

# Automata on Infinite Trees with Counting Constraints

Danièle Beauquier\*

LITP-IBP 4 Place Jussieu 75252 Paris Cedex 05 France

Damian Niwiński†

Institute of Informatics, University of Warsaw  
ul. Banacha 2, 02 097 Warszawa, Poland

## Abstract

We investigate finite automata on infinite trees with the usual Muller criterion for the success of an infinite computation path, but with the acceptance paradigm modified in that not all the computation paths need to be successful. Instead, it is required that the number of successful paths must belong to a specified set of cardinals  $\Gamma$ . We show that Muller automata with the acceptance constraint of the form “there are at least  $\gamma$  accepting paths” can be always simulated by tree automata with a weaker criterion for successful paths, namely Büchi acceptance condition. We also show that this is the most general class of constraints for which a simulation by Büchi automata is always possible. Next, we characterize the maximal class of constraints which can be simulated by classical Muller automata (known to be more powerful than Büchi automata). The condition required of the set  $\Gamma$  there, is that the intersection with natural numbers forms a recognizable set. Finally, we exhibit a set of trees which is recognized by a classical Büchi automaton but fails to be recognized by any Muller automaton with a non trivial cardinality constraint (i.e., except for  $\Gamma = 0$ ).

## 1 Introduction

The subject of automata on infinite objects (words or trees) has attracted the attention of computer scientists from both a practical and a theoretical point of view.

An essay by [E.A.EMERSON(1990)] gives an account of how the ideas of Büchi, originally conceived in context of mathematical logic, have later found

---

\*Supported partially by ESPRIT-BRA working group ASMICS

†Supported partially by Polish KBN grant n<sup>o</sup>211929101

multiple applications in the reasoning about concurrent programs with ideally nonterminating behaviour, this including such issues as specification, mechanical synthesis and verification of such programs.

From a theoretical viewpoint, the interesting feature is that finite automata provide an essentially finite (“finite state”) description of intrinsically infinite and often highly complicated objects, as e.g., strategies in infinite games. The subject is relevant to several topics, such as decidability of mathematical theories, expressiveness of logics, Borel and projective hierarchies of descriptive set theory, and determinacy of games (see [W. THOMAS(1990)] for a detailed survey).

The use of automata for reasoning about programs is based on the fact that, although the automata can accept highly undecidable sets of infinite objects due to infinite computations, yet still the emptiness problem for them is elementarily decidable ([M.O. RABIN(1969)], the actual complexity varies from polynomial to exponential time depending on the type of automata, c.f. [E.A. EMERSON AND C. JUTLA(1988)]). This fact has been used to show the elementary decidability of a variety of logics of programs. It is reported by [E.A. EMERSON(1990)] that automata *themselves* have also been proposed as a specification language for concurrent programs. In this context, the questions about different aspects of the expressive (or, defining) power of automata are of interest. In the present paper, we consider the question what kind of *counting* conditions can be expressed by automata.

The concept of “counting” is well known in computation theory. Generally speaking, it consists of a refinement of qualitative conditions like “there is a successful computation” or “each computation is successful” by giving a constraint on the *number* of successful computations as, e.g., “at least a half of possible computations is successful”.

Recall that a classical concept of the acceptance of an infinite tree by a finite automaton is based on the notion of a *successful path* of computation. An automaton starts its computation at the root of an input tree and then simultaneously works down the paths of the tree, level by level. In a Büchi automaton, a computation path is successful simply if some *accepting state* is assumed infinitely often. Rabin automata, as well as Muller automata, have a more sophisticated acceptance condition that can be viewed as a Boolean combination of Büchi conditions. For all these automata, the acceptance paradigm is the same: a whole computation is successful if *all* its paths are successful.

The idea of introducing counting constraints for automata on infinite trees was suggested by M. Nivat and has been first investigated by [D. BEAUQUIER, M. NIVAT AND D. NIWIŃSKI(1992)] for the case of Büchi automata. The acceptance criterion is now altered in that not all the computation paths need to be successful. Instead, it is required that the number of successful paths must belong to a specified set of cardinals  $\Gamma$ . It is shown in that paper that Büchi automata with a constraint “there are at least  $\gamma$  accepting paths” can be simulated by ordinary Büchi automata while any other type of a con-

straint leads to a class of tree languages which is incomparable with the class of Büchi recognizable sets of trees.

In the present paper we investigate the question how the counting constraints alter the power of Muller automata on infinite trees.<sup>1</sup>

We strengthen a result mentioned above by showing that any Muller automaton with a constraint “there are at least  $\gamma$  accepting paths” can be simulated by an ordinary Büchi automaton. This fact is of interest since the emptiness problem is considerably easier for Büchi automata (polynomial time vs. NP-hard, c.f. [E.A. EMERSON AND C. JUTLA(1988)]), while they are weaker than Muller automata in the expressive power. We complete this result to a full characterization, by showing that a Muller automaton with a constraint specified by a set of cardinals  $\Gamma$  can be simulated by a Büchi automaton if and *only if*  $\Gamma$  is of the form  $\{\alpha : \gamma \leq \alpha\}$ . Next we characterize those constraints  $\Gamma$  which can be simulated by (classical) Muller automata. The necessary and sufficient condition is that the intersection of  $\Gamma$  with the set of natural numbers (i.e., finite cardinals) should form a recognizable set. Finally, we address the question whether Muller automata with cardinality constraints are at least as powerful as classical Muller automata and answer this question negatively. Actually, we exhibit a set of trees that is recognized by a Büchi automaton but cannot be recognized by a Muller automaton with any counting constraint except for the trivial case  $\Gamma = 0$  (i.e., “there are no accepting paths”, which amounts to the acceptance by a dual automaton).

The remaining of the paper is organized as follows. In the introductory Section 2, we present some basic notions concerning trees and also prove two lemmas that are crucial for further results. In Section 3, we introduce automata on trees, first classically and then with the counting constraints. We also point out the relationship with monadic logic. Sections 4 and 5 are devoted to the characterizations of those counting constraints that can be simulated by Büchi and Muller automata respectively. Then, in Section 6, we exhibit a tree language that cannot be accepted by a Muller automaton with any nontrivial cardinality constraint. In the last section, we discuss some possible extensions of our results and try to place them in a more general context.

The *continuum hypothesis* is assumed throughout the paper, so the numbers in consideration are only the natural numbers,  $\omega$  and the continuum (the cardinals greater than continuum are not relevant here). This assumption can be shown inessential by an argument of the descriptive set theory, since the sets in consideration (i.e., the sets of accepting paths) are Borel and the continuum hypothesis is known to be generally valid for such sets.

