LIPIcs Leibniz International Proceedings in Informatics

Recurrence and Transience for Probabilistic Automata

M. Tracol[1]*, C. Baier[2], M. Grösser[2]

[1] LRI
University Paris-South
France
tracol@lri.fr

[2] Technische Universität Dresden
Germany
{baier|groesser}@tcs.inf.tu-dresden.de

ABSTRACT.
In a context of ω-regular specifications for infinite execution sequences, the classical Büchi condition, or repeated liveness condition, asks that an accepting state is visited infinitely often. In this paper, we show that in a probabilistic context it is relevant to strengthen this infinitely often condition. An execution path is now accepting if the *proportion* of time spent on an accepting state does not go to zero as the length of the path goes to infinity. We introduce associated notions of recurrence and transience for non-homogeneous finite Markov chains and study the computational complexity of the associated problems. As Probabilistic Büchi Automata (PBA) have been an attempt to generalize Büchi automata to a probabilistic context, we define a class of Constrained Probabilistic Automata with our new accepting condition on runs. The accepted language is defined by the requirement that the measure of the set of accepting runs is positive (probable semantics) or equals 1 (almost-sure semantics). In contrast to the PBA case, we prove that the emptiness problem for the language of a constrained probabilistic Büchi automaton with the probable semantics is decidable.

1 Introduction

In a context of system analysis, ω-regular specifications are used to evaluate the long term properties of a system [14]. An ω-regular specification can be decomposed into a safety part and a liveness part. Typically, if the system is an elevator reacting to a user, an ω-regular specification can ensure that the system will never do something "wrong" (for instance having its door open while moving), and that the system will eventually do something "good" after a stimulus (for instance the elevator should stop on level i after a finite number of steps if the user asks to). The avoidance of the "wrong" event is the safety part, and the "eventually good" event is the liveness part. The liveness can be violated only in the limit. As underlines [6], a weakness of the classical definition is that the requirements can be satisfied by evolutions of the system which are quite unsatisfactory because no bound can be put on the response time. For instance, as the elevator is used by different users, they may have to wait an increasing and maybe unbounded amount of time to reach their level. Alternative

*This research was funded by the French program on Computer Security **ANR-07-SESU-013**

Foundations of Software Technology and Theoretical Computer Science (Kanpur) 2009.
Editors: Ravi Kannan and K. Narayan Kumar; pp 395–406
Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.
Digital Object Identifier: 10.4230/LIPIcs.FSTTCS.2009.2335

definitions for liveness have been proposed, in order to bound the distance between consecutive responses [6, 3]. In [6], the authors present the alternative notion of *finitary liveness*: finitary liveness assumes the existence of an unknown bound b such that every stimulus is followed by a response within b transitions. In this paper, instead of asking for a bound on the number of steps between "good" events, we will ask that the *proportion* of "good" events on a run does not go to zero as the length of the run goes to infinity.

In [4], the authors consider ω-regular properties on Markov chains, and in [2], the authors extend Büchi automata to a probabilistic context. They introduce the class $PBA^{>0}$ of Probabilistic Büchi Automata, which can be seen as a resolution of the non-determinism on a Büchi automaton by a probabilistic choice. In [2], as for the classical Büchi condition, a run is accepted if it visits infinitely often an accepting state, and a word is accepted if the probability of the set of associated accepted runs is non zero. This definition leads to a class of languages which is closed under the elementary operations of union, intersection and complementation. Moreover, the class of languages defined by $\text{PBA}^{>0}$ strictly subsumes the class of ω-regular languages. Unfortunately, working on these objects is difficult since basic problems such as the emptiness problem for the language of an automaton in $PBA^{>0}$ is undecidable [1].

In this paper, we consider alternative definitions of accepting runs. We introduce the notion of the *Support* of a run: a state s is in the support of a run r if the portion of time the state s is visited by r between time 0 and T, does not go to zero as T goes to infinity. We introduce the class $CPBA$ of *Constrained Probabilistic Büchi Automata*. A run on an automaton in $CPBA$ is accepting if there exists an accepting state in its support. As for $\text{PBA}^{>0}$, a word is accepted by a $\text{CPBA}^{>0}$ if the probability of the set of associated accepted runs is non zero.

