq i \leq k$, $1 \leq j \leq n_i$, the language $T_{ij} \in DRec(\Sigma, X)$. □

Since one-element tree language $\{t\}$ is always DR-recognizable, every finite language belongs to $\mathcal{B}(DRec(\Sigma, X))$. The language $T = \{\sigma(\sigma(x, \ldots, x), x, \ldots, x), \sigma(x, \ldots, x, \sigma(x, \ldots, x))\}$ does not belong to $DRec(\Sigma, X)$ according to the proof of Lemma 2.1, but as finite it is in $\mathcal{B}(DRec(\Sigma, X))$. This observation gives Theorem 2.7.

Theorem 2.7 *If $\Sigma \neq \Sigma_0 \cup \Sigma_1$, then $DRec(\Sigma, X)$ is properly included in $\mathcal{B}(DRec(\Sigma, X))$.* □

3 $\mathcal{B}(DRec)$ is a Variety

Next we show that the Boolean closure of the DR-recognizable languages is a tree language variety. For a family of tree languages to form a variety it is required to be closed under Boolean operations, inverse translations and inverse homomorphisms [Ste79,Ste92].

Let Σ be fixed. If one has defined for every alphabet X a set $\mathcal{V}(X)$ of recognizable ΣX-tree languages, then the family $\mathcal{V} = \{\mathcal{V}(X)\}$ is called a *family of regular Σ-tree languages*. For instance, $Rec = \{Rec(\Sigma, X)\}$ itself, $Triv = \{\{\emptyset, F_\Sigma(X)\}\}$ and $\mathcal{B}(DRec) = \{\mathcal{B}(DRec(\Sigma, X))\}$ are families of regular Σ-tree languages.

Definition 3.1 *Let Σ be fixed. A* variety of Σ-tree languages *is a family of regular Σ-tree languages $\mathcal{V} = \{\mathcal{V}(X)\}$ such that the conditions*

(1) $\emptyset \neq \mathcal{V}(X) \subseteq Rec(\Sigma, X)$,
(2) $T \in \mathcal{V}(X)$ *implies* $F_\Sigma(X) \backslash T \in \mathcal{V}(X)$,
(3) $T, U \in \mathcal{V}(X)$ *implies* $T \cap U \in \mathcal{V}(X)$,
(4) $T \in \mathcal{V}(X), p \in Sp_\Sigma(X)$ *implies* $p^{-1}(T) \in \mathcal{V}(X)$, *and*
(5) *if* $\phi : \mathcal{F}_\Sigma(X) \to \mathcal{F}_\Sigma(Y)$ *is a morphism and* $T \in \mathcal{V}(Y)$, *then* $T\phi^{-1} \in \mathcal{V}(X)$

are satisfied for all alphabets X and Y.

For example, *Triv* and *Rec* are varieties of Σ-tree languages [Ste92]. The family $\mathcal{B}(DRec)$ is closed under Boolean operations by definition. So we need to study the inverse translations and the inverse homomorphisms.

A translation is based on the notion of a special tree. To show that an inverse image of a DR-recognizable language under a translation is again DR-recognizable we also need the concept of a run tree. The idea of a run tree is to associate with every node of a tree the state in which the recognizer has reached that node. Of course, the states associated depend on the initial state at the root. The run tree is defined using the alphabet $X \cup \{\xi\}$ to facilitate handling of special trees as well.

Let $\mathbf{A} = (A, \Sigma, X, a_0, \alpha)$ be a DR ΣX-recognizer and $\xi \notin X \cup \Sigma$. Then the *run tree* of a tree $t \in F_\Sigma(X \cup \{\xi\})$ in state $a \in A$ is $\mathrm{run}(\mathcal{A}, t, a) \in F_{\Sigma \times A}((X \cup \{\xi\}) \times A)$ defined as follows:

(1) If $y \in X \cup \{\xi\}$, then $\mathrm{run}(\mathcal{A}, y, a) = (y, a)$.
(2) If $\sigma \in \Sigma_0$, then $\mathrm{run}(\mathcal{A}, \sigma, a) = (\sigma, a)$.
(3) If $t = \sigma(t_1, \ldots, t_m)$, where $m \geq 1$ and $\sigma^{\mathcal{A}}(a) = (a_1, \ldots, a_m)$, then

$$\mathrm{run}(\mathcal{A}, t, a) = (\sigma, a)(\mathrm{run}(\mathcal{A}, t_1, a_1), \ldots, \mathrm{run}(\mathcal{A}, t_m, a_m)).$$

If the algebra $\mathcal{A}$ is clear from the context, we denote a run tree also by $\mathrm{run}(t, a)$.

With the help of a run tree we get a new way to find out whether a DR-recognizer accepts a tree.

Lemma 3.2 *Let $\mathbf{A} = (\mathcal{A}, a_0, \alpha)$ be a DR ΣX-recognizer. Then*

$$t \in T(\mathbf{A}) \quad \textit{iff} \quad a \in l\tilde{\alpha} \quad \textit{for all} \quad (l, a) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, t, a_0)).$$

Proof. By tree induction on t one can first prove that $b \in t\tilde{\alpha}$ if and only if $a \in l\tilde{\alpha}$ for all leaves $(l, a) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, t, b))$. Choosing then a_0 for b we get the claim. □

Before the main theorem of this subsection we need to study the product of run trees. This is done using the ξ-depth of a special tree.

The *ξ-depth* $\mathrm{dp}(p)$ of a special tree $p \in Sp_\Sigma(X)$ is the length of the path from the root to the ξ-leaf:

(1) If $p = \xi$, then $\mathrm{dp}(p) = 0$.
(2) If $p = \sigma(p_1, \ldots, p_i, \ldots, p_m)$, where $p_i \in Sp_\Sigma(X)$, then $\mathrm{dp}(p) = \mathrm{dp}(p_i) + 1$.