---

<sup>1</sup>We have chosen Muller automata rather than Rabin automata, that are maybe more commonly used in temporal logic, only for technical reasons. Muller automata are known to be effectively equivalent to Rabin automata in the expressive power (although the latter are in general more succinct) and the actual construction preserves not only the accepted sets but also the number of successful computation paths. All our results (and proofs) could be easily adapted to the case of Rabin automata.

## 2 Trees

The set of natural numbers  $0, 1, \dots$  is denoted by  $\omega$  and identified with the first infinite cardinal number. We say that a set  $X$  is *countable* if its cardinality,  $|X|$ , is less than or equal to  $\omega$ . The cardinality of the powerset of  $\omega$ , is denoted  $c$  (continuum).

For a set  $X$ ,  $X^*$  is the set of finite words over  $X$ , including the empty word  $\lambda$ . The *length* of a word  $w$  is denoted by  $|w|$ , note that  $|\lambda| = 0$ . The (proper) *initial segment relation* is denoted  $\leq$  ( $<$ ). The same symbols are occasionally used for the standard inequality relation on natural numbers, but confusion should not arise. The following relation will be also useful:

$$u \triangleleft v \text{ iff not } v \leq u$$

( $\triangleleft$  is not an ordering). The concatenation of words  $u, w \in X^*$  is presented by  $uw$ , this notation is also extended to sets of words  $L, K \subseteq X^*$ ,

$$LK = \{uw : u \in L \text{ and } w \in K\}.$$

If  $u = vw$ , we occasionally write  $w = v^{-1}u$ .

A nonempty subset  $T$  of  $X^*$  closed under initial segments is called a *tree*. The elements of  $T$  are usually called *nodes*, the  $\leq$ -maximal nodes are *leaves* and  $\lambda$  is the *root* of  $T$ . If  $u \in T, x \in X$  and  $ux \in T$  then  $ux$  is an *immediate successor* of  $u$  in  $T$ . An infinite sequence  $P = (w_0, w_1, \dots)$  such that  $w_0 = \lambda$  and, for each  $m, w_{m+1}$  is an immediate successor of  $w_m$  is called a *path* in  $T$ . We recall the celebrated *König's Lemma* (c.f., e.g., [K. KURATOWSKI AND A. MOSTOWSKI(1976)]).

If  $T \subseteq X^*$  is an infinite tree and each  $w \in T$  has only a finite number of immediate successors in  $T$  then  $T$  has an infinite path.

If  $S$  is an arbitrary set and  $T$  is a tree then a mapping  $t : T \rightarrow S$  is called an *S-valued tree* or shortly an *S-tree*; in this context  $T$  is the *domain* of  $t$  denoted by  $T = \text{dom}(t)$ . We say "root of  $t$ ", "path in  $t$ " etc., referring to the corresponding objects in  $\text{dom}(t)$ .

If  $P = (w_0, w_1, \dots)$  is a path in  $t$ , let

$$\text{Inf}(t, P) = \{s \in S : t(w_m) = s \text{ for infinitely many } m\}$$

Observe that if  $S$  is finite then  $\text{Inf}(t, P)$  is always nonempty and there is some  $m_0$ , such that  $(\forall m > m_0) t(w_m) \in \text{Inf}(t, P)$ .

For an S-tree  $t : \text{dom}(t) \rightarrow S$  and a node  $v \in \text{dom}(t)$ , the *subtree* of  $t$  induced by  $v$  is the S-tree denoted by  $t.v$  and defined by

- $\text{dom}(t.v) = \{w : vw \in \text{dom}(t)\}$
- $t.v(w) = t(vw)$ , for  $w \in \text{dom}(t.v)$ .

Now suppose that  $A \subseteq \text{dom}(t)$  is an *antichain* with respect to  $\leq$  (i.e. any two elements of  $A$  are incomparable) and let  $f$  be a function which associates an  $S$ -tree  $f(w)$  with each  $w \in A$ . Then the *substitution*  $t[f]$  is the  $S$ -tree defined by

- $\text{dom}(t[f]) = \{w \in \text{dom}(t) : \forall w' \in A, w \triangleleft w'\} \cup \bigcup_{w \in A} w \text{dom}(f(w)),$
- $t[f](u) = \begin{cases} f(w)(v) & \text{if } u = wv, w \in A \\ t(u) & \text{otherwise} \end{cases}$

In the case when  $A$  is finite, say  $A = \{w_1, \dots, w_k\}$ , we shall often express  $f$  explicitly, writing for example  $t[w_1 : t_1, \dots, w_k : t_k]$ .

We also introduce the concepts of *limit* and *iteration*. Suppose  $t_0, t_1, \dots$  is a sequence of  $S$ -trees such that  $\text{dom}(t_0) \subseteq \text{dom}(t_1) \subseteq \dots$ , and, for each  $w \in \bigcup_{n < \omega} \text{dom}(t_n)$  there is some  $m(w)$  such that  $\forall m \geq m(w), t_m(w) = t_{m(w)}(w)$ . Then we define the limit of the sequence  $t_n$ ,  $t = \lim t_n$  by

- $\text{dom}(t) = \bigcup_{n < \omega} \text{dom}(t_n),$
- $t(w) = t_{m(w)}(w), \text{ for } w \in \text{dom}(t).$

Now let  $t$  be an  $S$ -tree and let  $v$  and  $w$  be nodes of  $t$  such that  $v < w$  and  $t(v) = t(w)$ . Then we define the *iteration of  $t$  along the interval  $[v, w]$* , in symbols  $t^{[v, w]}$  as the limit of substitutions of  $t.v$  into (subsequently created copies of) the node  $w$ . Formally, we define a sequence of trees  $t_n$  and the sequence of nodes of corresponding trees,  $w_n$ , as follows. Let  $w = vu$ . We set

- $t_0 = t,$
- $w_0 = w,$
- $t_{n+1} = t_n[w_n : t.v],$
- $w_{n+1} = w_n u.$

Finally,

$$t^{[v, w]} =_{df} \lim t_n$$

In this context we shall call the path containing the chain  $w_0, w_1, \dots$ , a *trace of iteration*.

