Leiden Institute of Advanced Computer Science
Leiden University
The Netherlands

Technical Report 99-8

Domains of Partial Attributed Tree Transducers

Zoltán Fülöp, Sebastian Maneth

Domains of Partial Attributed Tree Transducers

Zoltán Fülöp *
Department of Computer Science
József Attila University
Árpád tér 2., H-6720 Szeged, Hungary
E-mail: fulop@inf.u-szeged.hu

Sebastian Maneth
LIACS
Leiden University, PO Box 9512
2300 RA Leiden, The Netherlands
E-mail: maneth@liacs.nl

Abstract

The domains of partial attributed tree transducers (patt's) are the tree languages recognized by tree walking automata in universal acceptance mode. Moreover, the domains of patt's having monadic output are the tree languages recognized by deterministic tree walking automata.

Key words: Formal Languages; Attribute Grammars; Tree Automata

1 Introduction

An attribute grammar (ag) [Knu68] can be seen as a device that translates derivation trees of a context-free grammar G_0 into expressions over some signature. Abstracting from G_0 we obtain an attributed tree transducer (att) [Fül81]. An att A is a total deterministic device that translates all trees in T_Σ (the set of trees over a ranked alphabet Σ) into output trees in T_Δ.

If we allow A to be a *partial* att (patt), then the domain of A is a tree language L over Σ, i.e., a subset of T_Σ. Now the question arises, which tree languages L can be the domain of patt's (equivalently: partial ag's)? From [Bar81] it is known that L is inside $REGT$, the class of regular tree languages. We show that L is the domain of a patt if and only if it can be recognized by a tree walking automaton (twa) in universal acceptance mode. A twa M is a sequential finite-state device that 'walks' on the nodes of a tree s (possibly changing its state). The transition of M is defined nondeterministically, however, the tree s is recognized by M in universal acceptance mode if and only if *every* possible walk of M on s leads to a final state (cf., e.g., [Kam83]). We also show that restricting the output of a patt to monadic trees corresponds to restricting the twa to determinism, which is simply a deterministic tree walking automaton (dtwa) [AU71].

The class of deterministic top-down recognizable tree languages (cf. [GS84]) is properly included in the class of domains of patt's, because (i) a deterministic top-down tree automaton (dta) is a patt with synthesized attributes only and (ii) the domain $\{\sigma(a,b), \sigma(b,a)\}$ of A_1 from Example 1 cannot be recognized by a dta. It is open if every recognizable

*Research of this author was supported by grant No. T 030084 of the Hungarian Scientific Foundation

(equivalently: regular) tree language L can be the domain of a patt. In fact, this is even open for the case of monadic output, i.e., it is not known whether there is an $L \in REGT$ which cannot be recognized by a dtwa (cf. [EHvB99]). This seems to be a deep problem in the distinction between sequential and parallel tree recognizing devices; maybe the characterization by patt's can help. Other possibilities might be the recent characterizations of dtwa's in terms of logic [EHvB99, NS].

Acknowledgement We are grateful to Joost Engelfriet for helpful discussions.

2 Preliminaries

For two nonnegative integers m, n, we denote by $[m, n]$ the set $\{i \mid m \leq i \leq n\}$. Moreover, $[1, n]$ is abbreviated by $[n]$. Let $\mathbb{N} = \{1, 2, \dots\}$ and denote by $\mathbb{N}^*$ the monoid of words (i.e. sequences of positive integers) generated by $\mathbb{N}$. The empty word is denoted by 0, hence for every $u \in \mathbb{N}^*$, $u0 = 0u = u$.