Lemma 3.3 *Let $\mathbf{A} = (A, \Sigma, X, a_0, \alpha)$ be a DR ΣX-recognizer. Let $t \in F_\Sigma(X), p \in Sp_\Sigma(X)$ and let a and b be states of $\mathbf{A}$. If $(\xi, b) \in \mathrm{leaf}(\mathrm{run}(p, a))$, then*

$$\mathrm{leaf}(\mathrm{run}(t \cdot_\xi p, a)) = \mathrm{leaf}(\mathrm{run}(t, b)) \cup \mathrm{leaf}(\mathrm{run}(p, a)) \backslash \{(\xi, b)\}.$$

Proof. By induction on the ξ-depth of the special tree p one can first verify

$$\mathrm{run}(t \cdot_\xi p, a) = \mathrm{run}(t, b) \cdot_{(\xi, b)} \mathrm{run}(p, a),$$

from which the claim follows. □

Theorem 3.4 *Let $p \in Sp_\Sigma(X)$ and $T \subseteq F_\Sigma(X)$. If T is DR-recognizable, then also $p^{-1}(T)$ is DR-recognizable.*

Proof. If $p^{-1}(T) = \emptyset$, then $p^{-1}(T) \in DRec(\Sigma, X)$. Thus we assume that $p^{-1}(T) \neq \emptyset$.

Let $\mathbf{A} = (A, \Sigma, X, a_0, \alpha)$ be a DR ΣX-recognizer that recognizes the forest T. Because $p^{-1}(T) \neq \emptyset$ there exists $t \in p^{-1}(T)$ which means $t \cdot_\xi p \in T = T(\mathbf{A})$. Then by Lemma 3.2 $a \in l\tilde{\alpha}$ for all $(l, a) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, t \cdot_\xi p, a_0))$. Since p is a special tree there exists exactly one state $b \in A$ such that $(\xi, b) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, p, a_0))$. Now according to Lemma 3.3 we have $a \in l\tilde{\alpha}$ specifically for all $(l, a) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, p, a_0)) \backslash \{(\xi, b)\}$.

Form the new recognizer $\mathbf{B} = (A, \Sigma, X, b, \alpha)$ that differs from $\mathbf{A}$ only by its initial state. Of course, also $\mathbf{B}$ is a DR ΣX-recognizer.

Now we show that $T(\mathbf{B}) = p^{-1}(T)$ and so $p^{-1}(T) \in DRec(\Sigma, X)$: for any ΣX-tree t,

$$\begin{aligned}
& t \in T(\mathbf{B}) \\
\text{iff}\quad & a \in l\tilde{\alpha} \ \text{ for all } (l, a) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, t, b)) \\
\text{iff}\quad & a \in l\tilde{\alpha} \ \text{ for all } (l, a) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, t, b)) \cup \\
& \qquad\qquad \mathrm{leaf}(\mathrm{run}(\mathcal{A}, p, a_0)) \backslash \{(\xi, b)\} \\
\text{iff}\quad & a \in l\tilde{\alpha} \ \text{ for all } (l, a) \in \mathrm{leaf}(\mathrm{run}(\mathcal{A}, t \cdot_\xi p, a_0)) \\
\text{iff}\quad & t \cdot_\xi p \in T(\mathbf{A}) \\
\text{iff}\quad & t \in p^{-1}(T).
\end{aligned}$$

□

The inverse image of a DR-recognizable forest under a homomorphism is studied in Theorem 3.5.

Theorem 3.5 *Let $\phi : \mathcal{F}_\Sigma(X) \to \mathcal{F}_\Sigma(Y)$ be a homomorphism and let $T \in F_\Sigma(Y)$. If T is DR-recognizable, then also $T\phi^{-1} = \{t | t\phi \in T\} (\subseteq F_\Sigma(X))$ is DR-recognizable.*

Proof. Let $\mathbf{A} = (A, \Sigma, Y, a_0, \alpha)$ be a DR ΣY-recognizer that recognizes the forest T. Form a new recognizer $\mathbf{B} = (A, \Sigma, X, a_0, \beta)$ which differs from $\mathbf{A}$ by its alphabet and its final assignment. The mapping $\beta : X \to pA$ is defined by putting $x\beta = x\phi\tilde{\alpha}$ for all $x \in X$. Also B is a DR ΣX-recognizer.

A proof by tree induction shows that $t\tilde{\beta} = t\phi\tilde{\alpha}$ for all $t \in F_\Sigma(X)$. Hence $t \in T(\mathbf{B})$ if and only if $t\phi \in T(\mathbf{A})$. This means that $T\phi^{-1} = T(\mathbf{B})$ is DR-recognizable. □

According to Theorem 3.4 and Theorem 3.5, every $\mathcal{B}(DRec(\Sigma, X))$ satisfies the conditions of Definition 3.1.

Theorem 3.6 *The family $\mathcal{B}(DRec) = \{\mathcal{B}(DRec(\Sigma, X))\}$ is a variety of Σ-tree languages.* □

4 $\mathcal{B}(DRec)$ is Properly Included in *Rec*

Next we show that there is a recognizable tree language that can not be constructed from DR-recognizable languages by finitely many Boolean operations. The proof is based on the pidgeon hole principle and uses Corollary 2.6.

In the beginning of this section we assume that $\Sigma_2 \neq \emptyset$ and that there are at least two variables in X, but later the results are generalized.

A tree $t \in F_\Sigma(X)$ is *balanced,* if all its paths have the same length. Denote the set of all balanced ΣX-trees of height h by $\mathrm{Bal}(h)$.

Let $\sigma \in \Sigma_2$ and $x, y \in X$. A balanced tree $t \in F_{\{\sigma\}}(X)$ is a *left xy-tree,* if $\mathrm{hg}(t) \geq 1$, $\{s \in \mathrm{sub}(t) | \mathrm{hg}(s) \leq 1\} \subseteq \{\sigma(x,y), \sigma(y,x), x, y\}$ and $\sigma(x,y)$ does not appear in t to the right of an occurrence of $\sigma(y,x)$. Thus in a left xy-tree all its subtrees $\sigma(x,y)$ are on the left-hand side and the subtrees $\sigma(y,x)$ are on the right. Denote the set of all left xy-trees of height h by $\mathrm{BLxy}(h)$, where $h \geq 1$. Then $\mathrm{BLxy}(h) \subseteq \mathrm{Bal}(h) \cap F_{\{\sigma\}}(X)$.

The trees in $\mathrm{BLxy}(h)$ differ from each other according to where the leftmost subtree $\sigma(y,x)$ occurs. This also determines how many subtrees $\sigma(x,y)$ it has. We now denote the tree in $\mathrm{BLxy}(h)$ with $n-1$ subtrees $\sigma(x,y)$ by $b(h,n)$, and say that it has the leftmost subtree $\sigma(y,x)$ *at place n.* The tree $b(3,4)$ is displayed in Figure 1 later.

