H. SPERBER (Erlangen)

FINITE AUTOMATA ACCEPTING ANIMALS AND LOWER SETS
(Summary)

INTRODUCTION

We consider finite automata which are equipped with n-ary transfer relations. Such automata can operate on very general structures, cf. [1]. The action on (labelled) trees has widely been investigated, cf. e.g. [2], [3], [8], [9]. The well known constructions and results of (string) automata theory neatly carry over to tree automata. Our interest is in the operation on (labelled) structures embedded in ω^n, the so-called *(directed) animals*, which can be thought of as a "commutative version" of (regular) trees. Now, some results still carry over (see "*positive*" below), but we are also confronted with phenomena (e.g. the undecidability of the emptiness problem), which are characteristic for 2-dimensional automata concepts. All the "*negative*" results can be obtained in the 2-dimensional, unlabelled case.

DEFINITIONS

1. (n,Σ)-automata

Let Σ be a finite alphabet and n a positive integer. A=(Q,T,i,F) is a *(n,Σ)-automaton* :<=> Q is a finite set ("states"), $T \subset Q^n \times \Sigma \times Q$ ("*transfer relation*"), $i \in Q$ ("*initial state*"), $F \subset Q$ ("*final states*"). Elements of T will be denoted by p--σ->q. A is *deterministic* :<=> for every pair (p,σ) there is at most one state q such that p--σ->q is in T.

2. Animals

Let $\omega = \{0,1,2,\ldots.\}$, $succ_k$ = the k-th successor function in ω^n. For $x,y \in X \subset \omega^n$, $x \leqslant_X y$:<=> there is a succ-path from x to y, which is completely contained in X. X is an ***animal*** :<=> X is finite and $x \in X$ implies $0 \leqslant_X x$. X is a ***lower set*** :<=> X is finite and $y \leqslant x \in X$ implies $y \in X$. If X is an animal, every $v : X \dashrightarrow \Sigma$ is called a Σ-animal. $\{\sigma\}$-animals and animals are identified. Any substitution $\pi : \Sigma \dashrightarrow \Gamma$ induces a mapping from Σ-animals to Γ-animals, which is called a ***projection***.

$$\begin{array}{ccccccc}
\sigma & \tau & \tau & & & & \\
\tau & & & \sigma & \sigma & \sigma & \\
\tau & \tau & \tau & \sigma & & & \\
\sigma & & & \tau & & & \\
\sigma & & \tau & \tau & \sigma & & \sigma \\
\tau & \tau & \tau & \tau & & & \tau \\
\sigma & \sigma & \tau & \tau & \sigma & \sigma & \tau
\end{array}
\qquad\qquad
\begin{array}{cccccccc}
\star & \star & & & & & & \\
\star & \star & \star & & & & & \\
\star & \star & \star & & & & & \\
\star & \star & \star & \star & & & & \\
\star & \star & \star & \star & \star & \star & \star & \\
\star & \star & \star & \star & \star & \star & \star & \star
\end{array}$$

A $\{\sigma,\tau\}$-animal A lower set

3. Recognition

For an animal X, bor(X) ("border") consists of all $y \in \omega^n \setminus X$, which are successor of some $x \in X$. bor($\emptyset$) := {0}. Let A=(Q,T,i,F) and $v : X \dashrightarrow \Sigma$. A Q-animal c with domain $X \cup$ bor(X) is an ***accepting A-computation on v*** :<=>

$c(y) = i$ for all $y \in$ bor(X),

$(c(succ_1 y),\ldots,c(succ_n y))\text{--}v(y)\text{->}c(y) \in T$ for all $y \in X$,

$c(0) \in F$.

