INFORMATIQUE THÉORIQUE ET APPLICATIONS

WOLFGANG THOMAS

On frontiers of regular trees

Informatique théorique et applications, tome 20, n° 4 (1986), p. 371-381.

<http://www.numdam.org/item?id=ITA_1986__20_4_371_0>

© AFCET, 1986, tous droits réservés.

L'accès aux archives de la revue « Informatique théorique et applications » implique l'accord avec les conditions générales d'utilisation (http://www.numdam.org/legal.php). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

NUMDAM

Article numérisé dans le cadre du programme
Numérisation de documents anciens mathématiques
http://www.numdam.org/

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 20, n° 4, 1986, p. 371 à 381)

ON FRONTIERS OF REGULAR TREES (*)

by Wolfgang THOMAS ([1])

Communicated by J. E. PIN

Abstract. – *In this note, some connections are pointed out between the theory of generalized words as developed by Courcelle and Heilbrunner, the model theory of linear orderings, and Rabin's theory of automata on infinite trees. As an application, a decision problem raised by Courcelle and Heilbrunner (on equivalence of countable word- or order-types) is settled.*

Résumé. – *Dans cet article, on étudie les connections entre la théorie de mots généralisés comme développée par Courcelle et Heilbrunner, la théorie des modèles des ordres linéaires et la théorie de Rabin des automates sur les arbres infinis. On obtient comme application une solution d'un problème de décision proposé par Courcelle et Heilbrunner (sur l'équivalence de types de mots dénombrables).*

1. INTRODUCTION

Given a finite alphabet A, we may identify a nonempty word over A with a map $w: M \rightarrow A$ where M is the domain of a finite linear ordering $(M, <)$. Usually, M is assumed to be an initial segment of the natural numbers; it represents the set of "positions" of the letters of the word. The present paper is concerned with *generalized words* which arise when one considers arbitrary at most countable linear orderings $(M, <)$ instead of finite ones. Familiar examples of such generalized words are ω-words or $\mathbb{Z}$-words.

We need some terminology and conventions. In the sequel we say "countable" to mean "at most countable". A (*generalized*) *word* over the alphabet A is a triple $W=(M, <, w)$ such that $(M, <)$ is a linear ordering, $M \neq \emptyset$, and w a map from M into A. (In [1] such triples are called "arrangements".) If M is finite, resp. countable, we speak of a finite, resp. countable word. Two

(*) Received in February 1985, revised in April 1986.
([1]) Lehrstuhl für Informatik II, Büchel 29-31, D-5100 Aachen, West-Germany.

0296-1598 86/04 371 11/$3.10/© Gauthier-Villars

generalized words $(M, <, w)$ and $(M', <', w')$ are *isomorphic* if there is an order-preserving bijection $\gamma : M \to M'$ such that $w = w' \circ \gamma$. By a (*word-*) *type* we mean an isomorphism type of generalized words. If A is a one-letter alphabet, one can view a generalized word over A as a linear ordering; in this case a word-type becomes an order-type.

Given generalized words $U = (M, <, u)$ and $U', <', u')$, the *concatenation* UU', the ω-*repetition* U^{ω} (also denoted $UU \ldots$) and the ω^*-*repetition* U^{ω^*} (also denoted $\ldots UU$) are defined in the standard way. For example, UU' is the generalized word $(N, \prec, w)$ where

$$N = (\{0\} \times M) \cup (\{1\} \times M'), \qquad w((0, x)) = u(x), \qquad w((1, x)) = u'(x)$$

and $(i, x) \prec (j, y)$ if $i < j$ or ($i = j$ and $x < y$, resp. $x <' y$). Similarly, U^{ω} and U^{ω^*} are defined over $\omega \times M$ using the natural ordering, resp. its reverse, on the set ω of the natural numbers.