Let Σ be a ranked alphabet. For $k \in \mathbb{N}$ we denote by $\Sigma^{(k)}$ the set of symbols in Σ having rank k. If $\sigma \in \Sigma^{(k)}$, then we denote this fact by $\sigma^{(k)}$. We define $maxrank(\Sigma)$ to be the integer $max\{k \mid \Sigma^{(k)} \neq \varnothing\}$. Σ is called monadic, if $maxrank(\Sigma) \leq 1$

Let A be a set. Then $|A|$ denotes the cardinality of A. As usual, we denote by $T_\Sigma(A)$ the set of trees over Σ indexed by A; for $A = \varnothing$, we write T_Σ for $T_\Sigma(A)$. Let $s \in T_\Sigma(A)$. The set of occurrences (or, nodes) of s is the subset $O(s)$ of $\mathbb{N}^*$ defined inductively as follows. For $s \in A$, $O(s) = \{0\}$, and for $s = \sigma(s_1, \dots, s_k)$, $\sigma \in \Sigma^{(k)}$, $k \geq 0$, and $s_1, \dots, s_k \in T_\Sigma(A)$, $O(s) = \{0\} \cup \{iu \mid i \in [k], u \in O(t_i)\}$. The label of s at $u \in O(s)$ and the subtree of s (rooted) at u are denoted by $lab(s, u)$ and s/u, respectively.

We fix the set $X = \{x_1, x_2, \dots\}$ of variable symbols. For $m \geq 0$, let $X_m = \{x_1, \dots, x_m\}$ and abbreviate $T_{\Sigma,m}$ by $T_\Sigma(X_m)$. Moreover, $\tilde{T}_{\Sigma,m}$ is the subset of $T_{\Sigma,m}$ consisting of trees $s \in T_{\Sigma,m}$ in which every variable of X_m occurs exactly once. For $s \in T_{\Sigma,m}$ and trees $s_1, \dots, s_m$, we denote by $s[s_1, \dots, s_m]$ the tree obtained from s by replacing, for every $i \in [m]$, every occurrence of x_i in s by s_i.

A tree language is a subset of T_Σ. A tree transformation is a partial mapping $\tau : T_\Sigma \to T_\Delta$, for some ranked alphabet Δ. The domain of τ is $dom(\tau) = \{s \in T_\Sigma \mid \exists t \in T_\Delta : \tau(s) = t\}$. For a class C of tree transformations, $dom(C) = \{dom(\tau) \mid \tau \in C\}$.

3 Partial Attributed Tree Transducers

A *partial attributed tree transducer* (for short, patt) is a tuple $A = (S, I, \Sigma, \Delta, a_0, R)$, where S and I are disjoint, unary ranked alphabets of *synthesized* and *inherited attributes*, respectively, Σ and Δ are ranked alphabets (disjoint with $S \cup I$) of *input* and *output symbols*, respectively, and $a_0 \in S$ is the *initial attribute*. The set $R = \bigcup\{R_\sigma \mid \sigma \in \Sigma\} \cup R_{root}$ is a finite set of *rules*, such that for every $\sigma \in \Sigma^{(k)}$ with $k \geq 0$,

(1) for every $a \in S$, the set R_σ contains at most one rule of the form $a(0) \to t[c_1(i_1), \dots, c_m(i_m)]$, where $m \geq 0, t \in \tilde{T}_{\Sigma,m}, c_1, \dots, c_m \in (S \cup I), i_1, \dots, i_m \in [0, k]$,

(2) for every $b \in I$ and $i \in [k]$, the set R_σ contains at most one rule of the form $b(i) \to t[c_1(i_1), \ldots, c_m(i_m)]$, where $t[c_1(i_1), \ldots, c_m(i_m)]$ is as in (1).

Finally,

(3) for every $b \in I$, the set R_{root} contains at most one rule of the form $b(0) \to t[c_1(0), \ldots, c_m(0)]$, where $m \geq 0, t \in \tilde{T}_{\Sigma,m}$, and $c_1, \ldots, c_m \in (S \cup I)$.

The patt A is *in Bochmann normal form* (Bnf), if in (1) and (2), $\forall j \in [m]$: if $c_j \in S$ then $i_j \in [k]$ and if $c_j \in I$ then $i_j = 0$, and in (3), $c_1, \ldots, c_m \in S$. If Δ is monadic, then A is called *monadic*.

Denote $S \cup I$ by Att. Let $s \in T_\Sigma$. We denote by $Att(O(s))$ the set $\{a(u) \mid a \in Att \text{ and } u \in O(s)\}$. Notice that especially $a_0(0) \in O(s)$. The *derivation relation induced by A on s* is the binary relation $\Rightarrow_{A,s}$ over $T_\Delta(Att(O(s)))$ such that, for $\xi_1, \xi_2 \in T_\Delta(Att(O(s)))$, $\xi_1 \Rightarrow_{A,s} \xi_2$ iff there is an occurrence v of ξ_1 with $\xi_1/v = c(u) \in Att(O(s))$, and one of the following three conditions hold.

