Minimal ascending tree automata

By F. GÉCSEG and M. STEINBY

To the memory of Professor László Kalmár

Here an 'ascending tree recognizer' is a finite, deterministic automaton that reads trees starting at the root proceeding then towards the leaves along all branches. It accepts or rejects the tree depending on the states at which it arrives at the leaves. In the literature they have also been called 'climbing automata', 'top-down tree recognizers' and 'root-to-frontier automata'. They were first studied by MAGIDOR and MORAN [5]. Although various forms of ascending tree transducers have been studied (cf. [3], for example) the ascending tree recognizers have received little attention. A brief discussion can be found in THATCHER'S [7] survey paper.

The minimization of (frontier-to-root) tree recognizers was first considered by BRAINERD [2]. Another formulation was given by ARBIB and GIVE'ON [1]. It turned out that Nerodes theorem (cf. [6]) and the classical minimization algorithms can be extended to them.

In this paper the minimization problem of ascending tree recognizers is studied. First we define some basic algebraic concepts for them (such as homomorphisms). In order to be able to generalize the results and procedures from the case of ordinary recognizers we have to restrict ourselves to 'normalized' ascending tree recognizers. However, every ascending tree recognizer is equivalent to such a normalized recognizer. From a connected normalized ascending tree recognizer a minimal recognizer can be obtained as a quotient recognizer. Also, it turns out that any two equivalent normalized minimal ascending tree recognizers are isomorphic. All steps involved in the process of transforming a given ascending tree recognizer into a minimal one are effective so a minimization algorithm results.

1. Trees and ascending tree recognizers

We shall define trees as polynomial symbols in the sense of GRÄTZER [4]. In this paper

$$F = \bigcup (F_m | m \geqq 1)$$

will be a finite set of *operational symbols*. For any $m \geqq 1$, F_m is the set of m-ary operational symbols and the sets F_m $(m \geqq 1)$ are assumed to be pairwise disjoint. Note that we exclude here 0-ary operational symbols.

For every $n \geqq 1$,

$$X_n = \{x_1, \ldots, x_n\}$$

is a fixed set of *variables* and the set $T_{F,n}$ of *n-ary F-trees* is defined as the smallest set U such that

(1) $X_n \subseteq U$ and

(2) $f(p_1, \ldots, p_m) \in U$ whenever $p_1, \ldots, p_m \in U$ and $f \in F_m$ for some $m > 0$.

Definitions and proofs concerning trees will usually follow the inductive pattern of this definition.

An *n-ary (deterministic) ascending F-recognizer* ($n > 0$) is a system

$$\mathfrak{A} = (A, F, a_0, \mathbf{a}),$$

where

(1) A is the finite nonempty set of *states,*

(2) $a_0 \in A$ the *initial state,*

(3) $\mathbf{a} = (A_1, \ldots, A_n) \in (2^A)^n$ the *final state vector* and

(4) every $f \in F_m$ ($m > 0$) is realized as a mapping

$$f^{\mathfrak{A}}: A \to A^m.$$

Henceforth $\mathfrak{A}$ and $\mathfrak{B}$ will be the n-ary ascending F-recognizers $(A, F, a_0, \mathbf{a})$ and $(B, F, b_0, \mathbf{b})$, respectively. Here $\mathbf{b} = (B_1, \ldots, B_n)$. Since F and n are always given (although arbitrary) we shall often speak about ascending tree recognizers or simply about recognizers.

The operation of $\mathfrak{A}$ can be described as follows. The recognizer begins the examination of a given tree $p \in T_{F,n}$ at the 'root' in its initial state. If the root is labelled by $f \in F_m$, then it has m direct successors which are the roots of the corresponding subtrees and it will continue its operation by examining these subtrees starting in the states $a_1, \ldots, a_m$, respectively, where $(a_1, \ldots, a_m) = f^{\mathfrak{A}}(a_0)$. The process is repeated until $\mathfrak{A}$ has reached the 'leaves' along every branch of the tree. Every leaf is labelled by a variable. If a given leaf is labelled by x_i then $\mathfrak{A}$ should reach it in a state belonging to A_i. The tree p is accepted if this condition is satisfied for every leaf. It is easier to formalize this procedure by tracing it from the leaves back to the root. To this end we define a map

$$\alpha_{\mathfrak{A}}: T_{F,n} \to 2^A$$

as follows:

(1) $\alpha_{\mathfrak{A}}(x_i) = A_i$, for all $x_i \in X_n$, and

(2) $\alpha_{\mathfrak{A}}(p) = \{a \in A \mid f^{\mathfrak{A}}(a) \in \alpha_{\mathfrak{A}}(p_1) \times \ldots \times \alpha_{\mathfrak{A}}(p_m)\}$, if $p = f(p_1, \ldots, p_m)$ with $m > 0$, $f \in F_m$ and $p_1, \ldots, p_m \in T_{F,n}$.