The countable words are closely connected with infinite (valued) trees. For simplicity we restrict ourselves to binary trees in this paper. The nodes of a binary tree are given by finite words over $\{l, r\}$ ("left", "right"); thus the partial tree ordering is the prefix relation on $\{l, r\}^*$ and the root of a tree is represented by the empty word ε. An *A-valued tree* is a map $t : \mathrm{dom}(t) \to A$ where $\mathrm{dom}(t) \subset \{l, r\}^*$ is a language closed under prefixes. The *frontier* of t, denoted $\mathrm{fr}(t)$, is defined by $\mathrm{fr}(t) = \{u \in \mathrm{dom}(t) \mid ul, ur \notin \mathrm{dom}(t)\}$. It is linearly ordered by the lexicographic ordering $\prec$ on $\{l, r\}^*$. Hence, for any A-valued tree t, we obtain a countable word $W = (\mathrm{fr}(t), \prec, w)$ where $w : \mathrm{fr}(t) \to A$ is the restriction of t to $\mathrm{fr}(t)$. We call this word the *frontier-word* of t. (It may also be called the "yield" of t. However, to avoid confusion with [11] where "yield" has a different meaning for infinite trees, we use the term "frontier-word".) Conversely, any countable word over A is the frontier-word of some A-valued tree. This is easily inferred from the following two facts: (1) Every countable linear ordering can be embedded in the ordering of the rational numbers, and (2) the latter ordering is isomorphic to the ordering of the frontier of a tree t say with $\mathrm{fr}(t) = (ll \cup r)^* lr$.

The subject of this paper is a certain constructively defined subclass of countable words. They emerged in the work of Courcelle [1] and Heilbrunner [4, 5] as solutions to systems of equations for words.

These systems are of the form

$$\Sigma : \quad x_1 = u_1, \quad x_2 = u_2, \quad \ldots, \quad x_n = u_n$$

where $x_1, \ldots, x_n$ are distinct variables and $u_i \in (\{x_1, \ldots, x_n\} \cup A)^*$ for $i = 1, \ldots, n$. A solution of Σ is an n-tuple of words which (substituted for

$x_1, \ldots, x_n$) satisfies Σ. Courcelle [1] showed how to associate with any system Σ a sequence of n trees such that their frontier-words are a solution of Σ. Heilbrunner [4, 5] presented an algorithm which produces, for any Σ, the explicit definition of n word-types yielding a solution of Σ. This explicit definition is given by "regular expressions", which are finite denotations for certain word-types. Over the alphabet A, the class $\mathrm{Reg}(A)$ of *regular expressions* over A is defined inductively as follows:

(*a*) Every finite word is in $\mathrm{Reg}(A)$;

(*b*) (Concatenation) If r, s are in $\mathrm{Reg}(A)$, so is (rs);

(*c*) (ω- and ω^*-iteration) If r is in $\mathrm{Reg}(A)$, so are r^ω and r^{ω^*};

(*d*) (Shuffle operation) If $r_1, \ldots, r_n$ are in $\mathrm{Reg}(A)$, so is $(r_1, \ldots, r_n)^\eta$.

For cases (*a*), (*b*), (*c*), the word-types denoted by the regular expressions are defined in the natural way, derived from the corresponding operations for words as explained above. It remains to define the word-type denoted by $(r_1, \ldots, r_n)^\eta$. Let us assume that $U_1, \ldots, U_n$ are words of the types denoted by $r_1, \ldots, r_n$ respectively. A generalized word U is of the type denoted by $(r_1, \ldots, r_n)^\eta$ if U is a countable word of the following form: it is composed of segments each of which is isomorphic to some U_i, such that between any two of these segments, before any such segment, and after any such segment there is, for $j=1, \ldots, n$, another segment isomorphic to U_j. A familiar argument of the theory of dense orderings shows that this condition fixes a countable word up to isomorphism (*cf.* [5] or [9], p. 115 ff.).

Example. – Consider nine generalized words $U_1, \ldots, U_9$ and let R be the set of rational numbers which have a decimal expansion $0, d_1 \ldots d_k$ with $d_i \in \{0, \ldots, 9\}$ for $1 \leq i < k$ and $d_k \in \{1, \ldots, 9\}$. If one associates with any such number $0, d_1 \ldots d_k$ the word U_{d_k}, the usual ordering of R induces a generalized word (composed of U_i-copies) which is a shuffling of $U_1, \ldots, U_9$.