Since now on, for notational convenience, we shall focus on full binary trees over  $\Sigma$ , i.e. the  $\Sigma$ -trees with  $\text{dom}(t) = \{1, 2\}^*$ . Thus, any node  $w \in \text{dom}(t)$  has exactly two successors  $w1$  and  $w2$ . The extension of our results to  $n$ -ary trees (with  $n \geq 2$ ) or ranked trees (where the number of successors of a node depends on the actual label) would present no difficulty.

We end this section by proving two useful lemmas about trees. The first one is a combinatorial observation that some “triangle-like” pattern must occur in a sufficiently large finite part of a tree. It is convenient to consider here finite rather than infinite trees. Let  $\{1, 2\}^{\leq n} = \{w \in \{1, 2\}^* : |w| \leq n\}$ .

**Lemma 2.1** *Let  $S$  be a finite alphabet and let  $K \geq |S|^2 + 1$ . Let  $t : \{1, 2\}^{\leq K} \rightarrow S$  be a (finite) tree. Then there exist nodes  $u, v, w \in \{1, 2\}^{\leq K}$ , such that  $u < v$ ,  $u < w$ ,  $w$  and  $v$  are incomparable,  $t(u) = t(v) = t(w)$  and moreover  $\{t(x) : u \leq x \leq v\} = \{t(x) : u \leq x \leq w\}$ .*

*Proof.* By induction on  $|S|$ . For  $|S| = 1$ , the claim is obvious. Suppose that the lemma holds for alphabets with cardinality less than  $|S|$  and let  $t : \{1, 2\}^{\leq K} \rightarrow S$  be a tree with  $K \geq |S|^2 + 1$ . Let  $t(\lambda) = q$ . Now consider two cases.

(1) For each  $w$  with  $|w| = |S| + 1$ , there exists  $v \geq w$  with  $t(v) = q$ . In this case, by counting argument we are already done.

(2) There is some  $w$  with  $|w| = |S| + 1$ , such that  $\forall v \geq w : t(v) \neq q$ . Then the subtree  $t.v$  is of the form  $t.v : \{1, 2\}^{\leq K - |S| - 1} \rightarrow S - \{q\}$  where  $K - |S| - 1 \geq |S|^2 + 1 - |S| - 1 \geq (|S| - 1)^2 + 1$  (for  $|S| > 1$ ). Then the claim follows from the induction hypothesis.  $\square$

The second lemma of this section will be a core of the proof of our results concerning the cardinality  $c$ . Some preparation is needed.

Let  $S$  be a finite set and let  $t : \{1, 2\}^* \rightarrow S$  be a tree. Observe that, by simple counting argument, there is a subset  $S' \subseteq S$  such that  $\text{Inf}(t, P) = S'$ , for a continuum of paths in  $t$ . We shall complete this observation, by showing that the set of nodes assuming the states from  $S'$  contains some tree-like structure.

We call a set of nodes  $D \subseteq \{1, 2\}^*$  dense if  $D \neq \emptyset$  and

$$(\forall x \in D \exists y, z \in D) x < y \ \& \ x < z \ \& \ y \text{ and } z \text{ are incomparable}$$

We state the following.

**Lemma 2.2** *Let  $S$  be a finite set and let  $t : \{1, 2\}^* \rightarrow S$  be a tree. Let  $S' \subseteq S$  be a set such that  $\text{Inf}(t, P) = S'$ , for a continuum of paths in  $t$  and let  $s \in S'$ . Then there exists a dense set  $D \subseteq \{1, 2\}^*$  such that  $\forall x \in D : t(x) = s$  and*

$$(\forall x, y \in D) x < y \Rightarrow \{t(w) : x \leq w \leq y\} = S'$$

*Proof.* Call a node  $w$  s.t.  $t(w) = s$  an  $s$ -node. Let  $\Pi$  be a set of paths  $P$  in  $t$  such that  $\text{Inf}(t, P) = S'$ . Call a node  $w_i$  of a path  $P = (w_0, w_1, \dots)$  in  $\Pi$  terminal for  $P$  if  $\forall m \geq i : t(w_m) \in S'$ . Call a node super terminal if it is terminal for a continuum of paths in  $\Pi$ . Clearly, each path in  $\Pi$  has a terminal  $s$ -node and, since there is only countably many nodes while  $|\Pi| = 2^\omega$ , there exists a super terminal  $s$ -node  $x$ .

We now show that, whenever  $x$  is a super terminal  $s$ -node, there also exist some super terminal  $s$ -nodes  $y, z > x$  such that  $y$  and  $z$  are incomparable and

$$\{t(w) : x \leq w \leq y\} = \{t(w) : x \leq w \leq z\} = S'$$

Let  $\Pi(x) \subseteq \Pi$  be the set of paths for which  $x$  is a terminal node. Let  $X$  be the set of  $s$ -nodes  $x' > x$  such that  $x'$  occurs in some path in  $\Pi(x)$  and  $\{t(w) : x \leq w \leq x'\} = S'$ . Notice that any path  $P$  in  $\Pi(x)$  contains some node

$x'$  in  $X$  (in fact, infinitely many), and such a node is terminal for  $P$ . So, if we write  $\Pi(x, x')$  for the set of those paths in  $\Pi(x)$  that contain a node  $x'$ , we have  $\Pi(x) = \bigcup_{x' \in X} \Pi(x, x')$ , and, by the cardinality argument there must be some super terminal  $s$ -node in  $X$ ,  $y$  say. By applying the same argument in turn to  $y$  and then repeating it *ad infinitum*, we obtain the existence of infinitely many super terminal nodes in  $X$ . In order to show that there exists a super terminal node  $z \in X$ , **incomparable** with  $y$ , suppose for the contrary that all super terminal nodes in  $X$  form a chain and so lie along a single infinite path. Then, except for this path, any path in  $\Pi(x)$  belongs to some  $\Pi(x, x')$ , where  $x'$  is *not* super terminal and therefore  $\Pi(x, x')$  is countable. Thus,  $\Pi(x)$  is countable (as a countable union of countable sets), a contradiction with the choice of  $x$ . Thus the existence of the desired  $y$  and  $z$  is proved and, by repeating the same argument for  $y$  and  $z$  etc., we eventually obtain a set  $D$  satisfying the condition of the lemma.  $\square$

### 3 Automata with Cardinality Constraints

We now fix a finite alphabet  $\Sigma$ . Let  $T_\Sigma$  be the collection of all full binary  $\Sigma$ -trees, i.e. the trees of the form  $t : \{1, 2\}^* \rightarrow \Sigma$ .

We recall the classical definition of an automaton on infinite trees with Muller acceptance condition.