We show that the class of languages associated to $\text{CPBA}^{>0}$ is not closed under complementation, however the emptiness problem is now in PSPACE. As it is done in [1] for the class PBA, we consider the class $CPBA^{=1}$ of Constrained Probabilistic Büchi Automata with an *almost sure semantics*. We prove that solving the emptiness problem of the language of an automaton in $CPBA^{=1}$ is equivalent to solving the same problem on an automaton in the class $PBA^{=1}$.

The fact that with positive probability an accepting state is in the support of a run can be seen as a recurrence property, by analogy with the classical homogeneous Markov chain theory. We define notions of recurrence and transience for non homogeneous probabilistic processes, in a context of Finite Probabilistic Tables (FPT, [16, 15]). An FPT can be seen as a non-homogeneous Markov chain on a finite state space with a finite number of transition functions. The main results of the paper are:

- Notions of weak and strong transience and recurrence for non homogeneous Markov processes.
- The study of the computational complexity of the associated problems, in particular the PSPACE-completeness of the strong recurrence problem, and the undecidability of the two states strong recurrence problem.
- The decidability of the emptiness problem for the languages of automata in $CPBA^{>0}$ and $CPBA^{=1}$.
- The study of the expressivity of our new classes of automata. In particular the set of the complement of languages of automata in $PBA^{=1}$ is expressible with automata in

$CPBA^{>0}$.

The paper is organized as follows: In section 2 we briefly recall the basic notions of finite probabilistic tables and define (constrained) probabilistic automata on infinite words. In section 3 we define the notions of transience and recurrence on finite probabilistic tables and study the computational complexity of the associated problems. In section 4 we consider the classes $CPBA^{>0}$ and $CPBA^{=1}$ and possible generalizations. Section 5 concludes the paper.

2 Preliminaries

Throughout the paper, we assume some familiarity with classical automata theory on infinite words [9]. We will use the notion of Finite Probabilistic Table (FPT) as a general framework for probabilistic automata. An FPT is the "structural part" of a probabilistic automaton, on which no acceptance condition has been made precise. An FPT can also be seen as a particular kind of non-homogeneous Markov chain, where only a finite number of transition functions are available. If S is a finite set, we write $\Delta(S)$ for the set of probability distributions on S.

DEFINITION 1._[Finite Probabilistic Tables [16]] A_ Finite Probabilistic Table _(FPT), is a tuple_ $\mathcal{T} = (S, \Sigma, \{M^a, a \in \Sigma\}, \alpha)$_, where_ S _is a finite set (representing the states),_ $\alpha \in \Delta(S)$ _is the initial distribution,_ Σ _is a finite set (representing the alphabet), and for all_ $a \in \Sigma$_,_ M^a _is a Markov matrix of order_ $|S|$ _(_M^a _represents the transition probabilities from state to state related to the symbol_ a_)._

We write $M^a = (m^a_{s_i,s_j})_{i,j\in\{1,\ldots,|S|\}}$. The component $m^a_{s,t}$ corresponds to the probability of going from state s to state t when the transition matrix M^a is chosen. If $w = a_1...a_l \in \Sigma^*$, we write M^w for the product $M^{a_1} \cdot ... \cdot M^{a_l}$, whose components are the $m^w_{s_i,s_j}$. Often, we will use the notation δ for the transition function: if $w \in \Sigma^*$ and $s, t \in S$, $\delta(s, w)(t)$ is the probability to arrive in t if we start on s and read w. In other words, $\delta(s, w)(t) = m^w_{s,t}$. We generalize the notation and write $\delta(s, w)$ for the set of states $t \in S$ such that $\delta(s, w)(t) > 0$. Finally, if $A \subseteq S$ (resp. $\alpha \in \Delta(S)$), $\delta(A, w)$ (resp. $\delta(\alpha, w)$) is the set of states $t \in S$ such that there exists $s \in A$ with $\delta(s, w)(t) > 0$ (resp. $s \in S$ s.t. $\alpha(s) > 0$ and $\delta(s, w)(t) > 0$). We will often define an FPT as a tuple $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, since we can compute easily δ and the $M^a, a \in \Sigma$ one from the other.