The *forest recognized* by $\mathfrak{A}$ can now be defined as

$$T(\mathfrak{A}) = \{p \in T_{F,n} \mid a_0 \in \alpha_{\mathfrak{A}}(p)\}.$$

The recognizers $\mathfrak{A}$ and $\mathfrak{B}$ are *equivalent* if $T(\mathfrak{A}) = T(\mathfrak{B})$. Furthermore, $\mathfrak{A}$ is called *minimal* if $|B| \geqq |A|$ whenever $\mathfrak{B}$ is a recognizer equivalent to $\mathfrak{A}$.

2. Some algebraic concepts

We shall now adapt some central algebraic notions for n-ary ascending tree automata.

A *homomorphism* of $\mathfrak{A}$ *onto* $\mathfrak{B}$ is a mapping $\varphi: A \to B$ onto B such that

(1) for all $m>0$, $f \in F_m$ and $a \in A$, $f^{\mathfrak{B}}(a\varphi)=(a_1\varphi, \ldots, a_m\varphi)$, where $(a_1, \ldots, a_m)=f^{\mathfrak{A}}(a)$,

(2) $a_0\varphi=b_0$ and

(3) for all $i=1, \ldots, n$, $A_i\varphi=B_i$ and $B_i\varphi^{-1}=A_i$.

If φ is a homomorphism of $\mathfrak{A}$ onto $\mathfrak{B}$, we write $\varphi: \mathfrak{A} \to \mathfrak{B}$ and call $\mathfrak{B}$ a *homomorphic image* of $\mathfrak{A}$. If φ is also bijective, then it is called an *isomorphism*. We say that $\mathfrak{A}$ and $\mathfrak{B}$ are *isomorphic* and write $\mathfrak{A} \cong \mathfrak{B}$ if there exists an isomorphism $\varphi: \mathfrak{A} \to \mathfrak{B}$. Obviously, $\cong$ is a reflexive, symmetric and transitive relation among n-ary F-recognizers.

Let ϱ be an equivalence relation on a set S. Then

(i) s/ϱ is the ϱ-class determined by a given element $s \in S$,

(ii) $\mathbf{s}/\varrho=(s_1/\varrho, \ldots, s_n/\varrho)$, if $\mathbf{s}=(s_1, \ldots, s_n) \in S^n$ $(n \geqq 1)$,

(iii) $U/\varrho=\{u/\varrho \mid u \in U\}$, if $U \subseteq S$ and

(iv) $\mathbf{U}/\varrho=(U_1/\varrho, \ldots, U_n/\varrho)$, if $\mathbf{U}=(U_1, \ldots, U_n) \in (2^S)^n$

A *congruence relation* of the recognizer $\mathfrak{A}$ is now defined as an equivalence relation ϱ on A such that

(1) for all $m>0, f \in F_m$ and $a, a' \in A$, $a/\varrho=a'/\varrho$ implies $f^{\mathfrak{A}}(a)/\varrho=f^{\mathfrak{A}}(a')/\varrho$, and

(2) for all $i=1, \ldots, n$ and $a \in A, a \in A_i$ implies $a/\varrho \subseteq A_i$.

If ϱ is a congruence relation of $\mathfrak{A}$, then the *quotient recognizer* of $\mathfrak{A}$ determined by ϱ is the n-ary F-recognizer

$$\mathfrak{A}/\varrho = (A/\varrho, F, a_0/\varrho, \mathbf{a}/\varrho)$$

where, for all $m>0$, $f \in F_m$ and $a \in A$,

$$f^{\mathfrak{A}/\varrho}(a/\varrho) = f^{\mathfrak{A}}(a)/\varrho.$$

It is easy to see that $\mathfrak{A}/\varrho$ is well-defined. As indicated in the next theorem the three concepts defined above are related to each other the same way their counterparts in algebra are. The straightforward proof is omitted.