(1) $c \in S$, $lab(s,u) = \sigma \in \Sigma^{(k)}$, $k \geq 0$, and there is a rule $c(0) \to t[c_1(i_1), \ldots, c_m(i_m)]$ in R_σ such that ξ_2 is obtained from ξ_1 by substituting the tree $t[c_1(ui_1), \ldots, c_m(ui_m)]$ for ξ_1/v.

(2) $c \in I$, $u = u'i$, $u' \in \mathbb{N}^*$, $lab(s,u') = \sigma \in \Sigma^{(k)}$, $k \geq 1$, $i \in [k]$, and there is a rule $c(i) \to t[c_1(i_1), \ldots, c_m(i_m)]$ in R_σ such that ξ_2 is obtained from ξ_1 by substituting the tree $t[c_1(u'i_1), \ldots, c_m(u'i_m)]$ for ξ_1/v.

(3) $c \in I$, $u = 0$, and there is a rule $c(0) \to t[c_1(0), \ldots, c_m(0)]$ in R_{root} such that ξ_2 is obtained from ξ_1 by substituting the tree $t[c_1(0), \ldots, c_m(0)]$ for ξ_1/v.

The normal form $nf(\Rightarrow_{A,s}, \xi)$ of a $\xi \in T_\Delta(Att(O(s)))$ with respect to $\Rightarrow_{A,s}$ is unique, if it exists (see, e.g., [FHVV93]). The *tree transformation τ_A realized by A* is defined as $\tau_A(s) = nf(\Rightarrow_{A,s}, a_0(0))$ if $nf(\Rightarrow_{A,s}, a_0(0)) \in T_\Delta$ (and undefined otherwise). The classes of tree transformations induced by patt's and monadic patt's are denoted by $PATT$ and $PATT_{\text{mon}}$, respectively.

Example 1 Let $A_1 = (S, I, \Sigma, \Delta, a_0, R)$ be the patt with $S = \{start, what, syn_a, syn_b\}$, $I = \{ok, inh_a, inh_b\}$, $\Sigma = \{\sigma^{(2)}, a^{(0)}, b^{(0)}\}$, $\Delta = \{a^{(1)}, b^{(1)}, \#^{(0)}\}$, and R as follows.

$$\begin{array}{lcl} R_\sigma & = & \{start(0) \to what(1), inh_a(1) \to syn_b(2), inh_b(1) \to syn_a(2), ok(2) \to ok(0)\} \\ R_a & = & \{what(0) \to a(inh_a(0)), syn_a(0) \to a(ok(0))\} \\ R_b & = & \{what(0) \to b(inh_b(0)), syn_b(0) \to b(ok(0))\} \\ R_{root} & = & \{ok(0) \to \#\}. \end{array}$$

Note that A_1 is monadic and in Bnf. Consider $s = \sigma(a,b)$. Then $start(0) \Rightarrow_{A_1,s} what(1) \Rightarrow_{A_1,s} a(inh_a(1)) \Rightarrow_{A_1,s} a(syn_b(2)) \Rightarrow_{A_1,s} a(b(ok(2))) \Rightarrow_{A_1,s} a(b(ok(0))) \Rightarrow_{A_1,s} a(b(\#))$.

Hence, $nf(\Rightarrow_{A_1,s}, start(0)) = a(b(\#)) \in T_\Delta$ and thus $\tau_{A_1}(\sigma(a,b)) = a(b(\#))$. For $s = \sigma(b,a)$ we obtain $\tau_{A_1}(s) = b(a(\#))$. Moreover, for every $s' \in T_\Sigma$ with $s' \notin \{\sigma(a,b), \sigma(b,a)\}$, $\tau_{A_1}(s')$ is undefined because $nf(\Rightarrow_{A_1,s'}, start(0)) \notin T_\Delta$. Thus, the domain $dom(\tau_{A_1})$ of A_1 is the tree language $\{\sigma(a,b), \sigma(b,a)\}$. $\square$

4 Tree-Walking Automata

The following definition of tree-walking automaton is the one of [EHvB99]. Note that we allow "stay" moves only at the root of a tree; this is not a restriction because any stay move at a non-root node can by simulated by an up move followed by a down move.