A *Muller automaton* on  $\Sigma$ -trees (henceforth often called just automaton) is a tuple  $\mathcal{A} = (Q, q_0, Trans, Accept)$ , where  $Q$  is a finite set of *states*,  $q_0$  is an *initial state*,  $Trans \subseteq Q \times \Sigma \times Q \times Q$  is a set of *transitions* and  $Accept \subseteq 2^Q$  is a family of successful sets of states.

A  $q$ -run of the automaton  $\mathcal{A}$  on a tree  $t$  is a  $Q$ -tree  $r : dom(t) \rightarrow Q$  such that  $r(\lambda) = q$  and for each node  $w \in dom(t)$ ,  $\langle r(w), t(w), r(w1), r(w2) \rangle \in Trans$ . A  $q_0$ -run is called just a *run*. A *path*  $P = (w_0, w_1, \dots)$  of a run  $r$  is *accepting* if  $Inf(r, P) \in Accept$ . A *run* is accepting if *all* its paths are accepting. A *tree*  $t$  is *accepted* by an automaton  $\mathcal{A}$  if there exists an accepting run of  $\mathcal{A}$  on  $t$ . The set of trees accepted by  $\mathcal{A}$  is denoted  $L(\mathcal{A})$ .

The notion of a *Büchi automaton* on  $\Sigma$ -trees differs from that of Muller automaton only in the acceptance condition that is actually simpler and can be identified with a set of states  $F \subseteq Q$ . A path  $P$  of a run  $r$  is accepting if  $Inf(r, P) \cap F \neq \emptyset$ . Clearly, a Büchi automaton can be presented as a Muller automaton, but the converse is not true. Let  $\Sigma = \{a, b\}$  and let  $M$  be the set of trees  $t \in T_\Sigma$  such that, any path  $P$  in  $t$  has only a finite number of occurrences of  $b$  (i.e.  $Inf(t, P) = \{a\}$ ). It can be easily seen that  $M$  can be accepted by a Muller automaton, but [M.O. RABIN(1970)] showed that

**Proposition 3.1** *There is no Büchi automaton that accepts  $M$ .*

Now let  $\gamma$  be a cardinal number. We say that a run of an automaton (of either type) is  $\{\gamma\}$ -*accepting* if it has precisely  $\gamma$  accepting paths. More generally, for a

set of cardinals  $\Gamma$ , a run will be called  $\Gamma$ -*accepting* if it is  $\{\gamma\}$ -accepting for some  $\gamma \in \Gamma$ . The corresponding sets of  $\Gamma$ -*accepted* trees will be denoted by  $L_\Gamma(\mathcal{A})$ . (Notice that “ $\omega$ -accepting” and “ $\{\omega\}$ -accepting” mean two different things.) Finally, let  $\Gamma$ -Muller $_\Sigma$  denote the class of all subsets of  $T_\Sigma$  that can be presented as  $L = L_\Gamma(\mathcal{A})$ , for some Muller automaton  $\mathcal{A}$ . We also use the notation Muller $_\Sigma$  and Büchi $_\Sigma$  to denote the families of classically Muller recognizable (resp. Büchi recognizable) sets of trees. We often omit the subscript  $\Gamma$  if the alphabet is known from the context or if the choice of a particular (finite) alphabet is irrelevant for the result. Let us remark that a run has at most  $c$  accepting paths. So, from now, **we only consider cardinals less than or equal to  $c$** .

*Examples.* A run is  $\omega$ -accepting if the number of accepting paths is finite; it is  $\{\alpha : \alpha \geq \omega\}$ -accepting or, equivalently,  $\{\omega, c\}$ -accepting if the number of accepting paths is infinite. Notice that  $L_{\{\omega\}}(\mathcal{A})$  is, in general, different from  $L(\mathcal{A})$ . The tree language  $L_{\{0\}}(\mathcal{A})$  coincides with the set accepted by the automaton *dual* to  $\mathcal{A}$ , i.e. the one that differs from  $\mathcal{A}$  only in that its accepting condition is the complement of *Accept* (w.r.t.  $2^Q$ ).

We now recall the basic facts concerning the monadic second-order theory of trees.

Recall that a (non-labelled) full binary tree can be considered as a logical structure of the form

$$T = (\{1, 2\}^*, \lambda, s_1, s_2, \leq)$$

where  $s_1$  and  $s_2$  are the left and the right successor functions over  $\{1, 2\}^*$  respectively,  $\lambda$  is the empty word (that can be viewed as a root of the tree) and  $\leq$  is the prefix relation. The formulas of *monadic second order logic* over this structure use individual variables  $x_0, x_1, \dots$  (ranging over nodes of the tree) and the set-variables  $X_0, X_1, \dots$  (ranging over sets of nodes). The *atomic formulas* are  $t_1 = t_2$ ,  $t_1 \leq t_2$ , and  $X_i(t)$ , where the  $t$ 's are *terms* built from the individual variables, the constant  $\lambda$  and the functional symbols  $s_i$ ,  $i = 1, 2$ . The other formulas are built as usual, using propositional connectives and quantifiers  $\exists, \forall$  applied to both kinds of variables.

The formulas of *weak* monadic second order logic have the same syntax but differ in the semantics: the set-variables which are quantified are assumed to range over finite sets of nodes.

The (weak) monadic second order theory of the structure described above is usually abbreviated  $(W)S2S$ .

In the following definition, it is convenient to have a distinguished set of variables, say  $Z_0, Z_1, \dots$  that are supposed to range over finite sets of nodes only. (The introduction of these variables does not increase the expressive power of the theory  $S2S$  since the property “ $X$  is finite” is expressible there). A formula of  $S2S$  is called a  $\Sigma^1_1$  formula if it is of the form

$$\exists Y_1 \dots \exists Y_m \varphi(Y_1, \dots, Y_m, X_1, \dots, X_n)$$

where, in  $\varphi$ , all quantifiers (if any) are restricted to variables among  $Z_0, Z_1, \dots$



Now, if the alphabet  $\Sigma$  can be embedded into  $\{0, 1\}^k$ , for suitable  $k$ , a tree  $t \in T_\Sigma$  can be coded by a  $k$ -tuple of sets of nodes  $X_1, \dots, X_k$ , such that  $w \in X_i$  iff the  $i$ -th component of  $t(w)$  is 1. (Conversely, a  $k$ -tuple of sets of nodes represents a tree over the alphabet  $\{0, 1\}^k$ .)

