Acta Cybernetica, Tom. 9, Fasc. 4, Szeged, 1990

Recognizable sets of finite bilabelled transition systems

A. Arnold
Laboratoire d'Informatique *
Université Bordeaux I

Abstract

In this note we propose a definition of a notion of automaton recognizing finite bilabelled transition systems, i.e. finite directed graphs with labels attached to both vertices and edges. The family of recognizable sets is a boolean algebra. Moreover every recognizable set contains the images and the inverse images of each of its element under surjective homomorphisms.

1 Introduction

Recognizable sets of finite words and of finite trees are defined by mean of automata: a recognizable set consists in all the elements recognized by such an automaton. Since words are special kinds of trees, the notion of tree automaton is an extension of the notion of word automaton. But trees are special kinds of directed acyclic graphs (*dags*, for short) and indeed a notion of dag automaton was recently introduced [3], which is an extension of the notion of tree automaton.

Since dags are special kinds of graphs, one can think of a notion of graph automaton which is an extension of the notion of dag automaton. However there is a major distinction between dags and graphs: when an automaton reads in a dag it goes from vertices to vertices along the directed edges, and, because of acyclicity, it never reads in twice the same vertex; therefore it is possible to assign a unique state of the automaton to every newly read in vertex according to the states assigned to previous vertices as specified by the transition function of the automaton. For graphs, the situation is different, since a vertex can be read in several times and then the state assigned to a vertex can change during the computation of the automaton.

But it is possible to interpret the results of Büchi on infinite words [4], and of Rabin on infinite trees [8], as an intuitive support to the thesis that there is a close connection between the notion of recognizability and the notion of definability by some monadic second order logic (see for instance [5]). Therefore one can investigate for a characterization of the set of graphs which are models of a given monadic second order formula in terms of automata.

*Unité de Recherche associée au Centre National de la Recherche Scientifique n° 1304

Since Branching Time Temporal Logics as well as the μ-calculus [6] are special cases of monadic second order logics which are used to express properties of processes, and since processes are usually represented by finite transition systems, one can expect to get a notion of automaton which recognizes those transition systems satisfying some given temporal properties. Indeed, the μ-calculus already gives a precious hint on what a graph automaton could be: when computing the value of a formula over a graph, the boolean value of each subformula is computed at every vertex of the graph; hence one can see every vector of boolean values of these subformulas as a possible state of the automaton. This set of states is naturally ordered by a partial order. This led us to consider as the set of states of a graph automaton a finite partially ordered set. Provided this set has a minimal element and the transition function is monotonic, one can define a *computation* of such an automaton in the following way: initially the minimal state is assigned to every vertex; due to the monotonicity of the transition function, when the state assigned to a vertex has to change it can only increase; when the states assigned to the vertices cannot be increased, the computation ends. In other words the assigment reached at the end of the computation is the least fixed-point of some mapping associated with the transition function. Let us remark that in the case of infinite trees, a *run* is sometimes defined as an assigment of states to nodes which satisfies some relations determined by the transition relation of the automaton.

In the first part we consider the simple case of complete deterministic bilabelled transition systems. They are defined as a set of vertices with a label attached to each vertex and with a mapping from the set of vertices into itself associated with each element of some alphabet. We define our notion of automaton in this simple case. In the second part, we consider the more general case where, with each element of the alphabet is associated a mapping from the set of vertices into its powerset. In the third part we define the product of automata and in then we define the acceptance criteria for these automata and give some properties of recognizable sets: the family of recognizable sets is a boolean algebra and each recognizable set is closed under surjective homomorphisms and inverse surjective homomorphisms. Therefore, every recognizable set is a union of *fibers*, where a fiber is a set of inverse surjective homomorphic images of one bilabelled transition system, and we show that a fiber is a recognizable set. The automata previously defined are deterministic and the sets they recognized are called "deterministically recognizable"; we define also "nondeterministically recognizable sets" as beeing the projections of deterministically recognizable ones.

This note consists mainly in definitions although some elementary open questions remain to be answered. But the main question raised by this definition of recognizable sets is the following: it is well known that, in the cases of words and of trees, there exist special kinds of grammars (the regular grammars) which generates exactly the recognizable sets; thus, among the large number of kinds of graphs grammars already introduced in the litterature, do there exist some kinds of them which can generate exactly the determistically or non deterministically recognizable sets of bilabelled transition systems.

2 A simple case

2.1 Complete deterministic bilabelled transition systems

A complete deterministic bilabelled transition system over an alphabet A and a set L of labels is a tuple $S = < V, \gamma, \lambda >$ where

- V is a finite set of vertices,

- γ is a mapping from $A \times V$ into V; $\gamma(a, v)$ is the unique vertex of V which is reachable from v by an edge labelled by a,
- λ is a mapping from V into L.

2.2 Deterministic cdbts automata

A deterministic cdbts automaton is a tuple $S =< A, L, Q, q_0, \delta >$ where

- A is an alphabet and L is a set of labels,
- Q is a finite partially ordered set of states, and q_0 is its minimal element,
- δ, the transition function, is a monotonic mapping from $L \times Q^A$, ordered componentwise, into Q.