The ***language of A*** is the set of all v, on which an accepting A-computation exists. A set of Σ-animals is ***(deterministically) recognizable*** if it is the language of some (deterministic) (n,Σ)-automaton. $REC(n,\Sigma)$, $REC^D(n,\Sigma)$ denote the class of (deterministically) recognizable sets.

$$
\begin{array}{ccccccc}
\star & & & \star & \star & & \\
\star & \star & \star & \star & & & \\
\star & & & \star & \star & & \\
\star & & & \sigma & \star & \star & \star \\
\star & \star & \star & \star & & &
\end{array}
\qquad\qquad
\begin{array}{cccccccc}
i & & & i & i & & & \\
\ast & i & i & \ast & \ast & i & & \\
\ast & \ast & \ast & \ast & i & & & \\
\ast & i & & q & \ast & i & i & \\
\ast & i & & r & p & \ast & \ast & i \\
\ast & \ast & \ast & \ast & i & & &
\end{array}
$$

The principle of computations (let $(p,q)\text{--}\sigma\text{->}r \in T$)

4. Local sets

A *neighbourhood* is a nonempty lower set. Let Ξ be a neighbourhood, X an animal, $x \in \omega^n$ and $v : X \text{-->} \Sigma$. $p_{x,\Xi}(v)$ denotes the Σ-animal $w : Y \text{-->} \Sigma$ with $Y = \{y \in \Xi \mid x \leqslant_X x+y\}$ and $w(y) = v(x+y)$ for all $y \in Y$. $\mathrm{parts}_\Xi(v)$:= the set of all $p_{x,\Xi}(v)$, where $x \in X$. A set of Σ-animals α is Ξ-*local* :<=> there are sets R ("*allowed root labels*") and I ("*allowed internal parts*"), where $R \subset \Sigma$ and I consists of Σ-animals, whose domains are contained in Ξ, such that a nonempty Σ-animal v is in α iff $v(0) \in R$ and $\mathrm{parts}_\Xi(v) \subset I$. α is *local*, if it is Ξ-local for some Ξ.

$$
\text{e. g. the }
\begin{array}{cc}
\star & \star \\
\star & \star \\
\star & \star
\end{array}
\text{ - parts of }
\begin{array}{cccc}
\sigma & & & \\
\tau & \tau & \tau & \\
\sigma & & \sigma & \sigma \\
\sigma & \sigma & \tau &
\end{array}
\text{ are:}
$$

$$
\begin{array}{cc}
\tau & \tau \\
\sigma & \\
\sigma & \sigma
\end{array},
\begin{array}{cc}
 & \tau \\
 & \sigma \\
\sigma & \tau
\end{array},
\begin{array}{cc}
\tau & \\
\sigma & \sigma \\
\tau &
\end{array},
\begin{array}{cc}
\sigma & \\
\tau & \tau \\
\sigma &
\end{array},
\begin{array}{cc}
 & \\
\tau & \\
\sigma & \sigma
\end{array},
\begin{array}{c}
 \\
 \\
\sigma
\end{array},
\begin{array}{cc}
 & \\
\sigma & \\
\tau & \tau
\end{array},
\begin{array}{cc}
 & \\
 & \\
\tau & \tau
\end{array},
\begin{array}{c}
 \\
 \\
\tau
\end{array}.
$$

EXAMPLE

An animal X is a tree :<=> for all $x \in X$, the succ-path within X from 0 to x is unique. Let $T(n)$:= the set of all trees in ω^n. $T(n)$ is the language of the (nondeterministic) automaton

$A = (Q,T,i,Q)$, where

$Q = \{1,2,\ldots,n\} \cup \{i\}$ and

T consists of all

$$(p_1,p_2,\ldots,p_n) \dashrightarrow q$$

which satisfy the conditions

$p_k \in \{i, k\} \qquad (k = 1,\ldots,n)$,

$q \in \{1,2,\ldots,n\}$.