In this sense, a set of trees  $L \subseteq T_\Sigma$  can be identified with a set of  $k$ -tuples of sets of nodes and therefore can be defined by a formula of  $S2S$ ,  $\phi(X_1, \dots, X_n)$ .

We recall the classical characterizations due to [M.O. RABIN(1970)].

**Theorem 3.2** *A set of trees  $L \subseteq T_\Sigma$  is recognizable by a (ordinary) Muller automaton iff it can be defined by an  $S2S$  formula.*

*A set of trees  $L \subseteq T_\Sigma$  is recognizable by a Büchi automaton iff it can be defined by a  $\Sigma_1^1$   $S2S$  formula.*

## 4 Büchi Definable Constraints

In this section we examine the cardinality constraints of the form “there are at least  $\gamma$  accepting paths”. We show that Muller automata with this kind of constraints can be always simulated by ordinary Büchi automata. We also show that this is the only kind of cardinality constraint having such a property.

**Theorem 4.1** *Let  $\gamma$  be a cardinal number. Then*

$$\{\alpha : \alpha \geq \gamma\} - \text{Muller} \subseteq \text{Büchi}$$

*Proof.* Let us abbreviate  $\{\alpha : \alpha \geq \gamma\} = [\gamma]$ .

Let  $\mathcal{A} = (Q, q_0, \text{Trans}, \text{Accept})$  be a Muller automaton. Observe first that any run is  $[0]$ -accepting and so it is easy to construct a suitable Büchi automaton (just set in  $\mathcal{A}$ ,  $F = Q$ ).

For  $\gamma > 0$ , it will be convenient to use the logical characterization of Büchi automata mentioned in the previous section. In doing this, we use the standard method of coding a run of an automaton with  $m$  states by an  $m$ -tuple of sets of nodes (see, e.g., [W. THOMAS(1990)]). If  $\gamma$  is a finite cardinal (i.e. an integer), the construction of a suitable  $\Sigma_1^1$  formula is easy. For  $\gamma = \omega$ , we use an observation that the existence of  $\omega$  paths satisfying a given condition is equivalent to the existence of an infinite “comb” structure, i.e. an infinite antichain of nodes each of them lying on some path satisfying the condition in question. Such a figure can be described by a  $\Sigma_1^1$  formula (a formal argument can be found in [D. BEAUQUIER, M. NIVAT AND D. NIWIŃSKI(1992)]). The case of  $\gamma = c$  is the most interesting one. We use here 2.2 from the introductory section. Then the existence of  $c$  paths can be reduced to the existence of some dense set of full nodes.  $\square$

One can ask whether the inclusions in the above theorem are proper. It is indeed the case and we shall see an actually stronger result in Section 5.

Another natural question is whether the class of constraints  $\Gamma$  considered above is a maximal one for the inclusion  $\Gamma - \text{Muller} \subseteq \text{Büchi}$ , or an apparently

weaker result  $\Gamma - \text{Büchi} \subseteq \text{Büchi}$ , holds. It turns out that it is indeed the case, i.e., the special form of  $\Gamma$  considered in the theorem above is necessary.

Let  $\Sigma = \{a, b\}$ . Consider a Büchi automaton defined as follows:  $\mathcal{A} = (\{q_a, q_b\}, q_a, \{(q, s, q_s, q_s) : q \in \{q_a, q_b\}, s \in \{a, b\}\}, \{q_b\})$ . It is easy to see that the accepted set  $L(\mathcal{A})$  consists of all trees in  $T_\Sigma$  such that any path  $P$  of  $t$  contains infinitely many  $b$ 's (i.e.  $b \in \text{Inf}(t, P)$ ). Moreover, a tree is  $\{\gamma\}$ -accepted by  $\mathcal{A}$  if it has exactly  $\gamma$  paths with infinitely many  $b$ 's.

**Lemma 4.2** *Let  $\Gamma$  be a set of cardinals. Then  $L_\Gamma(\mathcal{A}) \in \text{Büchi}$  iff  $\Gamma = \{\gamma\}$ , for some cardinal number  $\gamma$ .*

We need an auxiliary result for the proof of this lemma.

Let  $\mathcal{B} = (Q, q_0, \text{Trans}, \text{Accept})$  be a Büchi automaton. Suppose  $r$  is a  $q$ -run of  $\mathcal{B}$  on a tree  $t$ . A *demonic triangle* for  $r$  is a triple of nodes  $(u, v, w)$  such that  $v$  and  $w$  are incomparable and both greater than  $u$ ,  $r(u) = r(v) = r(w) \in \text{Accept}^2$  and there are some  $u < w_1 < v$  and  $u < w_2 < w$  such that  $t(w_1) = t(w_2) = b$ .

Now, for  $m \geq 1$ , we define a tree  $h^{(m)}$  similar to that used in [M.O. RABIN(1977)]. The tree  $h^{(m)}$  is  $b$ -labelled at nodes  $\lambda, 2^{n_1} 1, 2^{n_1} 12^{n_2} 1, \dots, 2^{n_1} 12^{n_2} 1 \dots 2^{n_m} 1$  for all  $n_1, \dots, n_m > 0$  and  $a$ -labelled at other nodes. The following lemma is proved in [D. BEAUQUIER, M. NIVAT AND D. NIWIŃSKI(1992)].

**Lemma 4.3** *If  $\mathcal{B}$  is a Büchi automaton with  $m$  states and  $r$  is an accepting  $q$ -run of  $\mathcal{B}$  on  $h^{(m)}$ , then there exists a demonic triangle for  $r$ .*

*Proof of Lemma 4.1*

We have to prove the "only if" part. Let us suppose that there are two cardinals  $\alpha$  and  $\beta$  such that  $\alpha \in \Gamma$ ,  $\beta \notin \Gamma$ ,  $\alpha < \beta \leq c$ . We claim that  $L_\Gamma(\mathcal{A}) \notin \text{Büchi}$ . Suppose that there is a Büchi automaton  $\mathcal{B}$  with  $m$  states such that  $L_\Gamma(\mathcal{A}) = L(\mathcal{B})$ . Fix some tree  $f$  in  $L_\Gamma(\mathcal{A})$  and let  $h^{(m)}$  be the tree defined above. We combine these two trees into a tree  $t$  such that  $t(\lambda) = a$  (for definiteness),  $t.1 = f$  and  $t.2 = h^{(m)}$ . Clearly,  $t \in L_\Gamma(\mathcal{A})$ . Let  $r$  be an accepting run of  $\mathcal{B}$  on  $t$ . Applying Lemma 4.2 to the subtree  $t.2$ , we obtain a demonic triangle of  $r$ , say  $(u, v, w)$ .