Let $\mathscr{M}(A)$ be the class of word-types denoted by regular expressions in $\mathrm{Reg}(A)$. Two regular expressions are called *equivalent* if they denote the same word-type. It is easy to see that different regular expressions may be equivalent. [As an example consider the expressions $(ab)^{\omega^*}(ab)^\omega$ and $(ba)^{\omega^*}(ba)^\omega$.] Courcelle [1] and Heilbrunner [5] raised the question whether the equivalence of regular expressions is decidable. (In [1] the question is stated in terms of frontiers of trees. In this framework, a modified version of the problem has meanwhile also been considered by Dauchet and Timmermann [3], [11], [12]; there the notion of frontier-word is replaced by the coarser operation of "yield" of a tree).

One aim of the present note is to answer the above question affirmatively. The proof uses a decidability result in the model theory of linear orderings.

For readers not familiar with model theory we give this argument in some detail. Another aim is to clarify the relation between generalized words and trees: We show that a word has a type in $\mathscr{M}(A)$ iff it is (isomorphic to) the frontier-word of an A-valued *regular tree*. As a consequence we obtain, using Rabin's theory of automata on infinite trees, that a countable word-type belongs to $\mathscr{M}(A)$ iff it is characterizable in the monadic second-order language of orderings.

2. ISOMORPHISM OF GENERALIZED WORDS

The class $\mathscr{M}(A)$ generalizes an important class of order types in model theory. Restricting to a one letter alphabet one obtains from $\mathscr{M}(A)$ the class $\mathscr{M}$ of order types introduced by Läuchli and Leonard in [6]; they showed that a sentence of the first-order language of linear orderings is satisfiable by some linear ordering iff it is satisfiable by a linear ordering whose type is in $\mathscr{M}$, obtaining as a consequence that the first-order theory of linear orderings is decidable. (For a detailed exposition *cf.* [9].)

In the following we make also use of the monadic second-order language of linear orderings and the stronger theorem, due to Rabin [7], that the monadic second-order theory of countable linear orderings is decidable. In order to decide equivalence of expressions in $\mathrm{Reg}(A)$, we simply characterize the types in $\mathscr{M}(A)$ by monadic second-order sentences and then apply Rabin's decidability theorem.

For this purpose, we fix an alphabet $A=\{a_1, \ldots, a_k\}$ and identify any word with a relational model. Namely, the word $(M, <, w)$ is represented by the structure $(M, <, P_1, \ldots, P_k)$ where $P_i \subset M$ is defined by $P_i = w^{-1}(a_i)$. By the obvious correspondence between words and these word-models we use the letters $U, V, W, \ldots$ to denote either of them. The isomorphism relation holds between two models $W=(M, <, P_1, \ldots, P_k)$ and $W'=(M', <', P'_1, \ldots, P'_k)$ iff there is an order-preserving bijection $\gamma: M \to M'$ such that $\gamma(P_i)=P'_i$ for $i=1, \ldots, k$.

Let us introduce the monadic second-order language $L_2(A)$ which allows to describe word-models corresponding to the alphabet $A=\{a_1, \ldots, a_k\}$. $L_2(A)$ contains variables $x, y, \ldots$ (for elements of orderings) and variables $X, Y, \ldots$ (for subsets of orderings). The atomic formulas are of the form $x<y$, $x=y$, $x \in X$, $x \in P_i$ $(1 \leqq i \leqq k)$; and arbitrary $L_2(A)$-formulas are combined from these by the propositional connectives $\neg$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$ and the quantifiers $\exists$, $\forall$ (acting on either kind of variable).

Formulas without free variables are called sentences. The satisfaction relation $\models$ between word-models U and $L_2(A)$-formulas φ is defined in the standard way; one writes $(M, <, P_1, \ldots, P_k) \models \varphi[Q]$ (for a subset Q of M) if the $L_2(A)$ formula $\varphi(X)$ with free variable X holds in $(M, <, P_1, \ldots, P_k)$ when interpreting X by Q.