Theorem 1. Let $\mathfrak{A}$ and $\mathfrak{B}$ be n-ary ascending F-recognizers.

a) If ϱ is a congruence of $\mathfrak{A}$, then $\mathfrak{A}/\varrho$ is a homomorphic image of $\mathfrak{A}$.

b) If $\varphi: \mathfrak{A} \to \mathfrak{B}$ is a homomorphism of $\mathfrak{A}$ onto $\mathfrak{B}$, then the kernel $\varrho=\varphi\varphi^{-1}$ of φ is a congruence relation of $\mathfrak{A}$ and $\mathfrak{B} \cong \mathfrak{A}/\varrho$.

The following observation will be used later.

Theorem 2. If $\mathfrak{B}$ is a homomorphic image of $\mathfrak{A}$, then $T(\mathfrak{A})=T(\mathfrak{B})$.

Proof. Let $\varphi: \mathfrak{A} \to \mathfrak{B}$ be a homomorphism of $\mathfrak{A}$ onto $\mathfrak{B}$. We show by induction on trees that for any $a \in A$, $a \in \alpha_{\mathfrak{A}}(p)$ iff $a\varphi \in \alpha_{\mathfrak{B}}(p)$.

(1) If $p=x_i \in X_n$, then it holds since $\alpha_{\mathfrak{A}}(p)=A_i$, $\alpha_{\mathfrak{B}}(p)=B_i$ and $\alpha_{\mathfrak{A}}(p)\varphi= \alpha_{\mathfrak{B}}(p)$, $\alpha_{\mathfrak{B}}(p)\varphi^{-1}=\alpha_{\mathfrak{A}}(p)$.

(2) Let $p=f(p_1, \ldots, p_m) \in T_{F,n}$ be such that $\alpha_{\mathfrak{B}}(p_i)=\alpha_{\mathfrak{A}}(p_i)\varphi$ and $\alpha_{\mathfrak{B}}(p_i)\varphi^{-1}= \alpha_{\mathfrak{A}}(p_i)$ $(i=1, \ldots, m)$.

Suppose $a \in \alpha_{\mathfrak{A}}(p)$. If $f^{\mathfrak{A}}(a)=(a_1, \ldots, a_m)$, then $a_1 \in \alpha_{\mathfrak{A}}(p_1), \ldots, a_m \in \alpha_{\mathfrak{A}}(p_m)$. Hence $a_1\varphi \in \alpha_{\mathfrak{B}}(p_1), \ldots, a_m\varphi \in \alpha_{\mathfrak{B}}(p_m)$ which implies

$$f^{\mathfrak{B}}(a\varphi) = (a_1\varphi, \ldots, a_m\varphi) \in \alpha_{\mathfrak{B}}(p_1) \times \ldots \times \alpha_{\mathfrak{B}}(p_m).$$

Thus $a\varphi \in \alpha_{\mathfrak{B}}(p)$.

Suppose now that $a\varphi \in \alpha_{\mathfrak{B}}(p)$ and let $f^{\mathfrak{A}}(a)$ be $(a_1, \ldots, a_m)$. Then $a_1\varphi \in \alpha_{\mathfrak{B}}(p_1), \ldots, a_m\varphi \in \alpha_{\mathfrak{B}}(p_m)$. This implies that

$$a_1 \in \alpha_{\mathfrak{A}}(p_1), \ldots, a_m \in \alpha_{\mathfrak{A}}(p_m).$$

Hence, $a \in \alpha_{\mathfrak{A}}(p)$.

Now $p \in T(\mathfrak{A})$ iff $a_0 \in \alpha_{\mathfrak{A}}(p)$
iff $a_0\varphi = b_0 \in \alpha_{\mathfrak{B}}(p)$
iff $p \in T(\mathfrak{B})$

which completes the proof.

3. Normalized, connected and reduced recognizers

For any state a of the n-ary ascending F-recognizer $\mathfrak{A}$ we put

$$T(\mathfrak{A}, a) = \{p \in T_{F,n} \mid a \in \alpha_{\mathfrak{A}}(p)\}.$$

The state a is called a *0-state* if $T(\mathfrak{A}, a) = \emptyset$.

We say that $\mathfrak{A}$ is *normalized* if, for all $a \in A$, $m > 0$ and $f \in F_m$, either all of the components of $f^{\mathfrak{A}}(a)$ are 0-states or none of them is a 0-state.