We have three cases to consider:  $\beta < \omega$ ,  $\beta = \omega$ ,  $\beta = c$ .

- $\beta < \omega$

Then, we can suppose without loss of generality that  $\beta = \alpha + 1$ . Let  $t' = t^{[u, w]}$  and  $r' = r^{[u, w]}$ . Since  $(u, v, w)$  is a demonic triangle,  $r'$  is an accepting run of  $\mathcal{B}$  on  $t'$ . On the other hand,  $t' \in L_{\alpha+1}(\mathcal{A})$  since we have created in  $t'$  a new path which has an infinite number of  $b$ . So,  $t' \notin L_\Gamma(\mathcal{A})$ , a contradiction with the hypothesis.

- $\beta = \omega$

---

<sup>2</sup>Recall, that for a Büchi automaton, *Accept* is a set of states.

In the tree  $t'$  defined above, let us consider, for  $i > 0$ , the nodes  $u'_i$  and  $v'_i$  defined in the following way:

$$u'_i = u(u^{-1}w)^i \text{ and } v'_i = u'_i(u^{-1}v).$$

Let  $t''$  be the limit of the sequence of trees  $t'^{[u'_1, v'_1]}$ ,  $t'^{[u'_1, v'_1][u'_2, v'_2]}$ ,  $\dots$ ,  $t'^{[u'_1, v'_1]\dots[u'_i, v'_i]}$ ,  $\dots$ . The tree  $t''$  is still accepted by  $\mathcal{B}$ , and  $t''$  has exactly  $\omega$  paths with an infinite number of  $b$ 's. Then,  $t'' \notin L_\Gamma(\mathcal{A})$ , a contradiction.

- $\beta = c$

We apply the following iteration construction to the triangle  $(u, v, w)$ . Let  $v = uv'$ ,  $w = uw'$ . Replace in  $t$  the two subtrees  $t.v$  and  $t.w$  by  $t.u$ , in the result replace the four subtrees induced by the nodes  $vv'$ ,  $vw'$ ,  $ww'$ ,  $ww'$  again by  $t.u$  and so on infinitely many times. The similar iteration is applied to the run  $r$ . Clearly the resulted tree is not in  $L_\Gamma(\mathcal{A})$  since it has a continuum set of paths with an infinite number of  $b$ 's, and nevertheless it is in  $L(\mathcal{B})$ , a contradiction.  $\square$

Clearly, the above result holds for any alphabet  $\Sigma' \supset \Sigma$ . Then, we can state the following.

**Corollary 4.4** *Let  $\Sigma$  have at least two elements and let  $\Gamma$  be a set of cardinals. Then  $\Gamma - \text{Muller} \subseteq \text{Büchi}$  iff  $\Gamma - \text{Büchi} \subseteq \text{Büchi}$  iff  $\Gamma = \{\alpha : \alpha \geq \gamma\}$ , for some ordinal  $\gamma$ .*

## 5 Muller Definable Constraints

In this section we consider the question: for what sets  $\Gamma$ ,  $\Gamma - \text{Muller} \subseteq \text{Muller}$ .

Call a set of integers  $A \subseteq \omega$  *recognizable* if it is recognizable by a finite automaton when identifying the integers with strings over one letter alphabet (c.f.[D.PERRIN(1990)]).

**Theorem 5.1** *Let the alphabet  $\Sigma$  have at least two elements. Then  $\Gamma - \text{Muller} \subseteq \text{Muller}$  iff  $\Gamma \cap \omega$  is recognizable.*

*Proof.* “only if”. Without loss of generality, we can set  $\Sigma = \{a, b\}$ . Suppose  $\Gamma - \text{Muller} \subseteq \text{Muller}$ . We shall construct an automaton  $\mathcal{A}$  such that the Muller recognizability of  $L_\Gamma(\mathcal{A})$  will imply the recognizability of  $\Gamma \cap \omega$ . Let  $\mathcal{A} = (Q, q_0, \text{Trans}, \text{Accept})$ , where

- $Q = \{q, \tilde{q}, q_a, \tilde{q}_a, q_b, \tilde{q}_b\}$
- $q_0 = q$
- $\text{Trans} = \{(q, a, q, q_a), (q_a, a, \tilde{q}_a, q_a), (\tilde{q}_a, a, \tilde{q}_a, \tilde{q}_a), (q, b, q_b, q_b), (q_b, b, q_b, q_b)\}$
- $\text{Accept} = \{\{q_a\}\}$

The automaton  $\mathcal{A}$  is deterministic and so, for any tree  $t \in T_\Sigma$ , there exists at most one run on  $t$ . Notice that the automaton  $\mathcal{A}$  admits a run on a tree  $t$  only in the case when either  $t(w) = a$  for all  $w$ , or, there exists an  $m_0 > 0$  such that, for all  $w \in \{1, 2\}^*$ ,  $t(1^{m_0}w) = b$  and  $t(v) = a$  otherwise. In this latter case,  $t \in L_\Gamma(\mathcal{A})$  iff  $m_0 \in \Gamma$ . Indeed, the only accepting paths are of the form  $(\lambda, 1, 11, \dots, 1^m, 1^m 2, 1^m 22, 1^m 222, \dots)$ , where  $m \geq 0$  provided that  $t(1^m) = a$  (and, consequently, all the nodes of the path are labelled by  $a$ ), so the number of accepting paths corresponds to the number of integers less than  $m_0$ . Let us denote a tree in the form considered above by  $t_{m_0}$ .

Now suppose that the set  $L_\Gamma(\mathcal{A})$  is recognized by a Muller automaton  $\mathcal{B}$ . We shall construct a finite automaton  $\mathcal{C}$  running on finite words over a single letter alphabet.