Let $W_2(A)$ be the class of countable word types t that are characterized by some $L_2(A)$-sentence φ (in the sense that a countable word satisfies φ iff it is of type t).

PROPOSITION 1. — *$\mathcal{M}(A) \subset W_2(A)$; in others words: For every regular expression r over A one can construct an $L_2(A)$-sentence ψ_r such that for any countable word W, $W \models \psi_r$ iff W is of the type denoted by r.*

Proof. — We shall find, for any regular expression r, an $L_2(A)$-formula $\varphi_r(X)$ such that for any countable word-model $(M, <, P_1, \ldots, P_k)$ and any $Q \subset M$, we have $(M, <, P_1, \ldots, P_k) \models \varphi_r[Q]$ iff the submodel of $(M, <, P_1, \ldots, P_k)$ with universe Q is of type r. Then we are done: Given r one takes $\exists X(\forall x\, x \in X \wedge \varphi_r(X))$ as the desired sentence ψ_r.

If r is finite, say $r = a_{j_1} \ldots a_{j_m}$, we take as $\varphi_r(X)$ the formula

$$\exists x_1 \ldots \exists x_m \left(\bigwedge_{i=1}^{m} x_i \in X \wedge \forall x \left(x \in X \rightarrow \bigvee_{i=1}^{m} x = x_i \right) \right.$$
$$\left. \wedge \bigwedge_{i=1}^{m-1} x_i < x_{i+1} \wedge \bigwedge_{i=1}^{m} x_i \in P_{j_i} \right)$$

which clearly characterizes the type (denoted by) r.

Given r, s and corresponding formulas $\varphi_r(X)$, $\varphi_s(X)$, the concatenation (rs) is characterized by the following formula:

$$\exists Y \exists Z (\text{“}X = Y \cup Z\text{”} \wedge \text{“}Y < Z\text{”} \wedge \varphi_r(Y) \wedge \varphi_s(Z)).$$

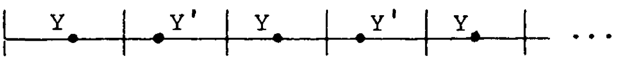
(Here and in the sequel we freely use shortwritings like "$X = Y \cup Z$", "$X < Y$", etc. if their formalization in $L_2(A)$ is straightforward.)

For the remaining steps it is convenient to use a formula "$X' = \mathrm{Comp}(x, X)$" which says that X' is the "component of x with respect to X", namely the largest interval (in the given model) that contains x and consists solety of elements from X: So $X' = \mathrm{Comp}(x, X)$ abbreviates

$$\forall z (z \in X' \leftrightarrow (z \in X \wedge \forall y (x \leqq y \leqq z \vee z \leqq y \leqq x \rightarrow y \in X))).$$

Now assume that $\varphi_r(X)$ characterizes the countable words of type r.

To obtain a characterization for r^{ω}, we use a formula $\varphi(X)$ which says that X can be cut into an ω-sequence of segments each of which satisfies φ_r. We represent this sequence in the form



using two sets Y, Y' and an auxiliary set Z (indicated by the dots).

The required segments occur then as components of the Z-elements, alternatively with respect to Y and Y'. As formula for r^{ω} we take

$$\begin{aligned}
&\exists Z \exists Y \exists Y' (\text{“}Z \text{ is of type } \omega\text{”} \wedge \text{“}\min(Z) \in Y\text{”} \\
&\qquad \wedge \text{“}X = Y \cup Y'\text{”} \wedge \text{“}Y \cap Y' = \varnothing\text{”} \\
&\qquad \wedge \forall x \exists z \exists C (z \in Z \wedge (C = \mathrm{Comp}(z, Y) \vee C = \mathrm{Comp}(z, Y')) \wedge x \in C) \\
&\qquad \wedge \forall z \forall z' (\text{“}z' \text{ is the } <\text{-successor of } z \text{ in } Z\text{”} \rightarrow (z \in Y \leftrightarrow z' \in Y')) \\
&\qquad \wedge \forall z (z \in Z \rightarrow \forall C (C = \mathrm{Comp}(z, Y) \vee C = \mathrm{Comp}(z, Y') \rightarrow \varphi_r(C))).
\end{aligned}$$

Note that the condition “Z is of type ω” is formalizable in $L_2(A)$ by expressing “Z is ordered discretely with first but without last element such that the induction axiom holds”.—The case r^{ω^*} is handled similarly.