A *tree-walking automaton* (for short, twa) is a tuple $M = (Q, \Sigma, \delta, q_0, F)$, where Q is a finite set of *states*, Σ is a ranked alphabet, $q_0 \in Q$ is the *initial state*, $F \subseteq Q$ is the set of *final states*, and $\delta : (Q \times \Sigma \times [0, m]) \to \mathcal{P}(Q \times \{\uparrow, \downarrow_1, \dots, \downarrow_m\})$ is a function, where $m = \mathit{maxrank}(\Sigma)$. The twa M is *deterministic* (for short, M is a dtwa), if $|\delta(q, \sigma, i)| \leq 1$ for all $q \in Q$, $\sigma \in \Sigma$, and $i \in [0, m]$.

Let $s \in T_\Sigma$ be a tree. The *set of configurations* (*of* M *on* s) is $Q \times O(s)$. A configuration (q, u) is *accepting*, if $q \in F$. The *walk-relation of* M *on* s is the binary relation $\vdash_{M,s}$ over $Q \times O(s)$ such that, for $(q, u), (q', u') \in Q \times O(s)$, $(q, u) \vdash_{M,s} (q', u')$ iff $\mathit{lab}(s, u) = \sigma$, $u = vi$, $v \in \mathbb{N}^*$, $i \in [0, m]$ (with $i \neq 0$ if $u \neq 0$), and either (i) $(q', \downarrow_j) \in \delta(q, \sigma, i)$ and $u' = uj$, or (ii) $(q', \uparrow) \in \delta(q, \sigma, i)$ and $u' = v$.

A *walk of* (*of* M *on* s) is a (possibly infinite) sequence $C_0, C_1, \dots$ of configurations such that $C_0 = (q_0, 0)$ and, for every $i \geq 0$, $C_i \vdash_{M,s} C_{i+1}$. A walk is *accepting* if it contains an accepting configuration, and *maximal* if it is either infinite or it is finite and for its last configuration C there is no configuration C' with $C \vdash_{M,s} C'$. The tree s is recognized by M (in universal mode) if every maximal walk on s is accepting. The tree language recognized by M (in universal mode) is denoted by $L_U(M)$. The classes of tree languages recognized (in universal mode) by twa's and deterministic twa's are denoted $UTWA$ and $DTWA$, respectively. Note that if M is deterministic, then there is exactly one maximal walk on s.

Example 2 Let $M_1 = (Q, \Sigma, \delta, q_0, F)$ be the twa with $Q = \{q_0, q, q_a, q_b, q_f\}$, Σ as in the previous example, $F = \{q_f\}$, and the transition function δ is defined as follows.

- $\delta(q_0, \sigma, 0) = \{(q, \downarrow_1)\}$,
- $\delta(q, a, 1) = \{(q_a, \uparrow)\}$, $\delta(q, b, 1) = \{(q_b, \uparrow)\}$,
- $\delta(q_a, \sigma, 0) = \{(q_a, \downarrow_2)\}$, $\delta(q_b, \sigma, 0) = \{(q_b, \downarrow_2)\}$,
- $\delta(q_a, b, 2) = \{(q_f, \uparrow)\}$, $\delta(q_b, a, 2) = \{(q_f, \uparrow)\}$, and
- $\delta(q, \gamma, i) = \varnothing$ for all q, γ, i not defined above.