A balanced binary tree of height $h-1$ has 2^{h-1} leaves. When these leaves are then replaced by subtrees $\sigma(x,y)$ and $\sigma(y,x)$, the place for the leftmost subtree $\sigma(y,x)$ can be chosen in $2^{h-1}+1$ ways. So there exist $2^{h-1}+1$ trees in $\mathrm{BLxy}(h)$. Hence

$$\mathrm{BLxy}(h) = \{b(h,n) | n = 1, \ldots, 2^{h-1}+1\}.$$

We also need a mapping $\Omega : \mathrm{BLxy}(h) \rightarrow \mathrm{Bal}(h)$ which replaces the leftmost $\sigma(y,x)$ by $\sigma(x,x)$. If a tree has no $\sigma(y,x)$ at all, then Ω leaves the tree unaltered, i.e. $\Omega(b(h, 2^{h-1}+1)) = b(h, 2^{h-1}+1)$. Note that Ω is an injection.

Lemma 4.1 *Let $TB = \{b(h,n_1), \ldots, b(h,n_p)\} \subseteq \mathrm{BLxy}(h)$, where $n_1 < n_2 < \cdots < n_p$. Then*

$$\Omega(TB \setminus \{b(h,n_p)\}) \subseteq \Delta(TB).$$

Proof. Consider the tree $\Omega(b(h,n_i))$, where $1 \leq i < p$. At place n_i it has a subtree $\sigma(x,x)$, and this is the only place where it differs from the original tree $b(h,n_i)$, which has a subtree $\sigma(y,x)$ at place n_i. The tree $b(h,n_p)$ has a subtree $\sigma(x,y)$ at place n_i. Thus $\Omega(b(h,n_i)) \in \Delta(\{b(h,n_i), b(h,n_p)\}) \subseteq \Delta(TB)$. □

Lemma 4.2 *Let $t_1 = b(h,n_1)$ and $t_2 = b(h,n_2)$, where $n_1 < n_2$. Then*

$$n_1 < n \leq n_2 \quad \text{implies} \quad b(h,n) \in \Delta(\{\Omega(t_1), \Omega(t_2)\}).$$

Proof. Consider a tree $b(h,n)$, where $n_1 < n \leq n_2$. Left to the place n it has only subtrees $\sigma(x,y)$ just like the tree $\Omega(t_2)$. At place n and right to it the tree $b(h,n)$ has only subtrees $\sigma(y,x)$ just like the tree $\Omega(t_1)$. □

Lemma 4.3 *Let $\sigma \in \Sigma_2, x, y \in X$ and $T \subseteq F_\Sigma(X)$. If no tree in T has a subtree $\sigma(x,x)$ and*

$$\mathrm{BLxy}(h)\backslash\{b(h,1), b(h, 2^{h-1}+1)\} \subseteq T \quad \textit{for all} \quad h \geq 2,$$

then T does not belong to $\mathcal{B}(DRec(\Sigma, X))$.

Proof. Suppose that $T \in \mathcal{B}(DRec(\Sigma, X))$. Then there exist $k, n_1, \ldots, n_k \geq 1$ and languages $T_{ij} \in DRec(\Sigma, X)(1 \leq i \leq k$ and $1 \leq j \leq n_i)$, such that

$$\begin{aligned} T \quad = \quad & (T_{11} \cap T_{12}^C \cap \cdots \cap T_{1n_1}^C) \cup \\ & (T_{21} \cap T_{22}^C \cap \cdots \cap T_{2n_2}^C) \cup \\ & \quad \vdots \\ & (T_{k1} \cap T_{k2}^C \cap \cdots \cap T_{kn_k}^C). \end{aligned}$$

Denote $m = \max_{1 \leq i \leq k} n_i$.

For any $h \geq 2$, the forest $\mathrm{BLxy}(h)\backslash\{b(h,1), b(h, 2^{h-1}+1)\}$ is a subset of T and it has $2^{h-1}-1$ elements. Choose then h so big that

$$2^{h-1} - 1 \geq k(m+1).$$

Then there exists an $i \in [1, k]$ such that

$$TB = (\mathrm{BLxy}(h)\backslash\{b(h,1), b(h, 2^{h-1}+1)\}) \cap T_{i1} \cap T_{i2}^C \cap \cdots \cap T_{in_i}^C$$

contains at least $m+1$ trees. This means that $|TB| \geq n_i + 1$. Note also that $TB \cap T_{ij} = \emptyset$, if $2 \leq j \leq n_i$.

Consider the set $\Omega(TB)$. Every tree in it has a subtree $\sigma(x,x)$, so no tree in $\Omega(TB)$ belongs to T. Especially, no tree in $\Omega(TB)$ belongs to the set $T_{i1} \cap T_{i2}^C \cap \cdots \cap T_{in_i}^C$.

Let $s = \max\{s_i | b(h, s_i) \in TB\}$. By Lemma 4.1

$$\Omega(TB\backslash\{b(h,s)\}) \subseteq \Delta(TB) \subseteq \Delta(T_{i1}) = T_{i1}.$$

We can not have $n_i = 1$; otherwise $T_{i1} = T_{i1} \cap T_{i2}^C \cap \cdots \cap T_{in_i}^C$ and the trees in $\Omega(TB\backslash\{b(h,s)\})$ would belong to T. Thus we assume $n_i \geq 2$. Also we can deduce that no tree in $\Omega(TB\backslash\{b(h,s)\})$ belongs to $T_{i2}^C \cap \cdots \cap T_{in_i}^C$.

The injectivity of Ω implies that $|\Omega(TB\backslash\{b(h,s)\})| = |TB\backslash\{b(h,s)\}| = |TB| - 1 \geq n_i$. This means that in $TB\backslash\{b(h,s)\}$ there are two trees $t_1 = b(h, s_1)$ and $t_2 = b(h, s_2)$, where $s_1 < s_2$, and one set T_{ij}^C of the sets $T_{i2}^C, \ldots, T_{in_i}^C$ such that $\Omega(t_1), \Omega(t_2) \notin T_{ij}^C$. In other words, $\Omega(t_1), \Omega(t_2) \in T_{ij}$. By Lemma 4.2

$$t_2 = b(h, s_2) \in \Delta(\{\Omega(t_1), \Omega(t_2)\}) \subseteq \Delta(T_{ij}) = T_{ij}.$$

On the other hand, $t_2 \in TB\backslash\{b(h,s)\}$. Thus $TB \cap T_{ij} \neq \emptyset$, which is a contradiction.