Given a cdbts $S =< V, \gamma, \lambda >$ over A and L and a deterministic automaton $\mathcal{A} =< A, L, Q, q_0, \delta >$, the set $\mathbf{A}$ of *assignments* is the ordered set Q^V, the minimal element of which is $\{q_0\}^V$. The mapping δ is extended into a monotonic mapping δ_S from $\mathbf{A}$ into $\mathbf{A}$ defined as follows:

Let α be an element of $\mathbf{A} = Q^V$; let v be an element of V; its *environment under* α is the tuple $\text{env}_S^\alpha(v) =< \lambda(v), \alpha(\gamma(a_1, v)), \ldots, \alpha(\gamma(a_n, v)) >$, where $A = \{a_1, \ldots, a_n\}$, which is an element of $L \times Q^A$ and $\delta(\text{env}_S^\alpha(v))$ is an element of Q. Then $\delta_S(\alpha)$ is the element of $\mathbf{A}$ defined by $\delta_S(\alpha)(v) = \delta(\text{env}_S^\alpha(v))$.

Obviously the mapping δ_S is monotononic, and since $\mathbf{A}$ has a minimal element, δ_S has a least fixed point in $\mathbf{A}$ which will be denoted by $\mu\delta_S$.

The following example mainly shows in which sense deterministic word automata are a special kind of cdbts automata. This construction can easily be extended to bottom-up deterministic tree automata.

Example 1. Let us consider a usual deterministic word automaton $\mathcal{A}$ over an alphabet L with Q as a set of states.

With a word $u = l_1 \ldots l_n$ we associate the cdbts S_u over a single-letter alphabet and with $L \cup \{\vdash\}$ as set of labels defined as follows:

- the set of vertices is $\{0, 1, \ldots, n\}$,
- there is an edge from i to $i - 1$ (with $0 - 1 = 0$)
- 0 is labelled with $\vdash$, and i, for $1 \leq i \leq n$ is labelled with l_i,

Then we define a cdbts automaton over a single-letter alphabet and the set of labels $L \cup \{\vdash\}$ by

- the set of states is $Q \cup \{\perp\}$, where $\perp$ is less than any other element,
- the transition function is defined by
  - $\delta(\vdash, q)$ is the initial state of Q, for every q in $Q \cup \{\perp\}$,
  - $\delta(l, \perp) = \perp$ for every l in L,
  - $\delta(l, q)$ is the state obtained by applying the transition function of the deterministic word automaton to $l \in L$ and $q \in Q$.

It is easy to see that in the assignment which is the least fixed point of δ_{S_u}, the initial state of $\mathcal{A}$ is assigned to the vertex 0 and the state reached by $\mathcal{A}$ reading $l_1 \ldots l_i$ and starting in the initial state is assigned to the vertex i. □

2.3 Homomorphisms of cdbts

Let $S_1 =< V_1, \gamma_1, \lambda_1 >$ and $S_2 =< V_2, \gamma_2, \lambda_2 >$ be two cdbts. An *homomorphism* from S_1 into S_2 is a mapping $h : V_1 \longrightarrow V_2$ such that $h(\gamma_1(a, v)) = \gamma_2(a, h(v))$ and $\lambda_1(v) = \lambda_2(h(v))$.

If $\mathcal{A} =< A, L, Q, q_0, \delta >$ is a cdbts automaton over A and L, let $\mu\delta_{S_1}$ and $\mu\delta_{S_2}$ be the least fixed points of δ_{S_1} and δ_{S_2}.

For every assignment β in Q^{V_2}, $\beta \circ h$, defined by

$$\beta \circ h(v) = \beta(h(v))$$

is an assignment in Q^{V_1}. In particular we have:

Lemma 2.1 *If h is a homomorphism from S_1 into S_2, and if β is an assignment in Q^{V_2}, then for any v in V_1,*

$$\text{env}^{\beta}_{S_2}(h(v)) = \text{env}^{\beta \circ h}_{S_1}(v)$$

Proof By definition of env we have

$$\text{env}^{\beta}_{S_2}(h(v)) =< \lambda_2(h(v)), \beta(\gamma_2(a_1, h(v))), \ldots, \beta(\gamma_2(a_n, h(v))) >$$

and

$$\text{env}^{\beta \circ h}_{S_1} =< \lambda_1(v), \beta(h(\gamma_1(a_1, v))), \ldots, \beta(h(\gamma_1(a_n, v))) >$$

Since h is an homomorphism, the two right-hand sides of these equations are equal, hence the result. □

It follows

Lemma 2.2 *If h is an homomorphism from S_1 into S_2, and if β is an assignment in Q^{V_2}, then*

$$\delta_{S_1}(\beta \circ h) = \delta_{S_2}(\beta) \circ h$$

Proof For any v in V_1 we have

$$\delta_{S_1}(\beta \circ h)(v) = \text{env}^{\beta \circ h}_{S_1}(v)$$

and

$$\delta_{S_2}(\beta) \circ h(v) = \delta_{S_2}(\beta)(h(v)) = \text{env}^{\beta}_{S_2}(h(v))$$

and the result immediately follows from Lemma 2.1 □

Then we get

Theorem 2.1 *If h is an homomorphism from S_1 into S_2, then*

$$\mu\delta_{S_1} = \mu\delta_{S_2} \circ h$$

Proof Let us denote by $\perp_k$, for $k = 1, 2$ the least mapping in Q^{V_k} which associates the minimal element q_0 of Q with every vertex v of V_k.

α, the least fixed point of δ_{S_1}, is the limit of the increasing sequence $(\alpha_i)_{i\geq 0}$ defined by $\alpha_0 = \perp_1$ and $\alpha_{i+1} = \delta_{S_1}(\alpha_i)$. Similarly, β, the least fixed point of δ_{S_2}, is the limit of the increasing sequence $(\beta_i)_{i\geq 0}$ defined by $\beta_0 = \perp_2$ and $\beta_{i+1} = \delta_{S_2}(\beta_i)$.