Note that M_1 is a dtwa. Consider $s = \sigma(a, b)$. Then $(q_0, 0) \vdash_{M_1,s} (q, 1) \vdash_{M_1,s} (q_a, 0) \vdash_{M_1,s} (q_a, 2) \vdash_{M_1,s} (q_f, 0)$. Hence $\sigma(a, b) \in L_U(M_1)$. It is easy to see that the tree language $L_U(M_1)$ recognized by M_1 is $\{\sigma(a, b), \sigma(b, a)\}$. □

5 Main Result

Let $M = (Q, \Sigma, \delta, q_0, F)$ be a twa. Since we consider universal acceptance, we may assume w.l.o.g. that M stops walking once it is in a final state (the 'stopping assumption'), i.e., for every $q \in F, \sigma \in \Sigma$, and $i \in [0,m]$: $\delta(q,\sigma,i) = \varnothing$. This is because all walks that contain a final state are accepting. Then, $L_U(M)$ equals $\{s \in T_\Delta \mid \{(q_0,0)\} \models^*_{M,s} \varnothing\}$, where $\models_{M,s}$ is the following extension of $\vdash_{M,s}$ to sets. For $K, K' \subseteq Q \times O(s)$, $K \models_{M,s} K'$, iff $\exists(q,u) \in K$ with $q \in F$ or $N \neq \varnothing$ and $K' = N \cup (K - \{(q,u)\})$, where $N = \{(q',u') \mid (q,u) \vdash_{M,s} (q',u')\}$. Note that in case $q \in F$, by the stopping assumption $N = \varnothing$.

Theorem (i) $dom(PATT) = UTWA$ and (ii) $dom(PATT_{\text{mon}}) = DTWA$.

The proof is split up into the following two lemmas. We assume the stopping assumption for the twa in the proof Lemma 3.

Lemma 3 (i) $UTWA \subseteq dom(PATT)$ and (ii) $DTWA \subseteq dom(PATT_{\text{mon}})$.

Proof. First we prove (i). Let $M = (Q, \Sigma, \delta, q_0, F)$ be a twa and let $m = maxrank(\Sigma)$. We now construct the patt $A = (S, I, \Sigma, \Delta, a_0, R)$ with $dom(\tau_A) = L_U(M)$. Let $S = Q \times [0,m] \cup \bar{Q} \times [0,m]$, $I = Q \times \Sigma \cup \bar{Q} \times \Sigma$, $a_0 = \langle q_0, 0\rangle$ and $\Delta = \{\#_0^{(0)}\} \cup \{\#_\nu^{(\nu)} \mid \nu = |\delta(q,\sigma,i)|$ for some $q \in Q, \sigma \in \Sigma, i \in [m]\}$. For every $\sigma \in \Sigma^{(k)}$ with $k \geq 0$, R_σ is the smallest set satisfying conditions (1) – (4).

(1) For all $q \in Q$ and $i \in [0,m]$, if $\delta(q,\sigma,i) = \{(q_1, move_1), \ldots, (q_\nu, move_\nu)\}$ with $\nu > 0$, then let $\langle q,i\rangle(0) \to \#_\nu(c_1(i_1), \ldots, c_\nu(i_\nu))$ be in R_σ, where, for every $j \in [\nu]$, $c_j(i_j) = \langle q_j, l\rangle(l)$, if $move_j = \downarrow_l$ and $c_j(i_j) = \langle \bar{q_j}, \sigma\rangle(0)$, if $move_j = \uparrow$.

(2) For every $\gamma \in \Sigma, q \in Q$, and $i \in [k]$, if $\delta(q,\gamma,i) = \{(q_1, move_1), \ldots, (q_\nu, move_\nu)\}$ with $\nu > 0$, then let $\langle q,\gamma\rangle(i) \to \#_\nu(c_1(i_1), \ldots, c_\nu(i_\nu))$ be in R_σ, where, for every $j \in [\nu]$, $c_j(i_j) = \langle \bar{q_j}, l\rangle(i)$, if $move_j = \downarrow_l$ and $c_j(i_j) = \langle q_j, \gamma\rangle(0)$, if $move_j = \uparrow$.

(3) For all $\bar{q} \in \bar{Q}, i \in [k]$, and $\gamma \in \Sigma$, let $\langle \bar{q}, i\rangle(0) \to \langle q, i\rangle(i)$ and $\langle \bar{q}, \gamma\rangle(i) \to \langle q, \sigma\rangle(0)$ be in R_σ. (Note that these rules do not depend on the transition function of M, hence we call them "built in" rules.)