This means that T does not belong to the Boolean closure of DR-recognizable languages. □

Next we study the case where there are no binary operators. Let τ be an m-ary operator for some $m > 2$. First we expand the trees in BLxy(h) by the following mapping $\Phi : F_{\{\sigma\}}(X) \to F_{\{\tau\}}(X)$:

(1) $\Phi(x) = x$ for all $x \in X$ and
(2) $\Phi(\sigma(t_1, t_2)) = \tau(\Phi(t_1), \Phi(t_2), x, \ldots, x)$.

In fact, Φ is a linear tree homomorphism, but more importantly it is an injection. Moreover, it preserves the height of a tree, and the subtrees of $\Phi(\text{BLxy}(h))$ of height 1 are in the set $\{\tau(x, y, x, \ldots, x), \tau(y, x, x, \ldots, x)\}$. The effect of Φ is illustrated by Figure 1.

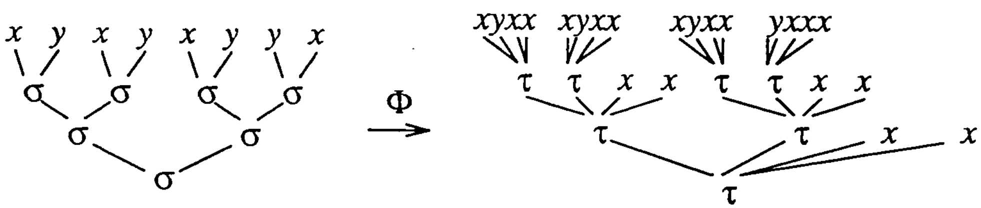

Figure 1. The effect of Φ on the tree $b(3,4)$.

Lemma 4.4 *Let $m \geq 2, \tau \in \Sigma_m, x, y \in X$ and $T \subseteq F_\Sigma(X)$. If no tree in T has a subtree $\tau(x, x, x, \ldots, x)$ and*

$$\Phi(\text{BLxy}(h)) \setminus \{\Phi(b(h,1)), \Phi(b(h, 2^{h-1}+1))\} \subseteq T \quad \textit{for all } h \geq 2,$$

then T does not belong to $\mathcal{B}(DRec(\Sigma, X))$.

Proof. We repeat the proof of Lemma 4.3 using the modified mapping $\Omega : \Phi(\text{BLxy}(h)) \to \Phi(F_\Sigma(X))$, which is defined to replace the leftmost $\tau(y, x, x, \ldots, x)$, by $\tau(x, x, x, \ldots, x)$. If a tree does not have a subtree $\tau(y, x, x, \ldots, x)$, then Ω leaves it unchanged. Also now Ω is an injection in the set $\Phi(\text{BLxy}(h))$.

Throughout Lemma 4.1, Lemma 4.2 and Lemma 4.3 the trees $\Phi(b(h,n))$ are used instead of the trees $b(h,n)$. The proofs of the first two lemmas consider only the ordering of the leaves of subtrees of height 1, and from this point of view the trees $b(h,n)$ and $\Phi(b(h,n))$ are essentially the same.

Lemma 4.3 is based on the fact that BLxy(h) can always be chosen sufficiently large by increasing h. Because Φ is an injection, the number of trees in $\Phi(\text{BLxy}(h))$ have the same property. Otherwise the rest of the proof continues identically to the proof of Lemma 4.3. □

Theorem 4.5 *If $\Sigma \neq \Sigma_0 \cup \Sigma_1$ and $|X| \geq 2$, then $\mathcal{B}(DRec(\Sigma, X))$ is properly contained in $Rec(\Sigma, X)$. Hence, $\mathcal{B}(DRec)$ is a proper subvariety of Rec.*

Proof. If $\tau \in \Sigma_m$, where $m \geq 2$, and $x, y \in X$, then the ΣX-tree language

$$\begin{aligned} T &= \{t | \{s \in \text{sub}(t) | \text{hg}(s) \leq 1\} \\ &= \{\tau(x, y, x, \ldots, x), \tau(y, x, x, \ldots, x), x, y\}\} \end{aligned}$$

is recognizable, and it satisfies the conditions of Lemma 4.4. Thus it distinguishes the families $\mathcal{B}(DRec(\Sigma, X))$ and $Rec(\Sigma, X)$. □

5 $\mathcal{B}(DRec)$ and Other Varieties

In this section we define the tree language varieties *D, RD, GD, Nil* and *Loc* and compare $\mathcal{B}(DRec)$ with them.

The inclusion relation of varieties is defined componentwise: if $\mathcal{V} = \{\mathcal{V}(X)\}$ and $\mathcal{U} = \{\mathcal{U}(X)\}$ are varieties and $\mathcal{V}(X) \subseteq \mathcal{U}(X)$ for every alphabet X, then we write $\mathcal{V} \subseteq \mathcal{U}$. The trivial variety $Triv = \{\{\emptyset, F_\Sigma(X)\}\}$ and the variety $Rec = \{Rec(\Sigma, X)\}$ of all recognizable languages are the smallest and the largest tree language varieties and $Triv \subset \mathcal{B}(DRec) \subset Rec$. The intersection of varieties $\mathcal{U}$ and $\mathcal{V}$ is $\mathcal{U} \cap \mathcal{V} = \{\mathcal{U}(X) \cap \mathcal{V}(X)\}$.

Definite, reverse definite and generalized definite tree languages were defined by Heuter [Heu89b] and shown to form varieties by Steinby [Ste92].

Definite tree languages. In a definite tree language the membership can be tested by looking at the nodes near the root. These nodes form a part of a tree called the k-root.

The k-root $r_k(t) \in F_\Sigma(X \cup \Sigma) \cup \{\varepsilon\}$ of a tree $t \in F_\Sigma(X)$ is defined as follows:

(1) $r_0(t) = \varepsilon$

(2) $r_1(t) = \mathrm{root}(t)$

(3) Let $k \geq 2$.

a) If $\mathrm{hg}(t) < k$, then $r_k(t) = t$.

b) If $\mathrm{hg}(t) \geq k$ and $t = \sigma(t_1, \ldots, t_m)$, then $r_k(t) = \sigma(r_{k-1}(t_1), \ldots, r_{k-1}(t_m))$.