Let $\mathcal{T} = (S, \Sigma, \delta, \alpha)$ be an FPT. A *run* on $\mathcal{T}$, or a run on S and Σ, is an alternating sequence $s_0a_1s_1a_2...$, finite or infinite, of states in S and letters in Σ. The *trace* of a run r, written $Tr(r)$, is the sequence of its letters, and $Inf(r)$ is the set of states which appear infinitely often in r. Given a finite run $r = s_0a_1s_1...a_ns_n$ we denote by $|r| = n$ the length of r and by $r_{|k} = s_0a_1s_1...a_ks_k$ its prefix of length k. Similarly for a finite word $w \in \Sigma^*$, $|w|$ is the length of w and $w_{|k}$ denotes its prefix of length k. We write Ω for the set of infinite runs on $\mathcal{T}$. If $n \in \mathbb{N}$, X_n is the random variable on Ω which associates to a run r its n-th state. The set of cones of the form $C_w = \{r \in \Omega | Tr(r_{|n}) = w\}$, for $w \in \Sigma^n$, induces a σ-field $\mathcal{F}$ on Ω which is the smallest σ-field with respect to which all the $X_n, n \geq 0$, are measurable. The initial distribution α on S, and an infinite word $w = a_1a_2... \in \Sigma^\omega$, uniquely determine a probability measure $\mathbb{P}^\alpha_w$ on $\mathcal{F}$ such that $X_n, n \geq 0$ is a non-homogeneous Markov chain on $(\Omega, \mathcal{F}, \mathbb{P}^\alpha_w)$, with $\mathbb{P}^\alpha_w(X_0 = s) = \alpha(s)$, and $\mathbb{P}^\alpha_w(X_{n+1} = t | X_n = s) = \delta(s, a_{n+1})(t)$ for all

$n \in N$ and $s, t \in S$. (See [11, 13, 18, 7]). We may forget the α in the notation when clear from the context.

DEFINITION 2.*[Support of an infinite sequence] Let* Σ *be a finite alphabet, and* $w = a_0, a_1, ... \in \Sigma^\omega$. *Let* $\rho = b_0 b_1 ... b_l \in \Sigma^*$. *We call the* proportion *of* ρ *in* w *the limit-sup of the proportion of time spent reading* ρ *when reading* $a_1, ..., a_n$*:*

$$prop(\rho, w) = \overline{Lim}_{n\to\infty} \frac{|\{i \in [1; n-l] \ s.t. \ a_i = b_0 \wedge ... \wedge a_{i+l} = b_l\}|}{n}.$$

The support *of the sequence* w*, written* $Supp(w)$*, is the set of words* $\rho \in \Sigma^*$ *such that* $prop(\rho, w) > 0$.

For instance, if we consider a run r on an automaton $\mathcal{A}$ as an infinite sequence on $S \cup \Sigma$, the set of states in the support of r can be seen as the set of states on which r spends a non negligible amount of time. It is a subset of $Inf(r)$, and the inclusion is strict in general. Instead of imposing acceptance conditions on the set of states that are visited infinitely often in a run, in this paper we will impose acceptance conditions on the set of states that are visited with a "non negligible" portion, i.e. that are in the support of the run. This gives rise to the class of constrained probabilistic automata.

A probabilistic automaton is just a pair $\mathcal{A} = (\mathcal{T}, \mathsf{Acc})$ where $\mathcal{T} = (S, \Sigma, \delta, \alpha)$ is an FPT and Acc is an acceptance condition. We consider here acceptance conditions of the following types: Büchi, where $\mathsf{Acc} \subseteq S$ is a subset of final states, Street and Rabin, where $\mathsf{Acc} = \{(H_1, K_1), \ldots, (H_n, K_n)\}$ is a set of acceptance pairs and Muller, where $\mathsf{Acc} \subseteq 2^S$ is a set of final sets. Given a subset $T \subseteq S$ of states we call T accepting according to a

- Büchi acceptance condition $\mathsf{Acc} \subseteq S$, if $T \cap \mathsf{Acc} \neq \emptyset$. In the sequel we will denote a Büchi acceptance condition by F.
- Rabin acceptance condition $\mathsf{Acc} = \{(H_1, K_1), \ldots, (H_n, K_n)\}$, if there exists $1 \leq i \leq n$ such that $T \cap H_i = \emptyset$ and $T \cap K_i \neq \emptyset$.
- Streett acceptance condition $\mathsf{Acc} = \{(H_1, K_1), \ldots, (H_n, K_n)\}$, if for every $1 \leq i \leq n$ it holds that $T \cap H_i \neq \emptyset$ or $T \cap K_i = \emptyset$.
- Muller acceptance condition $\mathsf{Acc} \subseteq 2^S$, if $T \in \mathsf{Acc}$.