(4) For every $q \in F$ and $i \in [0,m]$, let $\langle q,i\rangle(0) \to \#_0$ and for every $q \in F, \gamma \in \Sigma$ and $i \in [k]$, let $\langle q,\gamma\rangle(i) \to \#_0$ be in R_σ.

The set R_{root} is the smallest set satisfying the following conditions. For all $\sigma \in \Sigma$ with $k \geq 0$ and $q \in Q$, if $\delta(q,\sigma,0) = \{(q_1, move_1), \ldots, (q_\nu, move_\nu)\}$ with $\nu > 0$, then let $\langle q,\sigma\rangle(0) \to \#_\nu(c_1(i_1), \ldots, c_\nu(i_\nu))$ be in R_{root}, where, for every $j \in [\nu]$, $c_j(i_j) = \langle \bar{q_j}, l\rangle(0)$, if $move_j = \downarrow_l$ and $c_j(i_j) = \langle q_j, \sigma\rangle(0)$, if $move_j = \uparrow$.

Moreover, for every $q \in Q$ and $\sigma \in \Sigma$, let $\langle \bar{q}, \sigma\rangle \to \langle q, \sigma\rangle$ be in R_{root} (built in rules) and, for every $q \in F$ and $\sigma \in \Sigma$, let $\langle q,\sigma\rangle(0) \to \#_0$ be in R_{root}.

Note that A is not in Bnf because there may be rules in R_{root} which contain inherited attributes in their right-hand side.

Let $Att' = Q \times [0,m] \cup Q \times \Sigma$. A tree $t \in T_\Delta(Att'(O(s))$ is *well-formed* if, for every synthesized attribute occurrence $\langle q,i\rangle(u)$ in t, i is the last letter of u (in case $u = 0$ also $i = 0$) and, for every inherited attribute occurrence $\langle q,\sigma\rangle(u)$ in t, $lab(s,u) = \sigma$ holds. Moreover, for a well-formed tree t in $T_\Delta(Att'(O(s))$, let $Conf(t) = \{(q,u) \mid \langle q,d\rangle(u)$ occurs in $t\}$.

<u>*Claim:*</u> Let $s \in T_\Sigma$.

(i) For all $t, t' \in T_\Delta(Att'(O(s)))$, if t is well-formed and $t \Rightarrow_{A,s} t_1 \Rightarrow^*_{A,s} t'$ such that the rule applied in the first step is not a built in one and all rules applied in the further steps are built in ones, then t' is also well-formed and $Conf(t) \models_{M,s} Conf(t')$.

(ii) For all $K, K' \subseteq Q \times O(s)$, if $K \models_{M,s} K'$, then, for every well-formed $t \in T_\Delta(Att'(O(s)))$ with $Conf(t) = K$, there is a well-formed $t' \in T_\Delta(Att'(O(s)))$ with $Conf(t') = K'$ and $t \Rightarrow^*_{A,s} t'$.

It is straightforward to prove this claim by going through the different cases of $c \in Att'$ for $c(u)$ in t; we omit this. Then, the equality $dom(\tau_A) = L_U(M)$ can be proved as follows.

$dom(\tau_A) \subseteq L_U(M)$: Let $s \in dom(\tau_A)$, then there is a $t \in T_\Delta$ with $(\langle q_0, 0\rangle, 0) \Rightarrow^*_{A,s} t$. Since $\Rightarrow_{A,s}$ is confluent, there are $t_1, \dots, t_n \in T_\Delta(Att(O(s)))$ and $t'_1, \dots, t'_n \in T_\Delta(Att'(O(s)))$ such that $(\langle q_0, 0\rangle, 0) \Rightarrow_{A,s} t_1 \Rightarrow^*_{A,s} t'_1 \Rightarrow_{A,s} \dots \Rightarrow_{A,s} t_n \Rightarrow^*_{A,s} t'_n = t$, the rules applied in the single steps $(\langle q_0, 0\rangle, 0) \Rightarrow_{A,s} t_1, \dots, t'_{n-1} \Rightarrow_{A,s} t_n$ are not built in ones, and all rules applied in the other steps are built in ones. Hence, by applying (i) successively we obtain $\{(q_0, 0)\} = Conf((\langle q_0, 0\rangle, 0)) \models^*_{M,s} Conf(t) = \varnothing$, i.e., $s \in L_U(M)$.