The special symbol $\varepsilon \notin X \cup \Sigma$ means the empty tree.

For example, the k-roots of a tree $t = \sigma(\sigma(x, \gamma), y)$ are $r_0(t) = \varepsilon$, $r_1(t) = \sigma$, $r_2(t) = \sigma(\sigma, y)$ and $r_k(t) = t$ for all $k \geq 3$.

Let $k \geq 0$. A forest $T \subseteq F_\Sigma(X)$ is *k-definite,* if for all trees $s, t \in F_\Sigma(X)$,

$$(t \in T \quad \text{and} \quad r_k(s) = r_k(t)) \quad \text{imply} \quad s \in T.$$

The family of all k-definite ΣX-languages is denoted by $D(k, X)$. We write $D(k) = \{D(k, X)\}$. On the other hand, the family of definite Σ-tree languages is $D = \{D(X)\}$, where $D(X) = \bigcup_{k \geq 0} D(k, X)$.

For example, the language $\{\sigma(x, y), \sigma(y, x)\}$ belongs to $D(2, X)$. Note that according to Lemma 2.1 it is not DR-recognizable.

The definition of $D(k, X)$ can be rephrased by means of a congruence θ_k of the term algebra $\mathcal{F}_\Sigma(X)$ which is defined so that, for any ΣX-trees s and t,

$$s \theta_k t \quad \text{iff} \quad r_k(s) = r_k(t).$$

A ΣX-tree language T is k-definite iff it is saturated by θ_k, i.e. $T = T\theta_k$.

The members of a θ_k-class have all the same k-root, which fully determines the class. For a fixed k, there are only finitely many k-roots, and therefore the congruence θ_k is finite.

Reverse definite tree languages. To see whether a tree belongs to a reverse definite tree language only its subtrees lower than given height need to be known.

Let $h \geq 0$ and $t \in F_\Sigma(X)$. Denote by

$$S_h(t) = \{s \in \mathrm{sub}(t) | \mathrm{hg}(s) < h\}$$

all the subtrees of t of height at most $h-1$. A ΣX-tree language is *reverse h-definite,* if for every $s, t \in F_\Sigma(X)$,

$$(t \in T \quad \text{and} \quad S_h(s) = S_h(t)) \quad \text{imply} \quad s \in T.$$

For example, if $t = \sigma(\omega(x), \sigma(x,y))$, then $S_0(t) = \emptyset, S_1(t) = \{x,y\}, S_2(t) = \{\omega(x), \sigma(x,y), x, y\}$ and $S_3(t) = S_4(t) = \cdots = \text{sub}(t)$.

Let $h \geq 0$. The set of all reverse h-definite ΣX-languages is $RD(h,X)$. Also we denote $RD(h) = \{RD(h,X)\}$. The family of all reverse definite Σ-languages is $RD = \{RD(X)\}$, where $RD(X) = \bigcup_{h\geq 0} RD(h,X)$.

As in the case of definite languages there exists a finite congruence θ^h of $\mathcal{F}_\Sigma(X)$ that characterizes the reverse definite ΣX-tree languages. This relation is defined so that, for any $s, t \in F_\Sigma(X)$,

$$s\theta^h t \quad \text{iff} \quad S_h(s) = S_h(t).$$

Now a tree language is reverse h-definite if and only if it is saturated by θ^h.

Generalized definite tree languages. A tree language is generalized definite, if for some $h, k \geq 0$, the membership of a tree is determined only by the tree's k-root and its subtrees of height less than h.

For $h, k \geq 0$ and ΣX-trees s and t, the relation θ_k^h is defined so that

$$s\theta_k^h t \quad \text{iff} \quad (S_h(s) = S_h(t) \quad \text{and} \quad r_k(s) = r_k(t)).$$

Then θ_k^h is a finite congruence of $\mathcal{F}_\Sigma(X)$.

A forest $T \subseteq F_\Sigma(X)$ is *generalized h,k-definite* if and only if for all $s, t \in F_\Sigma(X)$,

$$(t \in T \quad \text{and} \quad s\theta_k^h t) \quad \text{imply} \quad s \in T.$$

Again, a tree language is generalized h,k-definite if and only if it is saturated by the congruence θ_k^h.

The family of all generalized h,k-definite ΣX-tree languages is $GD(h,k,X)$. Then we write $GD(h,k) = \{GD(h,k,X)\}$. Also

$$GD(X) = \bigcup_{h\geq 0} \bigcup_{k \geq 0} GD(h,k,X).$$

Now $GD = \{GD(X)\}$ is the family of all generalized definite ΣX-tree languages.

Comparison between definite varieties and $\mathcal{B}(DRec)$. It is easy to see that $D(0) = RD(0) = GD(0,0) = Triv$. For the general case, the connections between definite, reverse definite and generalized definite tree language families are established by

Theorem 5.1 [Ste92]. *Let $h, k \geq 0$. Then*

(1) $D(k), RD(h), GD(h,k)$ *and* D, RD, GD *are tree language varieties,*
(2) $D(0) \subseteq D(1) \subseteq \cdots \subseteq D \subseteq Rec$,
(3) $RD(0) \subseteq RD(1) \subseteq \cdots \subseteq RD \subseteq Rec$,
(4) $GD(0,k) = D(k)$,
$GD(h,0) = RD(h)$,
(5) $GD(h,k) \subseteq GD(h+1,k) \cap GD(h,k+1)$ *and*
(6) $GD(h,k) \subseteq GD \subseteq Rec$. □

If $\Sigma_0 = \emptyset$ and the ranked alphabet Σ is unary, then every forest is closed under Δ-operation, and $DRec = Rec$. Thus D, RD and GD are all included in $\mathcal{B}(DRec)$. That the inclusion is proper can be seen by considering the forest $T_1 = \{t_i \mid i \text{ is even}\}$, where

$$\begin{cases} (1) & t_0 = x \text{ and} \\ (2) & t_{n+1} = \sigma(t_n). \end{cases}$$

The language T_1 is DR-recognizable, but it does not belong to any of $D(k, X)$, $RD(h, X)$ or $GD(h, k, X)$ for any $h, k \geq 0$.