Let  $t_a$  be defined by  $(\forall w \in \{1, 2\}^*)t(w) = a$  (the similar for  $t_b$ ). Without loss of generality we can assume that the tree  $t_a$  is not in  $L_\Gamma$  (otherwise, we could take the tree language  $L_\Gamma(\mathcal{A}) - \{t_a\}$  that should be Muller recognizable as well). Let  $S$  be the set of states of  $\mathcal{B}$ . For  $s \in S$ , we shall use notation  $s \models t$  to mean "there is an accepting  $s$ -run (i.e. a run starting from  $s$ ) of  $\mathcal{B}$  on  $t$ ". We let the set of states of  $\mathcal{C}$  be  $S$  and the initial state be the same as in  $\mathcal{B}$ . A pair  $(s, s')$  is a transition of  $\mathcal{C}$  (we omit the single letter of the alphabet) if there is a transition  $(s, a, s', s'')$  of  $\mathcal{B}$  for some  $s''$  such that  $s'' \models t_a$ . Finally,  $s$  is an accepting state of  $\mathcal{C}$ , if  $s \models t_b$ . It is straightforward to see that there is an accepting finite run of  $\mathcal{C}$  on an integer  $k$  (considered as string over one letter alphabet) iff there is an accepting run of  $\mathcal{B}$  on a tree  $t_{k+1}$  which in turn is equivalent to  $k + 1 \in \Gamma$ . Then the set  $\{k : k + 1 \in \Gamma\}$  is recognizable and clearly, this induces that the set  $\Gamma$  itself is recognizable.

This completes the proof of the implication "only if" of the theorem.

We shall now prove the "if" implication.

We have  $L_\Gamma(\mathcal{A}) = \mathcal{L}_{-\cap\omega}(\mathcal{A}) \cup \mathcal{L}_{-\cap|\omega}(\mathcal{A})$ .

Since  $L_{|\omega}(\mathcal{A})$  and  $L_c(\mathcal{A})$  are in Büchi, Büchi  $\subset$  Muller and the family Muller is closed under boolean operations, we have only to prove that  $L_{\Gamma \cap \omega}(\mathcal{A}) \in$  Muller.

Recall that a recognizable set of integers can be always presented as a finite union of sets of the form  $\{p + na : n \in \omega\}$ , for some  $p, a$ .

Without loss of generality, we can assume that  $\Gamma$  itself is a set of integers in the above form. For notational simplicity, we let  $\Gamma = \{p + 2n \mid n \in \omega\}$ . Observe first that a run  $r$  of  $\mathcal{A}$  on a tree  $t$  is  $\{\gamma\}$ -accepting for some integer  $\gamma$  iff there is an antichain  $X$  consisting of  $\gamma$  elements such that:

- 1) for each successful path  $P$ ,  $X \cap P \neq \emptyset$
- 2) for each  $x \in X$ , there is a unique successful path  $P$  such that  $x \in P$ .

Now, in order to express the property that a run is  $\Gamma$  accepting, it is enough to show that a property " $X$  is a finite antichain the cardinality of which belongs to  $\Gamma$ " is expressible by an S2S formula. This fact is well known but we sketch the argument for the sake of completeness. Finiteness may be expressed by saying that there exists an antichain  $Y$  which majorizes  $X$  and each infinite path

intersects  $Y$ . Next, we have to say that  $X$  is a disjoint union of, say,  $X_1, X_2$  such that  $X_1$  has  $p$  elements and the cardinality of  $X_2$  is divisible by 2. The first is obvious (remember that  $p$  is fixed). Now observe that the lexicographical order  $\prec$  on  $X$  can be expressed by an S2S formula. Then,  $|X_2|$  is even iff either  $X_2$  is empty or

- i)  $X_2$  is a disjoint union  $X^{(0)} \cup X^{(1)}$
- ii) the minimal (w.r.t.  $\prec$ ) element of  $X_2$  exists and belongs to  $X^{(0)}$
- iii) the maximal element of  $X_2$  exists and belongs to  $X^{(1)}$
- iv) for each  $x \in X^{(0)}$  (resp.  $X^{(1)}$ ) not equal to the maximal element of  $X_2$ , its successor in  $X_2$  (w.r.t.  $\prec$ ) is in  $X^{(1)}$  (resp.  $X^{(0)}$ ).

Clearly the above clauses are expressible in the S2S. So the proof is done.  $\square$

## 6 Counterexample

We start with the following observation:

**Proposition 6.1**  $\{0\} - Muller = Muller$

*Proof.* For any of the inclusions, a suitable automaton can be obtained by complementing the family of accepting sets of states of the original automaton.  $\square$

We shall see that for any  $\Gamma \neq \{0\}$ , the situation is different. Let  $\Sigma = \{a, b\}$  and let  $L$  be the set of trees in  $T_\Sigma$  such that any path  $P$  of  $t$  contains infinitely many  $b$ 's (i.e.  $b \in Inf(t, P)$ ). We have already considered this language in Section 4, where we observed it is recognized by a Büchi automaton. On the other hand, we prove the following.

**Theorem 6.2** For any set of cardinals  $\Gamma \neq \{0\}$ ,  $L \notin \Gamma - Muller$ .

*Proof.*

Let us fix some  $m \geq 1$  and define a tree  $t$  by

$$t(w) = \begin{cases} b & \text{if } |w| = k(m+2), k < \omega \\ a & \text{otherwise} \end{cases}$$

We first state the following.

**Lemma 6.3** Let  $\mathcal{A}$  be a Muller automaton with  $m$  states. Suppose that there exists a run of  $\mathcal{A}$  on the tree  $t$  with  $\alpha$  accepting paths, where  $\alpha$  is a cardinal number greater than 0. Then there exists a tree  $t' \in T_\Sigma$  not in  $L$  and a run  $r'$  of  $\mathcal{A}$  on  $t'$  which has also precisely  $\alpha$  accepting paths.

Suppose that  $L = L_\Gamma(\mathcal{A})$ , for some Muller automaton  $\mathcal{A}$ , with  $m$  states.

Let  $r$  be a  $\gamma$ -accepting run of  $\mathcal{A}$  on the tree  $t$  (as defined above),  $\gamma \in \Gamma$ . If  $\gamma > 0$ , a contradiction follows immediately from the lemma above.

If  $\gamma = 0$ , choose  $\delta \in \Gamma$ ,  $\delta > 0$  (remember  $\Gamma \neq \{0\}$ ). We shall construct a tree  $t'$  not in  $L$  and a run  $r'$  of  $\mathcal{A}$  on  $t'$  that will have  $\delta$  accepting paths. The argument splits into three cases, depending on whether  $\delta$  is finite,  $\omega$ , or  $c$ .