For any $\mathfrak{A}$ we define an n-ary ascending F-recognizer

$$\mathfrak{A}^* = (A, F, a_0, \mathbf{a})$$

as follows:

(a) if $\mathfrak{A}$ has no 0-state, then $\mathfrak{A}^* = \mathfrak{A}$ and

(b) if $\mathfrak{A}$ has 0-states choose one of them, say d, and define for all $a \in A$, $m > 0$ and $f \in F_m$

$$f^{\mathfrak{A}^*}(a) = \begin{cases} (d, \ldots, d)\ (\in A^m), & \text{if } f^{\mathfrak{A}}(a) \text{ contains a 0-state} \\ f^{\mathfrak{A}}(a) & \text{otherwise.} \end{cases}$$

Theorem 3. If $\mathfrak{A}$ is any n-ary ascending F-recognizer, then $\mathfrak{A}^*$ is normalized and $T(\mathfrak{A}^*) = T(\mathfrak{A})$.

Proof. We show by tree induction that

$$\alpha_{\mathfrak{A}}(p) = \alpha_{\mathfrak{A}^*}(p), \tag{$*$}$$

for all $p \in T_{F,n}$.

(1) If $p = x_i \in X_n$, then $(*)$ holds since

$$\alpha_{\mathfrak{A}}(p) = A_i = \alpha_{\mathfrak{A}^*}(p).$$

(2) Let $p=f(p_1, \ldots, p_m)$, where $(*)$ holds for $p_1, \ldots, p_m$. Consider any $a \in A$. We have two possible cases:

(i) $f^{\mathfrak{A}}(a)$ contains no 0-state. Then $f^{\mathfrak{A}^*}(a)=f^{\mathfrak{A}}(a)$ and, by the inductive assumption,

$$\begin{aligned} a \in \alpha_{\mathfrak{A}}(p) \quad &\text{iff} \quad f^{\mathfrak{A}}(a) \in \alpha_{\mathfrak{A}}(p_1) \times \ldots \times \alpha_{\mathfrak{A}}(p_m) \\ &\text{iff} \quad f^{\mathfrak{A}^*}(a) \in \alpha_{\mathfrak{A}^*}(p_1) \times \ldots \times \alpha_{\mathfrak{A}^*}(p_m) \\ &\text{iff} \quad a \in \alpha_{\mathfrak{A}^*}(p). \end{aligned}$$

(ii) If $f^{\mathfrak{A}}(a)$ contains a 0-state, then it is easily seen that neither $a \in \alpha_{\mathfrak{A}}(p)$ nor $a \in \alpha_{\mathfrak{A}^*}(p)$ is possible.

The claim $T(\mathfrak{A}^*)=T(\mathfrak{A})$ follows immediately from $(*)$. Also, $(*)$ implies that no new 0-states were introduced in the construction of $\mathfrak{A}^*$ and hence that $\mathfrak{A}^*$ is normalized by its definition. □

We call two states a and a' of $\mathfrak{A}$ *equivalent* and write $a \equiv b(\varrho_{\mathfrak{A}})$ if $T(\mathfrak{A}, a)= T(\mathfrak{A}, a')$.

Clearly, $\varrho_{\mathfrak{A}}$ is an equivalence relation on A, and we call $\mathfrak{A}$ *reduced* if $\varrho_{\mathfrak{A}}$ is the identity relation on A.

Theorem 4. If $\mathfrak{A}$ is normalized then $\varrho_{\mathfrak{A}}$ is a congruence relation and $\mathfrak{A}/\varrho_{\mathfrak{A}}$ is reduced.

Proof. First we show that $\varrho_{\mathfrak{A}}$ is a congruence relation.

(1) Consider any $m>0$, $f \in F_m$ and $a, a' \in A$. Let

$$f^{\mathfrak{A}}(a) = (a_1, \ldots, a_m)$$

and

$$f^{\mathfrak{A}}(a') = (a'_1, \ldots, a'_m)$$

and suppose that $a \equiv a'$ $(\varrho_{\mathfrak{A}})$. Consider any i $(1 \leqq i \leqq m)$ and suppose $p_i \in T(\mathfrak{A}, a_i)$. Then a_i is not a 0-state and therefore none of the states $a_1, \ldots, a_m$ is a 0-state and there exist trees

$$p_1 \in T(\mathfrak{A}, a_1), \ldots, p_{i-1} \in T(\mathfrak{A}, a_{i-1}), \quad p_{i+1} \in T(\mathfrak{A}, a_{i+1}), \ldots, p_m \in T(\mathfrak{A}, a_m).$$