If Σ is trivial, then the construction of inclusions of Theorem 5.1 collapses and $Rec = GD(0,1) = GD(1,0) = GD(h,k)$ for all $h, k \geq 1$.

We show now that for any Σ with $\Sigma_0 = \emptyset$ the varieties $GD(1,k)$ for every $k \geq 0$, and hence, also varieties D and $RD(1)$ are contained in $\mathcal{B}(DRec)$.

Theorem 5.2 *Let $\Sigma_0 = \emptyset$. For all $k \geq 0$, the variety $GD(1,k)$ is included in $\mathcal{B}(DRec)$.*

Proof. Let X be an alphabet, $k \geq 0$ and $T \in GD(1, k, X)$. Because T is saturated by θ_k^1, it is the union of some θ_k^1-classes. This union is finite, since θ_k^1 is finite. Therefore it suffices to show that any θ_k^1-class belongs to $\mathcal{B}(DRec(\Sigma, X))$.

For $t \in F_\Sigma(X)$, let $t\theta_k^1$ be the θ_k^1-class of t. Because $t\theta_k^1 = t\theta^1 \cap t\theta_k$, we will prove $t\theta_k^1 \in \mathcal{B}(DRec(\Sigma, X))$ by studying $t\theta^1$ and $t\theta_k$ separately.

Firstly, the class $t\theta_k$ is recognizable. If $s \in \Delta(t\theta_k)$, then $r_k(s) = r_k(t)$. So $s \in t\theta_k$. This means $t\theta_k$ is also DR-recognizable.

Secondly, the trees in a θ^1-class have the same set of leaves, which the Δ-operation can only reduce. Thus $t\theta^1$ can be written in the form

$$t\theta^1 = \Delta(t\theta^1) \setminus \bigcup \{\Delta(s\theta^1) \mid \text{leaf}(s) \subset \text{leaf}(t)\}.$$

The sets $\Delta(t\theta^1)$ are DR-recognizable. Namely, if the leaves of t are all the same, say $\text{leaf}(t) = \{x\}$, then $\Delta(t\theta^1) = \{s \mid \text{leaf}(s) = \{x\}\} = t\theta^1$ is recognizable. But if t has at least two different leaves, then

$$\Delta(t\theta^1) \;=\; \bigcup \{u\theta^1 \mid \text{leaf}(u) \subseteq \text{leaf}(t)\} \setminus \{s \mid |\delta(s)| = 1\},$$

where both the union of θ^1-classes and the set of *chains* $\{s \mid |\delta(s)| = 1\}$ are recognizable. This means that $t\theta^1 \in \mathcal{B}(DRec(\Sigma, X))$. Hence, $T \in \mathcal{B}(DRec(\Sigma, X))$. $\square$

Next we show by generalizing the previously mentioned forest T_1 that the inclusion of Theorem 5.2 is proper, if Σ is not trivial.

Let $x \in X$ and $\sigma \in \Sigma_m$, for $m \geq 1$. For each $i \geq 0$, we define the special trees s^i so that

(1) $s^0 = \xi$,
(2) $s^1 = \sigma(\xi, x, \ldots, x)$ and
(3) $s^{n+1} = s^n \cdot_\xi \sigma(\xi, x, \ldots, x)$.

Note that the superscript i indicates the height and also the number of σ-nodes in s^i. The forest $T = \{x \cdot_\xi s^i \mid i \text{ is even}\}$ is DR-recognizable, since $\Delta(T) = T$. Lemma 5.3 shows that it is not generalized definite and thus neither definite nor reverse definite.

Lemma 5.3 *The forest $T = \{x \cdot_\xi s^i \mid i \text{ is even }\}$ is not generalized h,k-definite for any $h, k \geq 0$.*

Proof. Assume that there exist $h, k \geq 1$ such that $T \in GD(h,k,X)$. Now $t = x \cdot_\xi s^{2h+2k} \in T$. Its k-root is $r_k(t) = \sigma \cdot_\xi s^{k-1}$ and $S_h(t) = \{x \cdot_\xi s^i \mid i = 0, 1, \ldots, h-1\}$. On the other hand, the tree $u = x \cdot_\xi s^{2h+2k+1}$ has the same k-root as t and the same set of subtrees of height at most $h-1$. Therefore u belongs to T, which is contrary to the definition of T.

If $T \in GD(h,k,X)$ for $h < 1$ or $k < 1$, then $T \in GD(h,k,X)$ for $h, k \geq 1$ by Theorem 5.1 (5). Thus the claim holds for all $h, k \geq 0$. □

As a result we get

Theorem 5.4 *If $\Sigma \neq \emptyset, \Sigma_0 = \emptyset$ and $h, k \geq 0$, then*

(1) $D \subset \mathcal{B}(DRec)$,
(2) $RD(1) \subset \mathcal{B}(DRec)$,
(3) $GD(1,k) \subset \mathcal{B}(DRec)$,
(4) $\mathcal{B}(DRec) \not\subseteq RD(h)$,
(5) $\mathcal{B}(DRec) \not\subseteq GD(h,k)$,
(6) $\mathcal{B}(DRec) \not\subseteq RD$ *and*
(7) $\mathcal{B}(DRec) \not\subseteq GD$. □

To see that RD and GD are not included in $\mathcal{B}(DRec)$ we recall the language T of Theorem 4.5:

$$T = \{t \in F_\Sigma(X) \mid \; S_2(t) = \{\tau(x,y,x,\ldots,x), \tau(y,x,x,\ldots,x), x, y\}\}.$$

Now T is a reverse 2-definite tree language, and thus also a generalized definite tree language. Since T does not belong to the Boolean closure of DR-recognizable languages, we have the following

Theorem 5.5 *Let $\Sigma \neq \Sigma_0 \cup \Sigma_1$. For $h \geq 2$ and $k \geq 0$, we have*

(1) $RD(h) \not\subseteq \mathcal{B}(DRec)$,
(2) $GD(h,k) \not\subseteq \mathcal{B}(DRec)$,
(3) $RD \not\subseteq \mathcal{B}(DRec)$ *and*
(4) $GD \not\subseteq \mathcal{B}(DRec)$. □

Finite and cofinite tree languages. The tree language family $Nil = \{Nil(X)\}$ is a variety of Σ-tree languages [Ste92], where the family $Nil(X)$ consists of all finite and cofinite ΣX-tree languages. This variety is contained in both D and RD and it itself contains $Triv$.