We prove by induction that $\alpha_i = \beta_i \circ h$.

Obviously

$$\beta_0 \circ h = \perp_2 \circ h = \perp_1 = \alpha_0$$

and, by Lemma 2.2,

$$\beta_{i+1} \circ h = \delta_{S_2}(\beta_i) \circ h = \delta_{S_1}(\beta_i \circ h)$$

which is equal, by induction hypothesis, to

$$\delta_{S_1}(\alpha_i) = \alpha_{i+1}$$

□

3 A more general case

We want to extend the previous definition of automata to the case of non deterministic bilabelled transition systems (bts). In this case $\gamma(a, v)$ is no longer a single vertice, but a set (possibly empty) of vertices. Hence the following definition:

3.1 Non deterministic bilabelled transition systems

A non deterministic bilabelled transition system over an alphabet A and a set L of labels is a tuple $S = < V, \gamma, \lambda >$ where

- V is a finite set of vertices,
- γ is a mapping from $A \times V$ into $\wp(V)$; $\gamma(a, v)$ is the set of vertices of V which are reachable from v by an edge labelled by a,
- λ is a mapping from V into L.

3.2 The powerset of states

If Q is a partially ordered set of states with a minimal element $\perp$, we can still define the image of a set of vertices under an assignment in Q^V, but, for the mapping δ being monotonic we need some partial order on $\wp(Q)$. We choose to use the so called *Egli-Milner preorder* [7] defined by

$$\begin{aligned} X \sqsubseteq Y \text{ iff } \forall x \in X, \exists y \in Y : x &\leq y \\ \text{and } \forall y \in Y, \exists x \in X : x &\leq y. \end{aligned}$$

This preorder has the following properties:

Proposition 3.1

- *The empty set is not comparable with any non empty set.*
- *The set $\{\perp\}$ is less than any non empty set.*
- *If α and β are two assignments in Q^V, and if U is a set of vertices, then $\alpha \leq \beta$ implies $\alpha(U) \sqsubseteq \beta(U)$*

3.3 Deterministic bts automata

A deterministic bts automaton is a tuple $\mathcal{S} =< A, L, Q, q_0, \delta >$ where

- A is an alphabet and L is a set of labels,
- Q is a finite partially ordered set of states, and q_0 is its minimal element, $\wp(Q)$ being ordered by the Egli-Milner preorder,
- δ, the transition function, is a monotonic mapping from $L \times \wp(Q)^A$, ordered componentwise, into Q.

Given a bts $S =< V, \gamma, \lambda >$ over A and L and a deterministic automaton $\mathcal{A} = < A, L, Q, q_0, \delta >$, the set $\mathbf{A}$ of *assignments* is the ordered set Q^V, the minimal element of which is $\{q_0\}^V$. The mapping δ is extended into a monotonic mapping δ_S from $\mathbf{A}$ into $\mathbf{A}$ defined as in the previous case, where the notion of environment is modified to take into account the fact that the bilabelled transition system S is no longer deterministic:

Let α be an element of $\mathbf{A} = Q^V$; let v be an element of V; its *environment under* α is the tuple $\mathrm{env}_S^\alpha(v) =< \lambda(v), \alpha(\gamma(a_1, v)), \ldots, \alpha(\gamma(a_n, v)) >$, which is an element of $L \times \wp(Q)^A$ and $\delta(\mathrm{env}_S^\alpha(v))$ is an element of Q. Then $\delta_S(\alpha)$ is the element of $\mathbf{A}$ defined by $\delta_S(\alpha)(v) = \delta(\mathrm{env}_S^\alpha(v))$.

Obviously the mapping δ_S is monotononic, and since $\mathbf{A}$ has a minimal element, δ_S has a least fixed point in $\mathbf{A}$ which will be denoted by $\mu\delta_S$.

3.4 Homomorphisms of bts

Let $S_1 =< V_1, \gamma_1, \lambda_1 >$ and $S_2 =< V_2, \gamma_2, \lambda_2 >$ be two bts. An *homomorphism* from S_1 into S_2 is a mapping $h : V_1 \longrightarrow V_2$ such that the sets $h(\gamma_1(a, v))$ and $\gamma_2(a, h(v))$ are equal and $\lambda_1(v) = \lambda_2(h(v))$.

This definition of an homomorphism is related to the notion of bisimulation of transition systems [1]: two transition systems (all vertices of which having the same label) are in the bisimulation relation if and only if they have a common image under two surjectives homomorphisms.

It is clear from the definitions of homomorphisms and of environments that the Lemma 2.1 remains true (its proof does not change) and also the Lemma 2.2 and the Theorem 2.1.

Theorem 3.1 *If h is an homomorphism from S_1 into S_2, then*

$$\mu\delta_{S_1} = \mu\delta_{S_2} \circ h$$

3.5 Other extensions

The value of the transition function δ of an automaton at some vertex v of a transition system S under an assignment α depends on the environment of v under α, which consists in the label of v and the image under α of some sets of vertices associated with v. One can imagine others transition functions which depend on larger environments of a vertex. Here we present two such extensions. In the first one δ take into account the state assigned to the vertex and we show that this extension is not more powerful. In the second one we put in the environment of v under α the sets $\gamma^{-1}(a, v)$, equal to $\{v' | v \in \gamma(a, v)\}$. If one considers bts automata as bottom-up automata because states are propagated back along the edges (the state of v depends on the states of $\gamma(a, v)$), then top-down automata are those where only the sets $\gamma^{-1}(a, v)$ are in the environments, instead of the sets $\gamma(a, v)$, and, when both are in the environment, we get something like a bidirectional automaton.