$L_U(M) \subseteq dom(\tau_A)$: Let $s \in L_U(M)$, i.e., let $\{(q_0, 0)\} \models^*_{M,s} \varnothing$. Then, by successive application of (ii), there is a tree $t \in T_\Delta(Att'(O(s)))$ with $Conf(t) = \varnothing$ and $(\langle q_0, 0\rangle, 0) \Rightarrow^*_{A,s} t$. Then $t \in T_\Delta$ (because $Conf(t) = \varnothing$) and $s \in dom(\tau_A)$.

Clearly, if M is deterministic, then A is a monadic. This proves (ii). □

Consider some twa with $(q, \downarrow_k) \in \delta(r, \gamma, i)$ and $(r', \uparrow) \in \delta(q, \sigma, k)$. Then the set R of the patt A constructed in the proof of Lemma 3 contains the following rules.

$$R_\sigma \supseteq \{ \begin{array}{lll} \langle q,k\rangle(0) & \to & \dots\langle \bar{r}', \sigma\rangle(0)\dots, \\ \langle q,\sigma\rangle(k) & \to & \dots\langle r', \sigma\rangle(0)\dots, \\ \langle r,\gamma\rangle(i) & \to & \dots\langle \bar{q}, k\rangle(i)\dots \end{array} \} \qquad R_\gamma \supseteq \{ \begin{array}{lll} \langle r,i\rangle(0) & \to & \dots\langle q, k\rangle(k)\dots, \\ \langle r,\gamma\rangle(i) & \to & \dots\langle \bar{q}, k\rangle(i)\dots, \\ \langle q,\sigma\rangle(k) & \to & \dots\langle r', \gamma\rangle(0)\dots \end{array} \}$$

The top-most two rules are due to condition (1) of the construction in the proof of Lemma 3, and all other rules are due to condition (2).

Lemma 4 (i) $dom(PATT) \subseteq UTWA$ and (ii) $dom(PATT_{\mathrm{mon}}) \subseteq DTWA$.

Proof. First we prove (i). Let $A = (S, I, \Sigma, \Delta, a_0, R)$ be a patt in Bnf and let $m = maxrank(\Sigma)$. We now construct the twa $M = (Q, \Sigma, \delta, q_0, \{f\})$ with $L_U(M) = dom(\tau_A)$. Let $Q = S \cup (I \times [0, m]) \cup \{f\}$, $q_0 = a_0$, and define transition function δ as follows (for technical convenience $a \in S$ is also denoted by $\langle a, 0\rangle$ in the sequel).

For every $j \in [0, m]$, $\sigma \in \Sigma^{(k)}$, $k \geq 0$, and rule $a(\nu) \to r[c_1(i_1), \dots, c_n(i_n)]$ in R_σ, let $\delta(\langle a, \nu\rangle, \sigma, j) = \{(f, \uparrow)\}$ if $n = 0$ (i.e., $t \in T_\Delta$), and otherwise let $\delta(\langle a, \nu\rangle, \sigma, j)$ be the smallest set such that for every $c(i) \in \{c_1(i_1), \dots, c_n(i_n)\}$,

- if $c \in S$, then $(c, \downarrow_i) \in \delta(\langle a, \nu\rangle, \sigma, j)$ and
- if $c \in I$, then $(\langle c, j\rangle, \uparrow) \in \delta(\langle a, \nu\rangle, \sigma, j)$.

Note that $i = 0$ in case $c \in I$, because A is in Bnf. For every $\sigma \in \Sigma^{(k)}$, $k \geq 0$, and rule $b(0) \to r[c_1(0), \dots, c_n(0)]$ in R_{root}, let $\delta(\langle b, 0\rangle, \sigma, 0) = \{(f, \uparrow)\}$ if $n = 0$, and otherwise let $\delta(\langle b, 0\rangle, \sigma, 0) = \{(c_1, \uparrow), \dots, (c_n, \uparrow)\}$. Finally, let $\delta(q, \sigma, i) = \varnothing$ for all q, σ, i not defined above.