As indicated above we will distinguish between two types of automata, namely

- (classical) probabilistic automata, where a run is called accepting for $w \in \Sigma^\omega$ iff $Tr(r) = w$ and $Inf(r)$ is accepting and
- constrained probabilistic automata, where a run is called accepting for $w \in \Sigma^\omega$ iff $Tr(r) = w$ and $Supp(r)$ is accepting.

For both types of automata we distinguish two semantics, the probable semantics, where the accepted language of $\mathcal{A}$ is:

$$\mathcal{L}^{>0}(\mathcal{A}) = \{w \in \Sigma^\omega | \mathbb{P}_w(\{r | r \text{ is accepting for } w\}) > 0\}$$

and the almost-sure semantics where the accepted language of $\mathcal{A}$ is:

$$\mathcal{L}^{=1}(\mathcal{A}) = \{w \in \Sigma^\omega | \mathbb{P}_w(\{r | r \text{ is accepting for } w\}) = 1\}.$$

Given an automaton with a Büchi acceptance condition, we call a (classical) probabilistic automaton a PBA and we call a constrained probabilistic automaton a CPBA (the analogous notations apply to Street (PSA, CPSA), Rabin (PRA, CPRA) and Muller (PMA, CPMA) automata). The class of PBA is denoted PBA. In the following, when K is a class of automata and $x \in \{>0, =1\}$, we write $\mathcal{C}_{K^x}$ for the associated class of languages. We will sometimes write PBA^x, resp. PBA^x, to denote a PBA, resp. the class of PBA, with the associated semantics given by x.

By [2, 1], $\mathcal{C}_{PBA^{>0}}$ is closed under union, intersection and complementation. However, the emptiness problem of a PBA$^{>0}$ is shown to be undecidable. On the other hand, the emptiness problem for an automaton in $PBA^{=1}$ is shown to be decidable, but the class $\mathcal{C}_{PBA^{=1}}$ is not any more closed under complementation.

Remarks: Taking an *inf* limit instead of a *sup* limit in the definition of the support, we could express the fact that the proportion of time spent on a particular set of states stays bounded away from zero as the length increases. The two different possible definitions for the support of a run would lead to different classes of automata, which recognize different languages and can express different properties of interest. However, we will see that the same algorithms can be used on both classes of automata, for the natural problems such as the emptiness problem. In this paper, we will use the *limit-sup* to define the support of a run, but the results could be easily adapted to handle the *inf limit* case.

3 Finite non-homogeneous Markov chains

We are interested in basic questions concerning our models of probabilistic automata (PBA, $CPBA$), such as the emptiness problem of the language of a given automaton, or the universality problem for this language. Such problems can be presented in the general framework of finite non-homogeneous Markov chains. In the past, researcher working in this domain seem to have been mostly interested in considerations on the ergodic properties of such chains ([15, 17]). In general they did not take into account the fact that the number of transition functions of the process may be finite, which is crucial when dealing with probabilistic automata. We start with some remarks on homogeneous Markov chains, and next we study severall problems of interest concerning non-homogenous Markov chains.

3.1 Recurrence and transience for non-homogeneous Markov chains

Homogeneous Markov chains

We fix $X_i, i \geq 0$ a homogeneous Markov chain on a finite state space S. If $\alpha \in \Delta(S)$, $\mathbb{P}^\alpha$ is the probability distribution on the set of runs on the chain with initial distribution α. Recall, [11], that a state $s \in S$ is called *recurrent* if $\mathbb{P}^s(\{r|s \in Inf(r)\}) > 0$. Otherwise it is called *transient*. Note that we sometimes identify s with the Dirac distribution $\mu_s \in \Delta(S)$ with $\mu_s(s) = 1$.