3.5.1 Vertices need not to be in their environment.

Here we assume that the transition function of an automaton is a monotonic mapping from $Q \times L \times \wp(Q)^A$, ordered componentwise, into Q. Then $\delta_S(\alpha)$ is the element of $\mathbf{A}$ defined by $\delta_S(\alpha)(v) = \delta(\alpha(v), \mathrm{env}_S^\alpha(v))$.

Let us define the mapping δ^* from $L \times \wp(Q)^A$ into Q by:
$\delta^*(l, X_{a_1}, \ldots, X_{a_n})$ is the limit of the sequence $(q_i)_{i\leq 0}$ defined by

- q_0 is the least element of Q,
- $q_{i+1} = \delta(q_i, l, X_{a_1}, \ldots, X_{a_n})$.

This sequence is increasing and we have

$$\delta^*(l, X_{a_1}, \ldots, X_{a_n}) = \delta(\delta^*(l, X_{a_1}, \ldots, X_{a_n}), l, X_{a_1}, \ldots, X_{a_n})$$

Let α and β be the least fixed points of δ_S and of δ_S^*. Then we can show that $\alpha = \beta$.

From the definitions of δ_S^*, δ_S, and by the previous equality, we have

$$\begin{aligned}
\beta(v) &= \delta_S^*(\beta)(v) \\
&= \delta^*(\mathrm{env}_S^\beta(v)) \\
&= \delta(\delta^*(\mathrm{env}_S^\beta(v)), \mathrm{env}_S^\beta(v)) \\
&= \delta(\beta(v), \mathrm{env}_S^\beta(v)) \\
&= \delta_S(\beta)(v)
\end{aligned}$$

and then α is less than β.

Conversely, let us define β as the limit of the increasing sequence $(\beta_i)_{i\leq 0}$ with $\beta_0 = \perp$ and $\beta_{i+1} = \delta_S^*(\beta_i)$. We prove by induction that β_i is less than α.

Obviously, β_0 is less than α.

Since $\beta_{i+1} = \delta_S^*(\beta_i)$, since, by induction hypothesis, β_i is less than α, and since δ_S^* is monotonic, we get

$$\beta_{i+1} \leq \delta_S^*(\alpha)$$

and it remains to prove:

$$\delta_S^*(\alpha) \leq \alpha$$

For any v we have

$$\delta_S^*(\alpha)(v) = \delta^*(\mathrm{env}_S^\alpha(v))$$

and let $\delta^*(\mathrm{env}_S^\alpha(v))$ be the limit of the increasing sequence $(q_i)_{i\leq 0}$ with $q_0 = \perp$ and $q_{i+1} = \delta(q_i, \mathrm{env}_S^\alpha(v))$. We prove by induction that $q_i \leq \alpha(v)$, hence the result.

q_0 is less than $\alpha(v)$. Let us assume that q_i is less than α. Then $q_{i+1} = \delta(q_i, \mathrm{env}_S^\alpha(v))$, and since δ is monotonic, $q_{i+1} \leq \delta(\alpha(v), \mathrm{env}_S^\alpha(v))$. But

$\delta(\alpha(v), \mathrm{env}_S^\alpha(v)) = \delta_S(\alpha)(v) = \alpha(v)$, which ends the proof.

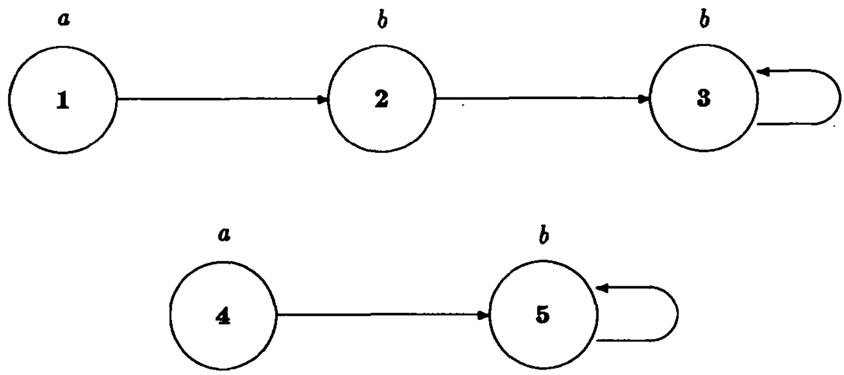

Figure 1.

3.5.2 Bidirectional automata

Let us consider the two following cdbts pictured in figure 1 with $\{a, b\}$ as set of labels and a single-letter alphabet.