Finally let $\varphi_1(X), \ldots, \varphi_n(X)$ be given characterizing $r_1, \ldots, r_n$ in the class of countable words. To characterize $(r_1, \ldots, r_n)^{\eta}$ in the class of countable words, we require the existence of a partition $X_1, \ldots, X_n$ of X such that the X_i-segments given by this partition satisfy φ_i, and the density conditions in the definition of the shuffle operation are fulfilled:

$$\begin{aligned}
&\exists X_1 \ldots \exists X_n \Big(\bigwedge_{i=1}^{n} \text{“}X_i \neq \varnothing\text{”} \wedge \text{“}X = X_1 \cup \ldots \cup X_n\text{”} \wedge \bigwedge_{i \neq j} \text{“}X_i \cap X_j = \varnothing\text{”} \\
&\qquad \wedge \bigwedge_{i=1}^{n} \forall y \forall C (C = \mathrm{Comp}(y, X_i) \rightarrow \varphi_i(C)) \\
&\qquad \wedge \forall x \forall z (x < z \wedge \text{“}x, z \text{ are on different components”} \\
&\qquad\qquad \rightarrow \bigwedge_{i=1}^{n} \exists y (x < y < z \wedge y \in X_i)) \\
&\qquad \wedge \forall x \bigwedge_{i=1}^{n} \exists y (x < y \wedge y \in X_i)) \\
&\qquad \wedge \forall x \bigwedge_{i=1}^{n} \exists y (y < x \wedge y \in X_i)).
\end{aligned}$$

Here "x, z are on different components" is expressed by

$$\bigvee_{i,j} \exists C \exists C' (C = \mathrm{Comp}(x, X_i) \wedge C' = \mathrm{Comp}(z, X_j) \wedge C < C'). \quad \blacksquare$$

COROLLARY. — *The equivalence problem for regular expressions [denoting word-types in $\mathscr{M}(A)$] is decidable.*

Proof. — Given two regular expressions r_1 and r_2, one constructs the corresponding $L_2(A)$-sentences ψ_1 and ψ_2 as in Proposition 1 and obtains that r_1 is equivalent to r_2 iff

(1) for all countable words W : $W \models \psi_1 \leftrightarrow \psi_2$, i. e., iff

(2) for all countable linear orderings $(M, <)$ and all partitions $P_1, \ldots, P_k$ of M :

$$(M, <, P_1, \ldots, P_k) \models \psi_1 \leftrightarrow \psi_2.$$

Let us replace the constants P_i in ψ_1 and ψ_2 by new variables, say $X_1, \ldots, X_k$, which yields formulas $\psi_1'(X_1, \ldots, X_k)$, $\psi_2'(X_1, \ldots, X_k)$.

Then (2) is equivalent to the condition that the $L_2(A)$-sentence

$$\forall X_1 \ldots \forall X_k (\forall x \text{ "} x \in X_1 \cup \ldots \cup X_k \text{"} \wedge \bigwedge_{i \neq j} \text{"} X_i \cap X_j = \varnothing \text{"} \rightarrow (\psi_1'(X_1, \ldots, X_k) \leftrightarrow \psi_2'(X_1, \ldots, X_k)))$$

holds in every countable linear ordering $(M, <)$, i. e. belongs to the monadic second-order theory of countable linear orderings. Hence, by the decidability of this theory [7], the equivalence of r_1 and r_2 can be tested in an effective way.