Theorem 5.6 *The variety Nil is contained in the variety $\mathcal{B}(DRec)$. The inclusion is proper if and only if $\Sigma \neq \Sigma_0$.* □

Local tree languages. Local tree languages are the languages in the Boolean closure of strictly local tree languages. The membership of a tree in a strictly local language is determined, when the root and the forks of a tree are known.

The *forks* of a tree $t \in F_\Sigma(X)$ form a set fork(t) defined as follows:

(1) If $t \in \Sigma_0 \cup X$, then fork$(t) = \emptyset$.
(2) If $t = \sigma(t_1, \ldots, t_m)$, then

$$\operatorname{fork}(t) = \{\sigma(\operatorname{root}(t_1), \ldots, \operatorname{root}(t_m))\} \cup \bigcup_{i=1}^{m} \operatorname{fork}(t_i).$$

The set of all forks in ΣX-trees $\operatorname{fork}(\Sigma, X)$ is

$$\operatorname{fork}(\Sigma, X) = \bigcup \{\operatorname{fork}(t) \mid t \in F_\Sigma(X)\}.$$

For example, the forks of $t = \sigma(\omega(x), \sigma(x, y))$ are $\sigma(\omega, \sigma), \omega(x)$ and $\sigma(x, y)$.

A forest $T \subseteq F_\Sigma(X)$ is *local in the strict sense* or *strictly local,* if there exist a set of forks $F \subseteq \operatorname{fork}(\Sigma, X)$ and a set of roots $R \subseteq \Sigma \cup X$ such that

$$t \in T \quad \text{iff} \quad (\operatorname{fork}(t) \subseteq F \quad \text{and} \quad \operatorname{root}(t) \in R).$$

Then we write $T = \mathrm{SL}(R, F)$.

For example, the languages $L_1 = \{\sigma^i \mid i \geq 1\}$ and $L_2 = \{\sigma_i \mid i \geq 1\}$, where σ^i and σ_i denote the trees

$$\begin{aligned} \sigma^0 &= x, & \sigma_0 &= x, \\ & & \text{and} & \\ \sigma^{n+1} &= \sigma(\sigma^n, x) & \sigma_{n+1} &= \sigma(x, \sigma_n), \end{aligned}$$

are local in the strict sense; they could be defined as $L_1 = \mathrm{SL}(\{\sigma\}, \{\sigma(\sigma, x), \sigma(x, x)\})$ and $L_2 = \mathrm{SL}(\{\sigma\}, \{\sigma(x, \sigma), \sigma(x, x)\})$ as well. But their union $L = L_1 \cup L_2$ is not strictly local. Namely, though the tree $\sigma(x, \sigma(\sigma(x, x), x))$ has σ as root and the forks of it are all forks of trees in L, it does not belong to L.

On the other hand, the intersection $T_1 \cap T_2 = \mathrm{SL}(R_1 \cap R_2, F_1 \cap F_2)$ of two strictly local tree languages $T_1 = \mathrm{SL}(R_1, F_1)$ and $T_2 = \mathrm{SL}(R_2, F_2)$ is strictly local. Note also that $\emptyset = \mathrm{SL}(\emptyset, \emptyset)$ and $F_\Sigma(X) = \mathrm{SL}(\Sigma \cup X, \operatorname{fork}(\Sigma, X))$ are strictly local. However, the previous remarks imply that the complement of a strictly local tree language is not always strictly local.

A forest $T \subseteq F_\Sigma(X)$ is *local,* if it is built from local forests in the strict sense by using finitely many Boolean operations. The family of local Σ-tree languages is $Loc = \{Loc(X)\}$.

The local forests in the strict sense and thereby the local forests are recognizable [GS84]. Furthermore, Loc is a tree language variety [Ste92].

Next we show that also the local tree languages have a characterizing family of congruences. It is easy to see that the relation θ defined by

$$s \theta t \quad \text{iff} \quad (\operatorname{root}(s) = \operatorname{root}(t) \quad \text{and} \quad \operatorname{fork}(s) = \operatorname{fork}(t))$$

is a finite congruence.

Lemma 5.7 *Let $L \subseteq F_\Sigma(X)$. Then L is local if and only if θ saturates L.*

Proof. Let $L \in Loc(\Sigma, X)$. Then there exist $k \geq 1$ and $n \geq 2$ such that L can be written in the form

$$\begin{aligned} L \quad = \quad & (L_{11} \cap L_{12}^{\mathcal{C}} \cap \cdots \cap L_{1n}^{\mathcal{C}}) \cup \\ & (L_{21} \cap L_{22}^{\mathcal{C}} \cap \cdots \cap L_{2n}^{\mathcal{C}}) \cup \\ & \qquad \vdots \\ & (L_{k1} \cap L_{k2}^{\mathcal{C}} \cap \cdots \cap L_{kn}^{\mathcal{C}}), \end{aligned}$$

where for every $1 \le i \le k$ and $1 \le j \le n$, $L_{ij} \in \mathrm{SL}(R_{ij}, F_{ij})$.

Let $t \in L\theta$. Then there exists an $l \in L$ such that $\mathrm{root}(t) = \mathrm{root}(l)$ and $\mathrm{fork}(t) = \mathrm{fork}(l)$. Now

$$l \in L_{i1} \cap L_{i2}^C \cap \cdots \cap L_{in}^C$$

for at least one $i \in [1, k]$. Because $l \in L_{i1}$, then also $t \in L_{i1}$. The reason why $l \notin L_{ij}$ for a $2 \le j \le n$ must be either $\mathrm{root}(l) \notin R_{ij}$ or $\mathrm{fork}(l) \not\subseteq F_{ij}$. In both cases also $t \notin L_{ij}$. Together this means $t \in L$. So $L = L\theta$.

Conversely assume $L = L\theta$. This means that L is the union of some θ-classes. To show that L is local one only needs to verify that any θ-class is local. It is, because if $l \in L$, then

$$l\theta = \mathrm{SL}(\mathrm{root}(l), \mathrm{fork}(l)) \setminus \bigcup \{\mathrm{SL}(\mathrm{root}(l), F) | F \subset \mathrm{fork}(l)\}.$$

□

Now we are ready to compare *Loc* with *DRec* and $\mathcal{B}(DRec)$.