Let us consider the set $Q = \{q_0, q_1, q_2\}$ with $q_0 \leq q_1 \leq q_2$, and the function δ from $\{a, b\} \times \wp(Q) \times \wp(Q)$ into Q, where the second argument of δ corresponds to γ^{-1} and the third one to γ, defined by:

$$\begin{aligned} \delta(a, X, Y) &= q_0, \forall X, Y \subseteq Q \\ \delta(b, \{q_0\}, Y) &= q_1, \forall Y \subseteq Q \\ \delta(b, X, Y) &= q_2, \forall X, Y \subseteq Q, X \neq \{q_0\} \end{aligned}$$

Then, on the first bts of the figure 1, the assigment α which is the least fixed point of δ is

$$\begin{aligned} \alpha(\mathbf{1}) &= q_0 \\ \alpha(\mathbf{2}) &= q_1 \\ \alpha(\mathbf{3}) &= q_2 \end{aligned}$$

and for the second one, the least assignment β is

$$\begin{aligned} \beta(\mathbf{4}) &= q_0 \\ \beta(\mathbf{5}) &= q_2 \end{aligned}$$

On the other hand, the mapping which sends **1** on **4** and **2** and **3** on **5** is a surjective bts homomorphism, and because of Theorem 2.1, the least assigment α and β of a "bottom-up" deterministic bts automaton should satisfy $\alpha(\mathbf{1}) = \beta(\mathbf{4})$ and $\alpha(\mathbf{2}) = \alpha(\mathbf{3}) = \beta(\mathbf{5})$, which shows up that bidirectional bts automata are more powerful than bottom-up bts automata.

4 Products of deterministic bts automata

Let $\mathcal{A}_1 =< A, L, Q_1, i_1, \delta_1 >$ and $\mathcal{A}_2 =< A, L, Q_2, i_2, \delta_2 >$, where

$$\delta_i : L \times \wp(Q_i)^A \longrightarrow Q_i$$

for $i = 1, 2$, be two deterministic bts automata over the alphabet A and the labels L.

The product $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$ of $\mathcal{A}_1$ and $\mathcal{A}_2$ is the deterministic bts automaton $< A, L, Q, i, \delta >$ with

$$Q = Q_1 \times Q_2,$$

$$i =< i_1, i_2 >,$$

$$\delta : L \times \wp(Q_1 \times Q_2)^A \longrightarrow Q_1 \times Q_2 \text{ defined by}$$

$$\delta(l, P_{a_1}, \ldots, P_{a_n}) =< \delta_1(l, \pi_1(P_{a_1}), \ldots, \pi_1(P_{a_n})), \delta_2(l, \pi_2(P_{a_1}), \ldots, \pi_2(P_{a_n})) >$$

where π_i is the canonical projection of $Q_1 \times Q_2$ onto Q_i.

Indeed, it is straightforward from its definition that δ is monotonic because it is very easy to prove that if $P \sqsubseteq P'$, $P, P' \subseteq Q_1 \times Q_2$, then $\pi_i(P) \sqsubseteq \pi_i(P')$.

Let us consider a bts $\mathcal{S} =< V, \gamma, \lambda >$. If α is an assigment in $(Q_1 \times Q_2)^V$, it can be seen as the product $\alpha_1 \times \alpha_2$ of the assignments α_i in Q_i^V defined by $\alpha_1 \times \alpha_2(v) = < \alpha_1(v), \alpha_2(v) >$. It follows:

$$\delta_{\mathcal{S}}(\alpha) = \delta_{1\mathcal{S}}(\alpha_1) \times \delta_{2\mathcal{S}}(\alpha_2)$$

and the least fixed point $\mu\delta_{\mathcal{S}}$ of δ is equal to the product $\mu\delta_{1\mathcal{S}} \times \mu\delta_{2\mathcal{S}}$ of the least fixed points of $\delta_{1\mathcal{S}}$ and $\delta_{2\mathcal{S}}$.

Theorem 4.1 *If* $\delta = \delta_1 \times \delta_2$ *then* $\mu\delta_{\mathcal{S}} = \mu\delta_{1\mathcal{S}} \times \mu\delta_{2\mathcal{S}}$

5 Acceptance criterion

In order to have a notion of recognizable set of bts we have to define an acceptance criterion for the bts automata. The criterion we choose is such that boolean combinations of recognizable sets are still recognizable, which is a very natural assumption.

5.1 Definitions

Let $\mathcal{A} =< A, L, Q, i, \delta >$ be a bts automaton. An acceptance criterion for $\mathcal{A}$ is a set $\mathcal{F}$ of subsets of Q.

Given a bts $\mathcal{S} =< V, \gamma, \lambda >$ and the least fixed point $\mu\delta_{\mathcal{S}}$ in Q^V of δ, we say that $\mathcal{S}$ is accepted by the pair $< \mathcal{A}, \mathcal{F} >$ iff $\mu\delta_{\mathcal{S}}(V)$ is an element of $\mathcal{F}$. A recognizable set is the set of all bts accepted by some pair $< \mathcal{A}, \mathcal{F} >$.

5.2 Properties of recognizable sets

Let $B(A, L)$ the set of all bts over the alphabet A and the labels L.

The following property is a straightforward consequence of the definitions:

Lemma 5.1 *If R is a subset of $B(A, L)$ recognized by the pair $< \mathcal{A}, \mathcal{F} >$ then its complement is recognized by the pair $< \mathcal{A}, \wp(Q) - \mathcal{F} >$.*

Let us consider two bts automata $\mathcal{A}_j =< A, L, Q_j, i_j, \delta_j >$ for $j = 1, 2$, and two acceptance criteria $\mathcal{F}_j$ in $\wp(Q_j)$. Let us denote by $\mathcal{G}_\cup$ (resp. $\mathcal{G}_\cap$) the set of all subsets F of $Q_1 \times Q_2$ such that $\pi_1(F) \in \mathcal{F}_1$ or (resp. and) $\pi_2(F) \in \mathcal{F}_2$ where π_j is the canonical projection of $Q_1 \times Q_2$ on Q_j.