THEOREM 3. *[Recurrence and the ergodic theorem, [11]] Given a homogeneous Markov chain with finite state space S and $s \in S$, s is recurrent iff $\mathbb{P}^s(\{r|s \in Inf(r)\}) = 1$, iff $\mathbb{P}^s(\{r|s \in Supp(r)\}) > 0$, iff $\mathbb{P}^s(\{r|s \in Supp(r)\}) = 1$.*

Thus, in the homogeneous case, a state s is recurrent if almost all the runs on the chain visit infinitely often s, or equivalently if almost all the runs spend a non negligible amount of time on s. We will see in the next subsection that this equivalence does not hold in the context of non-homogeneous Markov chains. Notice that the notion of finitary liveness of [6] is not adapted to the probabilistic context. Indeed, even if s is recurrent, on a homogeneous Markov chain, for almost all the runs on the chain, the distance between two consecutive occurrences of s is not bounded.

Non-homogeneous Markov chains

For the following we fix an FPT $\mathcal{T} = (S, \Sigma, \delta, \alpha)$.

Accessibility: a state $s \in S$ is said to be *accessible* in $\mathcal{T}$ if there exists $n \in \mathbb{N}$ and a word $\rho \in \Sigma^n$ such that $\delta(\alpha, \rho)(s) > 0$. That is, with positive probability the process can be in state s after a finite number steps. By simple reachability considerations, we can compute the set $Acc(\mathcal{T})$ of the accessible states in $\mathcal{T}$ in time polynomial in the size of the FPT.

Given a homogeneous Markov chain on S and $s \in S$, theorem 3 shows that $\mathbb{P}^s(\{r|s \in Inf(r)\}) > 0$ iff $\mathbb{P}^s(\{r|s \in Supp(r)\}) > 0$. This is not the case in the context of non-homogeneous Markov chains, which motivates the two following definitions for recurrence.

DEFINITION 4.*[Strong Recurrence, Weak Recurrence] Let $X_n, n \in \mathbb{N}$ be a non homogeneous Markov chain on a finite state space S, and $s \in S$. Let $\mathbb{P}$ be the probability distribution on the set of runs of the chain. We say that s is* weakly recurrent *(resp.* strongly recurrent*), if*

$$\mathbb{P}(\{r|s \in Inf(r)\}) > 0 \text{ (resp. } \mathbb{P}(\{r|s \in Supp(r)\}) > 0)$$

Otherwise, s is said to be weakly transient (resp. strongly transient).

Given an FPT $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, several algorithmic problems may arise, concerning transience and recurrence. The first question is whether we can find $w \in \Sigma^\omega$ such that a given state $s \in S$ is weakly, or strongly, recurrent, for the associated non-homogeneous Markov chain on $\mathcal{T}$.

Problem 1 (Weak recurrence (resp. strong recurrence))
Input: *An FPT $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, $F \subseteq S$.*
Question: *Is there $w \in \Sigma^\omega$ such that*

$$\mathbb{P}^\alpha_w[\{r|F \cap Inf(r) \neq \emptyset\}] > 0. \text{ (resp. } \mathbb{P}^\alpha_w[\{r|F \cap Supp(r) \neq \emptyset\}] > 0).$$

The undecidability of the emptiness problem for PBA$^{>0}$ [2], implies that the weak recurrence problem is undecidable. In contrast, we will see that the strong recurrence problem is PSPACE-complete (theorem 10). We cannot generalize our approach to several states, as we will prove that the following problem is undecidable (theorem 15):

Problem 2 (Two states strong recurrence)
Input: *An FPT $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, $s, t \in S$.*
Question: *Is there $w \in \Sigma^\omega$ s.t. $\mathbb{P}^\alpha_w[\{r|s \in Supp(r) \text{ and } t \in Supp(r)\}] > 0$?*

Consider now the *universal* analog of the weak recurrence problem (resp. of the strong recurrence problem): do we have that for all $w \in \Sigma^\omega$, $\mathbb{P}_w[\{r|s \in Inf(r)\}] > 0$? (resp. $\mathbb{P}_w[\{r|s \in Supp(r)\}] > 0$). By contraposition, these problems can be reformulated as follows.

Problem 3 (Universal weak recurrence (resp. universal strong recurrence))
Input: *An FPT $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, $F \subseteq S$.*
Question: *Is there $w \in \Sigma^\omega$ such that*

$$\mathbb{P}^\alpha_w[\{r|F \cap Inf(r) = \emptyset\}] = 1. \text{ (resp. } \mathbb{P}^\alpha_w[\{r|F \cap Supp(r) = \emptyset\}] = 1).$$

By the results of [1], since $\mathcal{C}_{PBA^{>0}}$ is closed under complementation, the universal weak recurrence problem is undecidable. We will show later that $\mathcal{C}_{CPBA^{>0}}$ is not closed under complementation, hence we cannot conclude directly for the complexity of the universal strong recurrence problem.