3. REGULAR TREES

In the previous section we described an application of model theory to language theory. The present section deals with an argument in the reverse direction; it offers the proof of a model- theoretic result (namely, of a converse to Proposition 1) by an automaton-theoretic method. The connection is provided by a characterization of the $\mathscr{M}(A)$-types in terms of regular trees.

Given an alphabet $A = \{a_1, \ldots, a_k\}$, an A-valued tree $t : \mathrm{dom}(t) \rightarrow A$ (where $\mathrm{dom}(t) \subset \{l, r\}^*$) will be called *regular* if it can be generated by a finite automaton on the binary tree. Here a *finite automaton* is of the form $\mathscr{A} = (Q, q_0, \delta, Q_0, \ldots, Q_k)$ where Q is the finite set of states, $q_0 \in Q$ the initial state, $\delta : Q \rightarrow Q \times Q$ the transition function, and $Q_0, \ldots, Q_k$ a partition of Q. For any such automaton there is a unique run of $\mathscr{A}$ on the (unlabelled) binary tree, given as a function $\rho : \{l, r\}^* \to Q$ satisfying $\rho(\varepsilon) = q_0$ and $\delta(\rho(w)) = (\rho(wl), \rho(wr))$, for $w \in \{l, r\}^*$. An A-valued tree t is *generated by* $\mathscr{A}$ if for $w \in \{l, r\}^*$ we have: $t(w) = a_i$ iff $\rho(w) \in Q_i (i = 1, \ldots, n)$, and $w \notin \operatorname{dom}(t)$ iff $\rho(w) \in Q_0$. It is easy to see that our definition of regular trees (as those generated by finite automata) is equivalent to other familiar definitions, e. g. that a tree is regular iff it has only finitely many distinct subtrees (*cf.* [2]).

Denote by $\mathscr{F}(A)$ the class of word types that are given by frontier-words of regular A-valued trees.

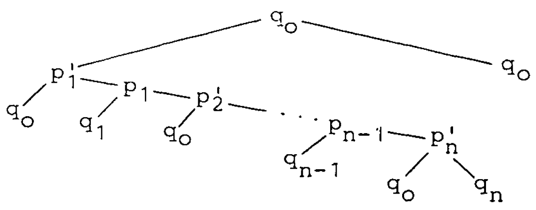
PROPOSITION 2. — $\mathscr{M}(A) = \mathscr{F}(A)$.

Proof. — The inclusion from left to right is shown by induction over $\mathscr{M}(A)$. First, a nonempty finite word is the frontier-word of a finite tree and hence of a regular tree. — If $W = UV$, where U and V are assumed to be frontier-words of trees generated by $\mathscr{A}_U$ and $\mathscr{A}_V$, respectively, then a tree with frontier-word W is generated by a finite automaton $\mathscr{A}$ whose state-set is the union of the two given (w. l. o. g. disjoint) state-sets together with a new initial state q_0, such that $\mathscr{A}$'s transition function δ is the union of the two given transition functions and $\delta(q_0) = (q_U, q_V)$ where q_U, q_V are the initial states of $\mathscr{A}_U$, $\mathscr{A}_V$. The letter $a_i (i \geqq 1)$ associated with the new state q_0 (by setting $q_0 \in Q_i$) is of course arbitrary. — Now let U be given as the frontier-word of the tree generated by $\mathscr{A}_U$ (again with initial state q_U); we find a finite automaton $\mathscr{A}$ generating a tree with frontier-word $U_\omega = UU\ldots$: Adjoin a new state q_0 (as initial state of $\mathscr{A}$) and extend the transition function δ of $\mathscr{A}_U$ by setting $\delta(q_0) = (q_U, q_0)$. [Similarly, for the word U^{ω^*} we let $\delta(q_0) = (q_0, q_U)$.] Finally, suppose we are given finite automata $\mathscr{A}_1, \ldots, \mathscr{A}_n$ (with disjoint state-sets and initial states $q_1, \ldots, q_n$) which generate trees with the frontier-words $U_1, \ldots, U_n$. A finite automaton $\mathscr{A}$ for a tree whose frontier-word is a shuffling of $U_1, \ldots, U_n$ as defined in the introduction is obtained as follows: Adjoin new states $q_0, p_1, \ldots, p_{n-1}, p'_1, \ldots, p'_n$ and define the transition function δ of $\mathscr{A}$ as the union of the given transition functions together with the requirements