For n=2, an accepting A-computation on

```
                              i i i
* * *                         2 1 1 i
*     *                       2 i   2 i
*   * * * *        is         2 i 2 1 1 1 i
*   *                         2 i 2 i i
* * * * *                     1 1 1 1 1 i      .
```

RESULTS

"positive"

1. $REC(n,\Sigma)$ is closed under $\cup$, $\cap$.
2. $REC^D(n,\Sigma)$ is closed under $\cup$, $\cap$, $\setminus$.
3. Every finite set is deterministically recognizable.
4. Images of recognizable sets under projections are recognizable.
5. Every recognizable set is the projection of some local set.
6. Every local set is recognizable.
7. The recognizable sets are exactly the projections of the local sets.

"negative"

For all $n \geq 2$ and every Σ,

8. $REC(n,\Sigma)$ is <u>not</u> closed under complementation.
9. $REC^D(n,\Sigma)$ is a <u>proper</u> subset of $REC(n,\Sigma)$.
10. 9. remains true, if inputs are restricted to lower sets.
11. The emptiness problem "Given an arbitrary deterministic (n,Σ)-automaton, is its language empty?" is <u>undecidable</u>.

Remarks on proofs:

1 to 5: Slight extensions of the standard techniques known for tree atomata (cf. [3]) are used. *6:* The states of the automaton are - roughly speaking - the allowed internal parts characterizing the local set. *7:* Follows from 4 & 5 & 6. *8:* The complement of $T(2)$ is not recognizable. A pumping argument is used. *9:* Follows from 2 & 8. *10:* The set α of triangles of length 2^j in ω^2 is not det. recognizable, since $\{\sigma^k \mid k=2^j\} \subset \sigma^*$ is not regular. Looking at Pascal's triangle mod 2, a local set with projection α can be given. *11:* The undecidable problem "Given an arbitrary Turing machine, does it halt on the empty input tape?" is reduced to the emptiness problem.

<u>REMARKS</u>

1. The details and some other results will be contained in [7].
2. More on lower set automata (relativized emptiness problems, connections to contextsensitive languages) can be found in [6].
3. In the case $|\Sigma| \geq 2$, $n=2$, result 10 is a consequence of a theorem in [4]. This article also deals with connections between $(2,\Sigma)$-automata (operating on rectangles) and other 2-dimensional automata concepts.
4. When inputs are restricted to rectangles, the undecidability of the emptiness problem follows from a theorem in [5].
5. The animals defined here are "directed animals with exactly one source" in the sense of [10].

REFERENCES

[1] M. A. ARBIB, Y. GIVE'ON, Algebra automata I: parallel programming as a prolegomena to the categorical approach, *Information and Control 12(1968)*, 331-345.

[2] J. DONER, Tree acceptors and some of their applications, *J. Comput. System Sci. 4(1970)*, 406-451.

[3] F. GECSEG, M. STEINBY, "Tree automata", Akademiai Kiado, Budapest 1984.

[4] K. INOUE, A. NAKAMURA, Some properties of two-dimensional on-line tesselation acceptors, *Information Sciences 13(1977)*, 95-121.

[5] A. NAKAMURA, On causal ω^2-systems, *J. Comput. System Sci. 10(1975)*, 253-265.

[6] H. SPERBER, "Idealautomaten" (Dissertation), Erlangen 1985.

[7] H. SPERBER, Finite automata accepting animals and lower sets, in preparation.

[8] J. W. THATCHER, Tree automata: an informal survey, in "Currents in the Theory of Computing" (A. V. Aho ed.), 143-172, Prentice-Hall, Englewood Cliffs, New Jersey, 1973.

[9] J. W. THATCHER, J. B. WRIGHT, Generalized finite automata theory with an application to a decision problem of second-order logic, *Mathematical Systems Theory 2(1968)*, 57-81.

[10] G. VIENNOT, Problèmes combinatoires posés par la physique statistique, *Seminaire Bourbaki, exposé n° 626 (1984)*.

Helmut Sperber
IMMD (Informatik) I
Martensstr. 3
D-8520 Erlangen
WEST GERMANY