Suppose first that  $\delta$  is finite. Choose  $k$  such that  $2^{k(m+2)} > \delta$  and select  $\delta$  different nodes of length  $k(m+1)$ , say  $v_1, \dots, v_\delta$ . Then, for each  $i = 1, \dots, \delta$ , choose  $w'_i, w''_i$  such that  $v_i < w'_i < w''_i$ ,  $(\forall u)w'_i \leq u \leq w''_i \Rightarrow t(u) = a$  and  $r(w'_i) = r(w''_i)$ . Observe that, for each  $i = 1, \dots, \delta$ , the set  $Q_i =_{df} \{r(u) : w'_i \leq u \leq w''_i\}$  is an accepting set of  $\mathcal{A}$ , (otherwise we could apply an iteration along  $[w'_i, w''_i]$  and obtain a run with 0 accepting path on a tree which is not in  $L$ .) Then apply the iteration to all pairs  $[w'_i, w''_i]$ . The resulted run  $r^{[w'_1, w''_1], \dots, [w'_\delta, w''_\delta]}$  has  $\delta$  successful paths, but the  $\{\delta\}$ -accepted tree  $t^{[w'_1, w''_1], \dots, [w'_\delta, w''_\delta]}$  is clearly not in  $L$ .

If  $\delta = \omega$ , the argument is essentially the same, but the corresponding selected nodes  $v_0, v_1, \dots$  cannot now, of course, be of the same length. Rather we can choose, for example,  $v_k = 2^{(k+1)(m+2)-1}1$ ,  $k = 0, 1, \dots$ . The rest of the argument is similar, we omit the details.

The case when  $\delta = c$  is slightly more difficult. (Notice that a consideration of this case is necessary, since  $\Gamma$  can be just  $\{0, c\}$ .) Let  $\tilde{t}$  be a tree defined as  $t$  above, but with replacing  $m$  by  $m^2 + 1$ , and  $\tilde{r}$  be a run of  $\mathcal{A}$  on  $\tilde{t}$  having 0 accepting paths. By applying Lemma 2.1 from the introductory section to the initial segment of this tree, we find that there exists a "triangle"  $u, v, w$ , such that  $|u|, |v|, |w| \leq m^2 + 1$ ,  $u < v, u < w$ ,  $v$  and  $w$  are incomparable,  $t(u) = t(v) = t(w)$  and  $\{t(x) : u \leq x \leq v\} = \{t(x) : u \leq x \leq w\}$ . Let us denote this last set by  $Q'$ . Now consider two cases.

If  $Q'$  is a nonaccepting set of  $\mathcal{A}$  then we can apply iteration to, e.g., the segment  $[u, v]$ . The resulted run  $\tilde{r}^{[u, v]}$  has still 0 accepting paths but  $\tilde{t}^{[u, v]} \notin L$ .

If  $Q'$  is accepting, we apply the iteration construction to the triangle. (More precisely, let  $v = vw', w = uw'$ . Replace in  $\tilde{t}$  the two subtrees  $\tilde{t}.v$  and  $\tilde{t}.w$  by  $\tilde{t}.u$ . Then, in the result, replace the four subtrees induced by the nodes  $vv', vw', ww', wv'$  again by  $\tilde{t}.u$  and so on. The construction can be easily formalized similarly as it has been done for the iteration along a single segment in the introductory section; we omit the details. The similar iteration is applied to the run  $\tilde{r}$ .) The resulted tree is not in  $L$  but the resulted run clearly has  $c$  accepting paths.

This completes the proof of the theorem.  $\square$

## 7 Conclusion

From a broader perspective, where, as we have mentioned in the introduction, automata on infinite trees can be viewed as a formalism for specification and verification of program properties, automata with counting constraints may prove to be useful for expressing those properties that explicitly refer to the cardinality. For example, we can express the fact that the number of computation

paths satisfying a fairness requirement is finite, or infinite, or continuum. From some point of view, the results of the paper may appear rather pessimistic: as pointed out by Theorems 4.1 and 6.1, the expressive power of these automata is rather limited. On the positive side, we believe that automata with counting constraints may be sometimes more convenient to use since they specify the property more directly and are in general more succinct than the corresponding Muller or Büchi automata. Yet still, in view of the characterizations provided by Corollary 4.1 and Theorem 5.1, for large classes of counting constraints, the new automata can be simulated by classical ones and, since the proofs of simulations are constructive, the standard techniques known for classical automata (as e.g., decision procedures for emptiness or containment) can apply.

## Acknowledgments

Most of this work was done while the second author visited LITP (Laboratoire d'Informatique Théorique et Programmation) in June and July 1992. Thanks go to the Laboratory members for this invitation and a stimulating working atmosphere. We also thank Jacques Sakarovitch for helpful ideas and A.A.Muchnik for enlightening remarks.

## References

- [D. BEAUQUIER, M. NIVAT AND D. NIWIŃSKI(1992)] The Effect of the Number of Successful Paths in a Büchi Tree Automaton, to appear in *Int. Jour. of Alg. and Comp.*
- [E.A.EMERSON(1990)] The Role of Büchi's Automata in Computing Science, *in: MacLane and Siefkes, eds., The Collected Works of J.R.Büchi, Springer Verlag, Berlin, 18-22.*
- [E.A. EMERSON AND C. JUTLA(1988)] The complexity of tree automata and logics of programs, *Proc. 29th IEEE Symp. on Foundations of Computer Science, N.Y.,328-337.*
- [F. GECSEG AND M. STEINBY(1984) ] Tree Automata, Akademiai Kiado, Budapest.
- [K. KURATOWSKI AND A.MOSTOWSKI(1976)] Set Theory, *North-Holland.*
- [D.PERRIN(1990) ] Finite Automata, *in: Handbook of Theoretical Computer Science, vol.B (J.van Leeuwen,ed.), 1-57.*
- [M.O. RABIN(1969) ] Decidability of second-order theories and automata on infinite trees, *Trans.Amer.Soc.*141, 1-35.

- [M.O. RABIN(1970) ] Weakly definable relations and special automata, *in: Mathematical Logic in Foundations of Set Theory* (Y.Bar-Hillel,ed.), 1-23.
- [M.O.RABIN(1972) ] Automata on infinite objects and Church 's problem, *Amer. Math.Soc.*, 1-22.
- [M.O. RABIN(1977) ] Decidable theories, *in: Handbook of Mathematical Logic* (J.Barwise, ed.).
- [W. THOMAS(1990)] Automata on infinite objects *in: Handbook of Theoretical Computer Science*, vol.B (J. van Leeuwen, ed.), 133-191.
- [M.Y. VARDI AND P.L.WOLPER(1986) ] Automata-theoretic techniques for modal logics of programs, *J. Comput. System Sci.* 32, 183-221.