The condition $\mathbb{P}_w[\{r|F \cap Inf(r) \neq \emptyset\}] > 0$ (as well as the condition $\mathbb{P}_w[\{r|F \cap Supp(r) \neq \emptyset\}] > 0$), can be seen as a Büchi condition. One can be interested in the co-Büchi condition: a run is accepted if no state in F is visited infinitely often. The associated problems in our context are the following.

Problem 4 (Weak transience (resp. strong transience))
Input: *An FPT $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, $F \subseteq S$.*
Question: *Is there $w \in \Sigma^\omega$ such that*

$$\mathbb{P}^\alpha_w[\{r|F \cap Inf(r) = \emptyset\}] > 0. \text{ (resp. } \mathbb{P}^\alpha_w[\{r|F \cap Supp(r) = \emptyset\}] > 0.)$$

The weak transience and strong transience problems are both PSPACE -complete (theorem 14). As before, we can consider the universal versions of these problems.

Problem 5 (Universal weak transience (resp. universal strong transience))
Input: *An FPT $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, $F \subseteq S$.*
Question: *Is there $w \in \Sigma^\omega$ such that*

$$\mathbb{P}^\alpha_w[\{r|F \cap Inf(r) \neq \emptyset\}] = 1. \text{ (resp. } \mathbb{P}^\alpha_w[\{r|F \cap Supp(r) \neq \emptyset\}] = 1.)$$

The universal weak and strong transience problems are PSPACE-complete (theorem 12). The following of the section is devoted to the proofs of the complexity of the previous problems.

3.2 Computational complexity of the recurrence problems.

Our decision procedures will often rely on the notion of *probabilistic loop*, which correspond to the set of homogeneous Markov chains that one can define on an FPT.

DEFINITION 5.*[Probabilistic loop] A* probabilistic loop *in $\mathcal{T}$ is a couple (C, ρ), where $C \subseteq S$ and $\rho \in \Sigma^*$ are such that $\delta(C, \rho) \subseteq C$.*

If $F \subseteq S$, a probabilistic loop around F *in $\mathcal{T}$ is a probabilistic loop (C, ρ) in $\mathcal{T}$ such that for all $s \in C$, there exists ρ'_s a prefix of ρ, such that $\delta(s, \rho'_s) \cap F \neq \emptyset$.*

A probabilistic loop (C, ρ) in $\mathcal{T}$ induces an homogeneous Markov chain $X_n, n \in \mathbb{N}$ with state space C and transitions probabilities given, for all $s, t \in C$, by $\mathbb{P}[X_{n+1} = t | X_n = s] = \delta(s, \rho)(t)$. Let A be the set of states in C which are recurrent for this chain. The *Support* of the loop (C, ρ) is the set of states t in S such that there exists $s \in A$ and ρ' a prefix of ρ with $\delta(s, \rho')(t) > 0$.

We consider first the strong recurrence problem. We fix an instance $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, $F \subseteq S$, of the strong recurrence problem. We can assume that $F = \{s\}$, with no loss on generality. We will prove in this subsection that s is strongly recurrent for a non-homogeneous Markov chain on the probabilistic table iff s is accessible and there exists a probabilistic loop around s in $\mathcal{T}$. This will imply the PSPACE completeness of the strong recurrence problem. The next example shows that this equivalence does not hold in general, if Σ is infinite.

Example 1 Let $S = \{s, t\}$. For $\delta \in]0; 1]$ consider the Markov matrix $M_\delta = \begin{pmatrix} 1-\delta & \delta \\ 0 & 1 \end{pmatrix}$. The graph of the associated Markov chain is: $\begin{tikzcd} s \arrow[loop above, "1-\delta"] \arrow[r, "\delta"] & t \arrow[loop above, "1"] \end{tikzcd}$