Now, for a tree $t \in T_\Delta(Att(O(s)))$, define $Conf(t) = \{(a,u) \mid a \in S, a(u) \text{ occurs in } t\} \cup \{(\langle a,i\rangle, v) \mid a \in I, a(u) \text{ occurs in } t, u = vi \text{ such that if } u \neq 0 \text{ then } i \neq 0\}$.

Claim: Let $s \in T_\Sigma$. Then (i) for all $t, t' \in T_\Delta(Att(O(s)))$, if $t \Rightarrow_{A,s} t'$ then $Conf(t) \models^*_{M,s} Conf(t')$, and (ii) For all $K, K' \subseteq Q \times O(s)$, if $K \models_{M,s} K'$, and there is a $t \in T_\Delta(Att(O(s)))$ such that $Conf(t) = K - \{f\} \times O(s)$, then there is a $t' \in T_\Delta(Att(O(s)))$ with $Conf(t') = K' - \{f\} \times O(s)$ and $t \Rightarrow^*_{A,s} t'$.

Again we omit the proof of this claim (it can be done by straightforward case distinction). The equality $dom(\tau_A) = L_U(M)$ can be proved as follows.

$dom(\tau_A) \subseteq L_U(M)$: Let $s \in dom(\tau_A)$, i.e., let $(\langle q_0, 0\rangle, 0) \Rightarrow^*_{A,s} t$ for some $t \in T_\Delta$. Then, by applying (i) successively we obtain $\{(q_0,0)\} = Conf((\langle q_0,0\rangle,0)) \models^*_{M,s} Conf(t) = \varnothing$, i.e., $s \in L_U(M)$.

$L_U(M) \subseteq dom(\tau_A)$: Let $s \in L_U(M)$, i.e., let $\{(q_0,0)\} \models^*_{M,s} \varnothing$. Then, by (ii), there is a tree $t \in T_\Delta(Att(O(s)))$ such that $(\langle q_0,0\rangle,0) \Rightarrow^*_{A,s} t$ and $Conf(t) = \varnothing - \{f\} \times O(s) = \varnothing$. Then $t \in T_\Delta$ hence $s \in dom(\tau_A)$.

Clearly, if A is monadic, then M is deterministic, which shows (ii). □

Consider some patt with $R_\sigma = \{a'(0) \to \sigma(b'(0), a'(1)), b(1) \to \sigma(b(0), a(1)), b(2) \to a(1)\}$, where σ has rank 2. Applying the construction in the proof of Lemma 4 we obtain, for every $j \in [m]$, $\delta(a', \sigma, j) = \{\langle b', j\rangle, \uparrow), (a', \downarrow_1)\}$, $\delta(\langle b, 1\rangle, \sigma, j) = \{(\langle b, j\rangle, \uparrow), (a, \downarrow_1)\}$, and $\delta(\langle b, 2\rangle, j) = \{(a, \downarrow_1)\}$.

References

[AU71] A. V. Aho and J. D. Ullman. Translations on a context-free grammar. *Inform. and Control*, 19:439–475, 1971.

[Bar81] M. Bartha. An algebraic definition of attributed transformations. In F. Gécseg, editor, *Proc. FCT'81*, volume 117 of *LNCS*, pages 51–60. Springer-Verlag, 1981.

[EHvB99] J. Engelfriet, H. J. Hoogeboom, and J.-P. van Best. Trips on trees. *Acta Cybernet.*, 14:51–64, 1999.

[FHVV93] Z. Fülöp, F. Herrmann, S. Vágvölgyi, and H. Vogler. Tree transducers with external functions. *Theoret. Comput. Sci.*, 108:185–236, 1993.

[Fül81] Z. Fülöp. On attributed tree transducers. *Acta Cybernetica*, 5:261–279, 1981.

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

[Kam83] T. Kamimura. Tree automata and attribute grammars. *Inform. and Control*, 57:1–20, 1983.

[Knu68] D.E. Knuth. Semantics of context-free languages. *Math. Systems Theory*, 2:127–145, 1968. (Corrections in *Math. Systems Theory*, 5:95-96, 1971).

[NS] F. Neven and T. Schwentick. Logical characterization of deterministic tree walking automata. In preperation.