Lemma 5.2 *A bts is accepted by the pair $< \mathcal{A}_1 \times \mathcal{A}_2, \mathcal{G}_\cup >$ iff it is accepted by the pair $< \mathcal{A}_1, \mathcal{F}_1 >$ or by the pair $< \mathcal{A}_2, \mathcal{F}_2 >$.*

A bts is accepted by the pair $< \mathcal{A}_1 \times \mathcal{A}_2, \mathcal{G}_\cap >$ iff it is accepted by the pair $< \mathcal{A}_1, \mathcal{F}_1 >$ and by the pair $< \mathcal{A}_2, \mathcal{F}_2 >$.

Proof We consider only the first case; the second one is proved exactly the same way.

From the definition of the product of automata we have (Theorem 4.1)

$$\mu\delta_S = \mu\delta_{1S} \times \mu\delta_{2S}$$

Hence $\pi_i(\mu\delta_S(V)) = \mu\delta_{iS}(V)$, and $\mu\delta_S(V)$ belongs to $\mathcal{G}_\cup$ iff $\mu\delta_{1S}(V)$ belongs to $\mathcal{F}_1$ or $\mu\delta_{2S}(V)$ belongs to $\mathcal{F}_2$. □

Theorem 5.1 *The set of all recognizable subsets of $B(A, L)$ is a boolean algebra.*

Proof By lemma 5.1, this set is closed under complement. By lemma 5.2 the union and the intersection of the two subsets recognized by the pairs $< \mathcal{A}_1, \mathcal{F}_1 >$ and $< \mathcal{A}_2, \mathcal{F}_2 >$ are recognized by the pairs $< \mathcal{A}_1 \times \mathcal{A}_2, \mathcal{G}_\cup >$ and $< \mathcal{A}_1 \times \mathcal{A}_2, \mathcal{G}_\cap >$.
□

A less usual property is:

Proposition 5.1 *Let $h : S_1 \longrightarrow S_2$ be a* surjective *homomorphism of bts. Then S_1 is accepted by the pair $< \mathcal{A}, \mathcal{F} >$ iff S_2 is accepted by this pair.*

Proof We already know, by Theorem 3.1 that $\mu\delta_{S_1} = \mu\delta_{S_2} \circ h$. Hence $\mu\delta_{S_1}(V) = \mu\delta_{S_2}(h(V)) = \mu\delta_{S_2}(V') = Q$. □

6 Fibers

Let us define the following binary relation between bts: two bts are in the relation if they have a common homomorphic image. It can be proved (see [1]) that this is an equivalence relation and, moreover, that each equivalence class has a canonical representant which is minimal in the following sense: it is an homomorphic image of every bts in its equivalence class. Therefore we call *fiber* such an equivalence class. From proposition 5.1 every recognizable set is a union of fibers. Here we prove that **every fiber is recognizable**.

Let $S =< V, \gamma, \lambda >$ be a bts. Let us define, for every vertex v and for every natural number n the bts $T(v, n)$, which is indeed a tree, as follows:

- $T(v, 0)$ is a single vertex labelled by a special symbol, say $\perp$,

- $T(v, n+1)$ is the disjoint union of all $T(v', n)$ for all v' in $\bigcup_{a\in A} \gamma(a, v)$ together with a new vertex v_0 labelled by $\lambda(v)$ and, for any letter a in A, a set of edges labelled by a from v_0 to every $T(v', n)$, for v' in $\gamma(a, v)$.

For short, $T(v, n+1)$ can be written as $\lambda(v)(\bigcup_{a\in A} \bigcup_{v'\in\gamma(a,v)} a \cdot T(v', n))$.

The set of all such $T(v, n)$ can be ordered, by induction on n, by:

- $T(v, 0)$ is less than anything,
- $T(v, n+1)$ is less than $T(v', n'+1)$, for $n \leq n'$ iff
  - $\lambda(v) = \lambda(v')$,
  - for any a in A, the set $\{T(v'', n) \mid v'' \in \gamma(a, v)\}$ is less (with respect to the induced Egli–Milner ordering) than the set $\{T(v'', n') \mid v'' \in \gamma(a, v')\}$.

Lemma 6.1 *If* $T(v, n+1) = T(v', n+1)$, *then* $T(v, n) = T(v', n)$.

Proof The proof is by induction on n.
•If $T(v, 1) = t(v', 1)$, then obviously $T(v, 0) = T(v', 0) = \perp$.

•If $T(v, n+2) = T(v', n+2)$, then $\lambda(v) = \lambda(v')$ and for any a, $\{T(v'', n+1) \mid v'' \in \gamma(a, v)\} = \{T(v'', n+1) \mid v'' \in \gamma(a, v')\}$. Then for any $u \in \gamma(a, v)$, (resp. $\in \gamma(a, v')$) there exists $u' \in \gamma(a, v')$ (resp. $\in \gamma(a, v)$) such that $T(u, n+1) = T(u', n+1)$. By induction hypothesis, $T(u, n) = T(u', n)$; hence $\{T(v'', n) \mid v'' \in \gamma(a, v)\} = \{T(v'', n) \mid v'' \in \gamma(a, v')\}$ and $T(v, n+1) = T(v', n+1)$. □

Then we define the following family of equivalence relations between the vertices of a bts.

$$v \overset{i}{\sim} v' \text{ iff } T(v, i) = T(v', i)$$

By the previous lemma we get

$$v \overset{i+1}{\sim} v' \Rightarrow v \overset{i}{\sim} v'$$

Let k_i be the number of classes of $\overset{i}{\sim}$. We have

- $1 \leq k_i \leq |V|$, where $|V|$ is the number of elements of V;
- $k_i \leq k_{i+1}$;
- if $k_i = k_{i+1}$ then $\overset{i}{\sim} = \overset{i+1}{\sim} = \overset{i+j}{\sim}$ for every $j \geq 1$.