Then

$$f(p_1, \ldots, p_i, \ldots, p_m) \in T(\mathfrak{A}, a) = T(\mathfrak{A}, a')$$

implies $p_i \in T(\mathfrak{A}, a'_i)$. Similarly, $p_i \in T(\mathfrak{A}, a'_i)$ implies $p_i \in T(\mathfrak{A}, a_i)$. Hence $a_i \equiv a'_i$ $(\varrho_{\mathfrak{A}})$.

(2) If $a \in A_i$ and $a \equiv a'(\varrho_{\mathfrak{A}})$, for some $i=1, \ldots, n$ and $a, a' \in A$, then $x_i \in T(\mathfrak{A}, a)=T(\mathfrak{A}, a')$ implies $a' \in A_i$.

Since $\varrho_{\mathfrak{A}}$ is a congruence the quotient recognizer $\mathfrak{A}/\varrho_{\mathfrak{A}}$ can be defined. It is reduced as

$$a/\varrho_{\mathfrak{A}} \equiv a'/\varrho_{\mathfrak{A}} \ (\varrho_{\mathfrak{A}/\varrho_{\mathfrak{A}}}) \quad (a, a' \in A)$$

implies

$$a/\varrho_{\mathfrak{A}} = a'/\varrho_{\mathfrak{A}}$$

since, by Theorem 2,

$$T(\mathfrak{A}, a) = T(\mathfrak{A}/\varrho_{\mathfrak{A}}, a/\varrho_{\mathfrak{A}}) = T(\mathfrak{A}/\varrho_{\mathfrak{A}}, a'/\varrho_{\mathfrak{A}}) = T(\mathfrak{A}, a'). \quad \square$$

Let $a, a' \in A$. We write $a \Rightarrow_{\mathfrak{A}} a'$ if there exist an $m>0$ and an $f \in F_m$ such that a' appears in $f^{\mathfrak{A}}(a)$. The reflexive, transitive closure of the relation $\Rightarrow_{\mathfrak{A}}$ is denoted by $\Rightarrow^*_{\mathfrak{A}}$. If $a \Rightarrow^*_{\mathfrak{A}} a'$, then we say that a' is *reachable* from a. The recognizer $\mathfrak{A}$ is said to be *connected* if every state is reachable from the initial state.

The *connected component*

$$\mathfrak{A}^c = (A^c, F, a_0, \mathbf{a}^c)$$

of $\mathfrak{A}$ is the n-ary ascending F-recognizer defined as follows:

(i) $A^c = \{a \in A \mid a_0 \Rightarrow^*_{\mathfrak{A}} a\}$,

(ii) $\mathbf{a}^c = (A_1 \cap A^c, \ldots, A_n \cap A^c)$ and

(iii) for all $m > 0$ and $f \in F_m$, $f^{\mathfrak{A}^c}$ is defined as the restriction of $f^{\mathfrak{A}}$ to A^c. Clearly, the operations of $\mathfrak{A}^c$ are completely defined.

Lemma 5. Let $\mathfrak{A}$ be any n-ary ascending F-recognizer. Then

(1) $\mathfrak{A}^c$ is connected,

(2) $\mathfrak{A} = \mathfrak{A}^c$ iff $\mathfrak{A}$ is connected,

(3) $T(\mathfrak{A}^c) = T(\mathfrak{A})$ and

(4) if $\mathfrak{A}$ is normalized, then so is $\mathfrak{A}^c$.

Lemma 6. Let $\mathfrak{A}$ and $\mathfrak{B}$ be normalized, $a \in A$, $b \in B$, $m > 0$, $f \in F_m$, $f^{\mathfrak{A}}(a) = (a_1, \ldots, a_m)$ and $f^{\mathfrak{B}}(b) = (b_1, \ldots, b_m)$. If $T(\mathfrak{A}, a) = T(\mathfrak{B}, b)$, then $T(\mathfrak{A}, a_i) = T(\mathfrak{B}, b_i)$, for all $i = 1, \ldots, m$.

The straightforward proofs of these lemmas are omitted.