$$\delta(q_0) = (p'_1, p_0),\ \delta(p'_i) = (q_0, p_i),\ \delta(p_i) = (q_i, p'_{i+1}) \qquad (\text{for } 1 \leqq i < n),$$

$$\delta(p'_n) = (q_0, q_n).$$

The unique run of $\mathscr{A}$ starts in the following way:



It is easy to verify that the frontier-word of this tree is a shuffling of $U_1, \ldots, U_n$.

The converse direction of Proposition 2 can be obtained from Heilbrunner's theorem [4]. If $\mathscr{A}=(Q, q_0, \delta, Q_0, \ldots, Q_k)$ is a finite automaton, associate with it a system $\Sigma(\mathscr{A})$ of equations for words, where Q is the set of variables, by including in the system:

– the equation $q=q' q''$ for any q such that $\delta(q)=(q', q'')$ where $q' \notin Q_0$, $q'' \notin Q_0$;

– the equation $q=q'$ (resp. $q=q''$) for any q such that $\delta(q)=(q', q'')$ and $q' \notin Q_0$, $q'' \in Q_0$ (resp. $q' \in Q_0$, $q'' \notin Q_0$);

– the equation $q=a_i$ for any $q \in Q_i$ such that $\delta(q)=(q', q'')$ where $q' \in Q_0$ and $q'' \in Q_0$.

Then a countable word is a solution of $\Sigma(\mathscr{A})$ iff it is (isomorphic to) the frontier-word of the tree generated by $\mathscr{A}$. Heilbrunner's algorithm yields a regular expression denoting the type of a solution of $\Sigma(\mathscr{A})$. Hence the frontier-word of the tree generated by $\mathscr{A}$ is of a type in $\mathscr{M}(A)$. ■

Note that in Proposition 2 the transition from regular expressions [for $\mathscr{M}(A)$-types] to corresponding automata (defining regular trees) is effective, as is the converse direction using Heilbrunner's algorithm. Hence, by Proposition 1:

COROLLARY. – *The problem whether two regular trees have isomorphic frontier-words is decidable.*

In [6] Rabin suggested the use of regular trees in model theory. Using Proposition 2 we give here such a model theoretic application by proving $W_2(A) \subset \mathscr{M}(A)$, i.e. the converse to Proposition 1. The argument is based on Rabin's results concerning tree automata that recognize *sets* of trees. Rabin's tree automata accept A-valued trees with domain $\{l, r\}^*$ (i.e., where the nodes form the full binary tree). In order to handle the case of arbitrary trees over $A=\{a_1, \ldots, a_k\}$ we introduce a letter a_0 and use it as value for the nodes outside the domain of an A-valued tree. We call a set of (arbitrary)

A-valued trees *Rabin-recognizable* if the corresponding set of $(A \cup \{a_0\})$-valued trees is recognized by a Rabin tree automaton (as defined in [7]). Also we use the monadic second-order language for $(A \cup \{a_0\})$-valued trees. It is defined as $L_2(A)$ in Section 1 for linear orderings, with the exception that it includes, instead of the symbol $<$ for the order relation, two symbols $+l$, $+r$ for the successor functions on $\{l, r\}^*$; moreover a set constant P_0 (for the nodes with value a_0) is added. Note that the partial tree ordering and the lexicographic (linear) ordering $\prec$ of the nodes are both definable in this language (*cf.* [7]). Call a set T of A-valued trees *monadic second-order definable* if there is a sentence φ of this language such that $t \in T$ iff the corresponding $(A \cup \{a_0\})$-valued tree satisfies φ.

We shall apply the following results due to Rabin:

(1) A set of A-valued trees is Rabin-recognizable iff it is monadic second-order definable [7].