This last point can be easily proved in the following way: since $\overset{i+1}{\sim}$ is included in $\overset{i}{\sim}$, if they have the same number of classes, they are equal. Now let us assume that $\overset{n}{\sim} = \overset{n+1}{\sim}$. $T(v, n+2)$ can be written as

$$\lambda(v)(\bigcup_{a\in A} \bigcup_{v'\in\gamma(a,v)} a \cdot T(v', n+1)),$$

and $T(u, n+2)$ as

$$\lambda(u)(\bigcup_{a\in A}\bigcup_{v'\in\gamma(a,u)} a\cdot T(v', n+1)).$$

If they are equal, it means that $\lambda(v) = \lambda(u)$ and that for every a, if $v' \in \gamma(a, v)$, there is some $u' \in \gamma(a, u)$ (and conversely) such that $T(v', n+1) = T(u', n+1)$, hence $T(v', n) = T(u', n)$ and $T(v, n+1) = T(u, n+1)$.

Thus we get the following lemma:

Lemma 6.2 $\overset{|V|}{\sim} = \overset{|V|+1}{\sim}$

Let us define $\sim$ to be $\overset{|V|}{\sim}$ and let us denote by $[v]$ the equivalence class of v for $\sim$, and for any subset V' of V, by $[V']$ the set $\{[v] \mid v \in V'\}$. We can define the quotient bts $S/\sim = < [V], \gamma', \lambda' >$ of S in the following way:

- $\lambda'([v]) = \lambda(v)$; this is independent on the choice of v in $[v]$ because of the very definition of $\sim$;
- $\gamma'(a, [v]) = [\gamma(a, v)]$; this is also independent on the choice of v because $v \sim v'$ implies $v \overset{|V|+1}{\sim} v'$, hence $T(v, |V|+1) = T(v', |V|+1)$; it follows that $\{T(v'', |V|) \mid v'' \in \gamma(a, v)\} = \{T(v'', |V|) \mid v'' \in \gamma(a, v')\}$, hence $[\gamma(a, v)] = [\gamma(a, v')]$.

It follows from this definition that $S/\sim$ is the image of S under a surjective homomorphism, from which we derive the lemma:

Lemma 6.3 *If S is minimal, then $S = S/\sim$, and this implies that $v \neq v' \Rightarrow T(v, |V|) \neq T(v', |V|)$.*

Now let us consider a minimal bts S. Let us define the finite set Q as beeing the set of all (tree-shaped) transition systems obtained by deleting some subtrees in some $T(v, |V|)$ and replacing them by a vertex labelled by $\perp$. This set is obviously ordered and contains $\{T(v, n) \mid v \in V, n \leq |V|\}$ as an ordered subset. Let us define the following bts automaton $\mathcal{A} = < A, L, Q \cup \{\sigma\}, \perp, \delta >$ where Q is defined as above, σ is a new state greater than any other state and where δ is defined by

$$\delta(l, X_{a_1}, \ldots, X_{a_n}) = \tau$$

where τ is defined as follows:
•first, let τ' be

$$l(\bigcup_{a\in A}\bigcup_{v\in X_a} a\cdot v),$$

•second, replace all subtrees of τ' at depth greater than $|V|$ by $\perp$, getting τ'',
•then if $\tau'' \in Q$ then $\tau = \tau''$ otherwise $\tau = \sigma$.

It is easy to see that δ is increasing.

Finally let us define the acceptance condition as consisting only of the set $F = \{T(v, |V|) \mid v \in V\}$ which is obviously included in Q.

Let now $\mu\delta_S$ the least fixed point of the transition function δ over S. Then it can be shown that $\mu\delta_S(v) = T(v, |V|)$: clearly all the states assigned to v during the computation of $\mu\delta_S$ are less than $T(v, |V|)$; let us assume that some $\mu\delta_S(v)$ is strictly less than $T(v, |V|)$; that means that in this state $\mu\delta_S(v)$, $\perp$ appears at a depth less than $|V|$ which implies by an inductive argument that $\mu\delta_S(v') = \perp$ for some v' which is impossible because every vertex is assigned at least a tree whose root is labelled by the label of this vertex.

Thus S is accepted by the pair $< \mathcal{A}, \{F\} >$.

Conversely, let us assume that some S' is also accepted by the pair $< \mathcal{A}, \{F\} >$. We have to show that S is a homomorphic image of S'.

Let $\alpha = \mu\delta'_S$. Let us define the mapping associating with a vertex v of S' a vertex u of S such that $\alpha(v) = T(u, |V|)$; the result does not depend on the choice of u since, S beeing minimal, by lemma 6.3, $T(u, |V|) = T(u', |V|)$ implies $u = u'$. Moreover it is surjective since for every vertex v of S, there exists a vertex v' of S' such that $\alpha(v') = T(v, |V|)$. It remains to prove that this mapping is a bts homomorphism, which is a straightforward consequence of the fact that α is a fixed point of the transition function δ and of the definition of δ.

7 Non deterministic bts automata

The bts automata previously defined are deterministic in the sense that the transition relation δ is indeed a function from $L \times \wp(Q)^A$ into Q. If an automaton is non deterministic, then δ has to be one-to-many, but in this case it is difficult to define a condition of monotonicity which guarantees the existence of a least fixed point.