Theorem 7. Let $\mathfrak{A}$ and $\mathfrak{B}$ be connected, normalized n-ary ascending F-recognizers. Then $T(\mathfrak{A}) = T(\mathfrak{B})$ iff $\mathfrak{A}/\varrho_{\mathfrak{A}} \cong \mathfrak{B}/\varrho_{\mathfrak{B}}$.

Proof. If $\mathfrak{A}/\varrho_{\mathfrak{A}}$ and $\mathfrak{B}/\varrho_{\mathfrak{B}}$ are isomorphic, then $T(\mathfrak{A}) = T(\mathfrak{A}/\varrho_{\mathfrak{A}}) = T(\mathfrak{B}/\varrho_{\mathfrak{B}}) = T(\mathfrak{B})$ by Theorems 1 and 2.

Assume now that $T(\mathfrak{A}) = T(\mathfrak{B})$. We define a mapping

$$\varphi\colon A/\varrho_{\mathfrak{A}} \to B/\varrho_{\mathfrak{B}}$$

by

$$(a/\varrho_{\mathfrak{A}})\varphi = b/\varrho_{\mathfrak{B}} \quad \text{if} \quad T(\mathfrak{A}, a) = T(\mathfrak{B}, b)$$

($a \in A$, $b \in B$). The following steps (i)—(v) show that φ gives the required isomorphism.

(i) $(a/\varrho_{\mathfrak{A}})\varphi$ is defined for all $a/\varrho_{\mathfrak{A}} \in A/\varrho_{\mathfrak{A}}$. Since $\mathfrak{A}$ is connected there exists for every $a \in A$ an integer $k \geqq 0$ and states $a_0, a_1, \ldots, a_k \in A$ such that

$$a_0 \Rightarrow_{\mathfrak{A}} a_1 \Rightarrow_{\mathfrak{A}} \ldots \Rightarrow_{\mathfrak{A}} a_{k-1} \Rightarrow_{\mathfrak{A}} a_k = a.$$

By induction on the length of the shortest such 'derivation' of a it can easily be shown using Lemma 6 that there exists for every $a \in A$ a $b \in B$ such that $T(\mathfrak{A}, a) = T(\mathfrak{B}, b)$.

(ii) φ is well-defined. If $T(\mathfrak{A}, a) = T(\mathfrak{B}, b) = T(\mathfrak{B}, b')$ for some $a \in A$ and $b, b' \in B$, then $b/\varrho_{\mathfrak{B}} = b'/\varrho_{\mathfrak{B}}$.

(iii) φ is injective. Obvious.

(iv) φ is surjective. Repeating the argument used in (i) with the rôles of $\mathfrak{A}$ and $\mathfrak{B}$ reversed we see that there exists for every $b \in B$ an $a \in A$ such that $T(\mathfrak{A}, a) = T(\mathfrak{B}, b)$.

(v) φ is a homomorphism. That φ preserves operations follows from Lemma 6. If $a/\varrho_{\mathfrak{A}} \in A_i/\varrho_{\mathfrak{A}}$ $(1 \leqq i \leqq m)$ and $(a/\varrho_{\mathfrak{A}})\varphi = b/\varrho_{\mathfrak{B}}$, then $x_i \in T(\mathfrak{A}, a) = T(\mathfrak{B}, b)$ implies $b/\varrho_{\mathfrak{B}} \in B_i/\varrho_{\mathfrak{B}}$. Likewise, $(a/\varrho_{\mathfrak{A}})\varphi = b/\varrho_{\mathfrak{B}} \in B_i/\varrho_{\mathfrak{B}}$ implies $a/\varrho_{\mathfrak{A}} \in A_i/\varrho_{\mathfrak{A}}$. □

4. Minimal recognizers and minimization

Suppose $\mathfrak{A}$ is minimal. From Lemma 5 it follows that $\mathfrak{A}$ is connected and from Theorem 3 that we may assume that $\mathfrak{A}$ is normalized. Then $T(\mathfrak{A}/\varrho_{\mathfrak{A}})=T(\mathfrak{A})$ by Theorems 2 and 4. Hence, $\mathfrak{A}$ is reduced.

Conversely, if $\mathfrak{A}$ is connected, normalized and reduced, then it is minimal and every normalized minimal recognizer equivalent to it is also isomorphic to it (Theorem 7).

These facts imply that the following three steps yield for any $\mathfrak{A}$ an equivalent minimal recognizer $\mathfrak{B}$. Moreover, this $\mathfrak{B}$ is normalized and it depends, up to isomorphism, on $T(\mathfrak{A})$ only.