(2) Any nonempty Rabin-recognizable set of A-valued trees contains a regular tree [8].

(A third key result, proved in [7], states that the emptiness problem for Rabin-tree automata is decidable; it implies the decidability of the monadic second-order theory of the (unlabelled) binary tree and, as a corollary, of the monadic second-order theory of countable linear orderings.)

PROPOSITION 3. — *$W_2(A) \subset \mathscr{M}(A)$, i.e. a countable word W over A which is characterized up to isomorphism (in the class of countable words) by an $L_2(A)$-sentence is of a type in $\mathscr{M}(A)$.*

Proof. — Let φ be an $L_2(A)$-sentence characterizing the isomorphism type of W (in the class of countable words). Rewrite φ as a sentence $\bar{\varphi}$ in the monadic second-order language of $(A \cup \{a_0\})$-valued trees, such that $\bar{\varphi}$ defines the set T of trees whose frontier-words satisfy φ. (For this purpose, $<$ has to be replaced by the definition of $\prec$ and all quantifiers of φ have to be relativized to the frontier, i.e. to the set of nodes where both successors carry a_0 as value.) By (1), T is Rabin-recognizable. Since $T \neq \varnothing$, (2) applies and yields a regular tree $t \in T$. By Proposition 2, the frontier-word of t has a type in $\mathscr{M}(A)$. But by assumption on φ, this frontier-word is isomorphic to W.

In model-theoretic terminology, Propositions 1 and 3 say that (with respect to the monadic second-order language) a generalized word satisfies an $\aleph_0$-categorical sentence iff its type is in $\mathscr{M}(A)$. For the case of linear orderings (i.e. the case of singleton alphabets) this result was first stated by Shelah in [10], Theorem 0.2; a proof hint is given there saying that the techniques in

Section 5 of the paper yield the result. The present approach using tree automata and regular trees provides an interesting (and perhaps more transparent) alternative for the proof. Moreover, it seems possible to extend the model theoretic application of regular trees beyond linear orderings to further structures which are also embeddable in the binary tree.

REFERENCES

1. B. Courcelle, *Frontiers of Infinite Trees*, RAIRO Inform. Theor., Vol. 12, 1978, pp. 319-337.
2. B. Courcelle, *Fundamental Properties of Infinite Trees*, Theor. Comput. Sci., Vol. 25, 1983, pp. 95-169.
3. M. Dauchet and E. Timmerman, *Decidability of Yield's Equality for Infinite Regular Trees*, in: Automata on Infinite Words, M. Nivat and D. Perrin Eds., Springer Lect. Notes in Comput. Sci., Vol. 192, 1984, pp. 118-136.
4. S. Heilbrunner, *Gleichungssysteme für Zeichenreihen*, Technische Universität München, Abt. Math., Bericht Nr. 7311, 1973.
5. S. Heilbrunner, *An Algorithm for the Solution of Fixed-point Equations for Infinite Words*, RAIRO Inform. Théor., Vol. 14, 1980, pp. 131-141.
6. H. Läuchli and J. Leonard, *On the Elementary Theory of Linear Order*, Fund. Math., Vol. 59, 1966, pp. 109-116.
7. M. O. Rabin, *Decidability of Second-order Theories and Automata on Infinite Trees*, Trans. Amer. Math. Soc., Vol. 141, 1969, pp. 1-35.
8. M. O. Rabin, *Automata on Infinite Objects and Church's Problem*, Reg. Conf. Ser. in Math. No 13, Amer. Math. Soc., Providence, R.I., 1972.
9. J. G. Rosenstein, *Linear Orderings*, Academic Press, New York, 1982.
10. S. Shelah, *The Monadic Theory of Order*, Ann. Math., Vol. 102, 1975, pp. 379-419.
11. E. Timmerman, *Yields of Infinite Trees*, in: *Ninth Colloquium on Trees in Algebra and Programming*, B. Courcelle Ed, Cambridge Univ. Press, 1984, pp. 299-312.
12. M. Dauchet and E. Timmerman, RAIRO Inform. Theor. (to appear).