Theorem 5.8 *If $\Sigma \neq \Sigma_0$, we have*

(1) $DRec \not\subseteq Loc$,
(2) $\mathcal{B}(DRec) \not\subseteq Loc$, *and*
(3) $Loc \subset Rec$.

Proof. Let $x \in X \cup \Sigma_0$ and $\sigma \in \Sigma_m$ for $m \ge 1$. The DR-recognizable forest $T = \{\sigma(\sigma(x, \ldots, x), x, \ldots, x)\}$ is not local, since θ does not saturate it. □

The tree language $T_1 = \{\sigma(x, y), \sigma(y, x)\}$ is not DR-recognizable by Lemma 2.1, but clearly it is local. Hence, $Loc \not\subseteq DRec$. However, T_1 does belong to the Boolean closure of DR-recognizable languages. So the question now is, whether this holds for all local languages. For this purpose we consider the following language.

Let $\sigma \in \Sigma_m$, where $m \ge 2$, and $x, y \in X$. Define F as the set of forks $F = \{\sigma(\sigma, \sigma, x, \ldots, x), \sigma(x, y, x, \ldots, x), \sigma(y, x, x, \ldots, x)\}$. Then the forest $\mathrm{SL}(\{\sigma\}, F)$ satisfies the conditions of Lemma 4.4 and thus does not belong to the Boolean closure of DR-recognizable languages. This leads us to

Theorem 5.9 *Let $\Sigma \neq \Sigma_0 \cup \Sigma_1$. Then*

(1) $Loc \not\subseteq DRec$ *and*
(2) $Loc \not\subseteq \mathcal{B}(DRec)$. □

If Σ is unary and $\Sigma_0 = \emptyset$, then *Loc* is contained in $DRec = Rec$. Theorem 5.8 shows that this inclusion is proper. But if Σ is trivial, then every language is local and $Loc = DRec = Rec$.

Figure 2 shows the inclusion relations of varieties for $\Sigma_0 = \emptyset$. If also $\Sigma \neq \Sigma_0 \cup \Sigma_1$, the inclusions are proper and those varieties not connected are incomparable.

Acknowledgment I am grateful to Professor Magnus Steinby for his expert guidance and valuable suggestions during the course of this work.

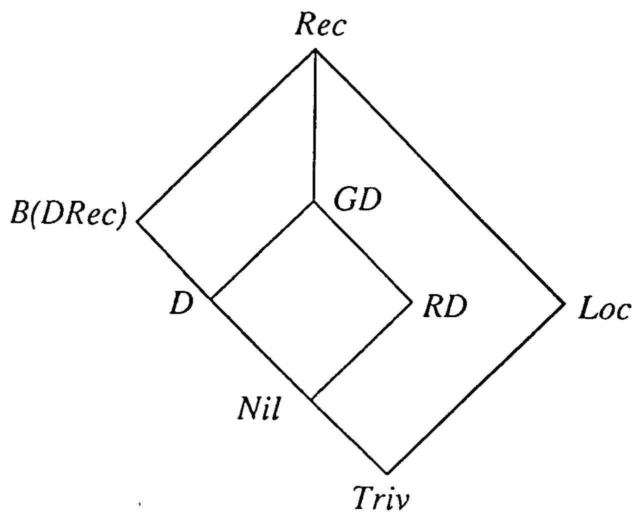

Figure 2. Comparation of studied varieties.

References

[Cou78a] Courcelle, B., A representation of trees by languages I, Theoret. Comput. Sci. 6 (1978), 255-279.

[Cou78b] Courcelle, B., A representation of trees by languages II, Theoret. Comput. Sci. 7 (1978), 25-55.

[Don70] Doner, J.E., Tree acceptors and some of their applications, J. Comput. Syst. Sci. 4 (1970), 406-451.

[GS78] Gécseg, F. and Steinby M., Minimal ascending tree automata, Acta Cybernet. 4 (1978), 37-44.

[GS84] Gécseg, F. and Steinby M., Tree automata, Akadémiai Kiadó, Budapest, 1984.

[Heu88] Heuter, U., Definite tree languages, Bull. EATCS 35 (1988), 137-144.

[Heu89a] Heuter, U., Generalized definite tree languages, Mathem. Found. Comput. Sci. (Proc. Symp., Porabka-Kozubnik, Poland 1989) Lect. Notes in Comput. Sci. 379, Springer-Verlag, Berlin, 1989, pp. 270-280.

[Heu89b] Heuter, U., Zur Klassifizierung regulärer Baumsprachen, Dissertation, Technical University of Aachen, Aachen, 1989.

[MM69] Magidor, M. and Moran, G., Finite automata over finite trees, Technical Report 30, Hebrew University, Jerusalem, 1969.

[Rab69] Rabin, M.O., Decidability of second-order theories and automata on infinite trees, Trans. Am. Math. Soc. 141 (1969), 1-35.

[Sik64] Sikorski, R., Boolean algebras, Springer-Verlag, Berlin, 1964.

[Ste79] Steinby, M., Syntactic algebras and varieties of recognizable sets, Les arbres en algèbre et en programmation, 4ème Coll. Lille (Proc. Coll., Lille 1979), Université de Lille, 1979, pp. 226-240.

[Ste92] Steinby, M., A theory of tree language varieties, Tree Automata and Languages (Nivat, M. and Podelski, A., eds.), Elsevier Publishers, Amsterdam, 1992, pp. 57-81.

[Tha73] Thatcher, J. W., Tree automata: an informal survey, Currents in the Theory of Computing (Aho, A.V., ed.), Prentice-Hall, Englewood Cliffs, N.J., 1973, pp. 143-172.

[TW68] Thatcher, J. W. and Wright, J. B., Generalized finite automata theory with an application to a decision problem of second order logic, Math. Syst. Theory 2 (1968), 57-81.

[Tho84] Thomas, W., Logical aspects in the study of tree languages, Ninth colloquium on trees in algebra and programming (Courcelle, B., ed.), Cambridge Univ. Press, Cambridge, 1984, pp. 31-51.

[Vir80] Virágh, J., Deterministic ascending tree automata I, Acta Cybernet. 5 (1980), 33-42.

Received August 4, 1992.