Step 1. Form $\mathfrak{A}^*$.

Step 2. Form $\mathfrak{A}^{*c}$.

Step 3. Form $\varrho_{\mathfrak{A}^{*c}}$ and put $\mathfrak{B}=\mathfrak{A}^{*c}/\varrho_{\mathfrak{A}^{*c}}$.

We shall now verify that these steps are effectively realizable and thus constitute a minimization algorithm for ascending tree recognizers.

Let us define the sets $H_k \subseteq A$, $k=0, 1, \ldots,$ as follows:

(i) $H_0=\{a_0\}$

and, for all $k=0, 1, \ldots,$

(ii) $H_{k+1}=H_k \cup \{a \in A \mid a' \Rightarrow_{\mathfrak{A}} a,$ for some $a' \in H_k\}$. Clearly,

$$H_0 \subseteq H_1 \subseteq H_2 \subseteq \ldots$$

and

$$A^c = \bigcup (H_k \mid k \geq 0).$$

Since $H_{k+1}=H_k$ implies $H_k=H_{k+j}$, for all $j \geq 0$, A^c can be obtained as $H_{|A|-1}$.

For any ascending tree recognizer $\mathfrak{A}$ an equivalent (frontier-to-root) tree recognizer can be constructed (cf. [5] or [7], for example). Thus the questions "$T(\mathfrak{A}, a)=\emptyset$?" and "$T(\mathfrak{A}, a)=T(\mathfrak{A}, a')$?" are decidable. (This could easily be shown directly without any reference to frontier-to-root tree automata.) Hence the 0-states can be found and $\varrho_{\mathfrak{A}}$ can be formed. Thus Steps 1 and 3 are also effective.

These results are summed up in the following theorem.

Theorem 8. A normalized ascending tree recognizer is minimal iff it is connected and reduced. For any ascending tree recognizer there exists an equivalent normalized minimal ascending tree recognizer. This is unique up to isomorphism and it can effectively be constructed.

The reduction of an ascending tree automaton can also be done the same way as ordinary finite recognizers (and tree recognizers in [2]) are reduced. Given $\mathfrak{A}$ we define a sequence of equivalence relations $\varrho_0, \varrho_1, \ldots,$ on A as follows: for any $a, a' \in A$

(i) $a \equiv a'(\varrho_0)$ iff $a \in A_i \Leftrightarrow a' \in A_i$, for all $i=1, \ldots, n$, and for $k=0, 1, \ldots,$

(ii) $a \equiv a'(\varrho_{k+1})$ iff $a \equiv a'(\varrho_k)$ and $f^{\mathfrak{A}}(a)/\varrho_k = f^{\mathfrak{A}}(a')/\varrho_k$ for all $m>0, f \in F_m$.

It is easy to see that $\varrho_{\mathfrak{A}}=\varrho_k$, for some $k<|A|$.

UNIVERSITY OF SZEGED
DEPARTMENT OF COMPUTER SCIENCE
6720 SZEGED, HUNGARY

UNIVERSITY OF TURKU
DEPARTMENT OF MATHEMATICS
20500 TURKU 50, FINLAND

References

[1] ARBIB, M. A. and Y. GIVE'ON, Algebra automata 1: Parallel programming as a prolegomena to the categorical approach, *Information and Control,* v. 12, 1968, pp. 331—345.
[2] BRAINERD, W. S., The minimization of tree-automata, *Information and Control,* v. 13, 1968, pp. 484—491.
[3] ENGELFRIET, J., Bottom-up and top-down tree transformations, A comparison, *Math. Systems Theory,* v. 9, 1975, pp. 198—231.
[4] GRÄTZER, G., *Universal algebra,* Van Nostrand, Princeton, N. J., 1968.
[5] MAGIDOR, M. and G. MORAN, Finite automata over finite trees, *Tech. Rep. Hebrew Univ.,* Jerusalem, No. 30, 1969.
[6] RABIN, M. O. and SCOTT, D., Finite automata and their decision problems, *IBM J. Res. Develop.,* v. 3, No. 2, 1959, pp. 114—125.
[7] THATCHER, J. W., Tree automata: an informal survey, Currents in the theory of computing (A. V. Aho, ed.), Englewood Cliffs, 1973, pp. 143—172.

(Received Jan. 3, 1978)