A first approach to this problem is to consider that a non deterministic transition relation is not a one-to-many mapping but a set of functions: applying such a transition relation consists in choosing one of the function in the set and applying it. In this case, one has just to assume that each one of these functions is monotonic. Such a point of view about non deterministic functions (sets of deterministic functions rather than multivocal functions) has already been fruitfully applied to the semantics of non deterministic recursive program schemes [2]. For the classical cases (words, trees), a transition function can always be defined this way provided two sets $\mathcal{F}$ and $\mathcal{F}'$ of functions are considered equivalent if for every argument x the two sets $\{f(x) \mid f \in \mathcal{F}\}$ and $\{f(x) \mid f \in \mathcal{F}'\}$ are equal, which means that the two sets define the same multivocal function.

This is probably not enough to guarantee the exixtence of a least fixed point: each time a vertex is "visited" one has to apply one of the transition functions appliable to this vertex; since it is not necessarily the same one which is chosen at every visit, there is no reason for the value of the state assigned to this vertex only increases. This problem disappears if, once one of the function appliable to some vertex is chosen, at each further visit, this function will be chosen too. In this case we are sure that a least fixed point will be reached.

But then, it is equivalent to add to each vertex another label, which indicates which is the function to be applied at this vertex and, on such a transition system, the automaton becomes deterministic. Therefore one can say that a set of bts is *non deterministically recognizable* if it is the *projection* of a set recognized by a (deterministic) bts automaton, where the projection of a bts is defined as follows:

A bts $S =< V, \gamma, \lambda >$ over A and L is a *projection* of $S' =< V', \gamma', \lambda' >$ over A and L' if there exist a bijection β between V' and V and a mapping p from L' in L such that

- for every v in V, $\lambda'(\beta(v)) = p(\lambda(v))$;

- for every v in V and for every a in A, $\gamma'(a, \beta(v)) = \beta(\gamma(a, v))$.

In other words, S is obtained from S' by replacing the label of every vertex by its image under p.

An example Let us define the binary operator $\oplus$ which associates, with two bts, their disjoint union. Let us extend this operator to sets of bts by

$$K \oplus K' = \{S \oplus S' \mid S \in K, S' \in K'\}.$$

It is easy to see that if K and K' are both recognizable by (deterministic) automata, then $K \oplus K'$ is non deterministically recognizable:

- first of all, consider two isomorphic copies K_1 and K_2 of K and K', with disjoints sets of vertex labels.
- K_1 and K_2 are still recognizable by two deterministic automata and one can assume that they have disjoint sets of states.
- the "disjoint union" (the intuitive meaning of this notion is obvious) is still a deterministic bts automaton and one can easily define an acceptance criterion such that it recognizes $K_1 \oplus K_2$.

Then $K \oplus K'$ is the projection of $K_1 \oplus K_2$. Intuitively speaking the modifications of the vertex labels of K and K' simply allows to select which automaton has to run when visiting some vertex of K or K'.

On the other hand the disjoint union (in the sense defined above) of two deterministically recognizable sets is not necessarily deterministically recognizable. Indeed, it seems probable that there exist examples of deterministically recognizable sets, disjoint union of which is not. But it remains to find out such examples and to show that their disjoint union is not deterministically recognizable, which could be not so easy.

Also the family of nondeterministically recognizable sets need not to be a boolean algebra: it is obviously closed under union but probably not under complementation, nor even under intersection. Here again counter-examples and proofs have to be given and are presumably not immediate.

8 Recognizable sets and graph grammars

It is well known that in the cases of words and of trees, there exist some kinds of grammars, the regular grammars, which generate exactly the recognizable sets of words and of trees. The question is quite open for the case studied here: a lot of different kinds of graph grammars have already been defined in the litterature; do there exist some kinds of graph grammars which generate exactly the deterministically and/or the nondeterministically recognizable sets of bilabelled transition systems, and which, therefore, will be deserved to be named regular, as far as the notion of recognizability defined in this paper can be considered as a correct extension of the similar notion for words and trees.

References

[1] A. Arnold and A. Dicky. *An algebraic characterization of transition system equivalences.* Technical Report I-8603, Université de Bordeaux I, 1986.

[2] A. Arnold, P. Naudin, and M. Nivat. On semantics of non deterministic program schemes. In M. Nivat and J. Reynolds, editors, *Algebraic methods in semantics*, pages 1–33, Cambridge University Press, 1985.

[3] F. Bossut and B.Warin. *Rationalité et reconnaissabilité dans des graphes*. PhD thesis, Université de Lille, 1986.

[4] J. R. Büchi. On a decision method in restricted second order arithmetic. In E. Nagl, editor, *Logic, Methodology, and Philosophy of Science*, pages 1–11, Stanford Univ. Press, 1960.

[5] B. Courcelle. Some applications of logic of universal algebra, and of category theory to the theory of graph transformations. *Bull. EATCS*, 36:161–218, october 1988.

[6] D. Kozen. Results on the propositional μ-calculus. *Theoretical Comput. Sci.*, 27:333–354, 1983.

[7] G. D. Plotkin. A powerdomain construction. *SIAM J. Comput.*, 5:452–487, 1976.

[8] M. O. Rabin. Decidability of second-order theories and automata on infinite trees. *Trans. Amer. Math. Soc.*, 141:1–35, 1969.

(Received September 25, 1989)