Suppose that the chain is initiated on state s: $\alpha = \{s\}$. Consider now the family of matrices $\mathcal{M} = \{M_{1/2^i}, i \in \mathbb{N}\}$. It is not difficult to see that for any finite product of matrices in $\mathcal{M}$, the associated homogeneous Markov chain $X_n, n \geq 0$ on S is aperiodic and t is the only state in the support of the stationary distribution. By theorem 3, this implies that s is transient for the (homogeneous) chain. This implies that there exists no probabilistic loop around s in $\mathcal{T}$. However, if we consider the non-homogeneous Markov chain $X_n, n \geq 0$ on S whose transitions probabilities are given by the matrices $M_{1/2}, M_{1/2^2}, M_{1/2^3}, ...$, then $\mathbb{P}^{\alpha}_{1/2,1/2^2...}[\{r | \forall n \in \mathbb{N}\ X_n(r) = s\}] > 0$, and in particular $\mathbb{P}^{\alpha}_{1/2,1/2^2...}[\{r | s \in Supp(r) > 0\}] > 0$, which proves that s is strongly recurrent for the (non-homogeneous) chain.

We give a couple of definitions and lemma to prove our theorem. The notion of *filter* will allow us to build a probabilistic loop around a state s by aggregating the successors of this state.

DEFINITION 6. *[Filters] Let S be a finite state space, and Σ be a finite alphabet. A* filter *on S and Σ is a finite sequence of couples on $S \cup \{\cdot\}$ and $\Sigma \cup \{\cdot\}$, where $\cdot$ is a special symbol denoting an "indefinite place".*

A filter can be seen as a word in $((S \cup \{\cdot\})(\Sigma \cup \{\cdot\}))^*$. Two filters x and y will be said to coincide, written $x = y$, if they have the same length and at each place either they have the same elements, or at least one has got an empty place. If u and v are two filters on S and Σ, then uv is the natural concatenated filter: For instance, if $w = a_1 ... a_l \in \Sigma^*$ and $s \in S$, then (s, w, s) is the filter $(s, a_1), (\cdot, a_2), ..., (\cdot, a_l), (s, \cdot)$.

We start with a combinatorial lemma. The proportion $prop(w, r)$ of a filter w in a run r is naturally defined the same way as we defined the proportion of a subword in a run, using a limit-sup.

LEMMA 7. *Let S be a finite state space and Σ be a finite alphabet. Let r be a run on S and Σ, and let u be a filter on S and Σ. Suppose $prop(u, r) > 1/N$, where $N \in \mathbb{N}$ and $N > |u|$. Then there exists another filter v on S and Σ such that $prop(uvu, r) > 1/(2 \cdot N)$. Moreover, we can choose v such that $|v| \leq 2 \cdot N$.*

We will apply recursively the following lemma to build a probabilistic loop around s.

LEMMA 8. *Let $\rho \in \Sigma^*$. Suppose $\mathbb{P}^{\alpha}_{w}[\{r | prop((s, \rho), r) > 0\}] > 0$, and let $t \in \delta(s, \rho)$. Then, there exists $\rho' \in \Sigma^*$ such that:*

$$s \in \delta(t, \rho'), \text{ and } \mathbb{P}^{\alpha}_{w}[\{r | prop((s, \rho\rho'), r) > 0\}] > 0.$$

THEOREM 9. *Let $\mathcal{T} = (S, \Sigma, \delta, \alpha)$, $s \in S$, be an instance of the strong recurrence problem. Then the following are equivalent:*

- *There exists $w \in \Sigma^\omega$ such that s is strongly recurrent for the associated non-homogeneous Markov chain on $\mathcal{T}$.*
- *s is accessible, and there exists a probabilistic loop around s in $\mathcal{T}$.*

Moreover, in the positive case, the letters of the trace of the loop can all be taken in the support of w.

PROOF. (sketch) Notice that one way is easy: if there exists $\rho_0 \in \Sigma^n$ such that $\delta(\alpha, \rho_0)(s) > 0$ and if there exists a probabilistic loop (C, ρ) around s, then $\mathbb{P}^{\alpha}_{\rho_0 \cdot \rho^\omega}(\{r | s \in Supp(r)\}) > 0$, and s is strongly recurrent for the chain associated to $w = \rho_0 \cdot \rho^\omega$.

We prove now that the strong recurrence problem is PSPACE complete. First, we know that we can compute in PTIME if s is accessible from α. Thus, the strong recurrence problem is PTIME equivalent to the problem of finding if there exists a probabilistic loop around s. We reduce the problem of Finite Intersection of Regular Languages, which is known to be PSPACE complete [12], to our strong recurrence problem.

Problem 6 (Finite Intersection of Regular Languages)
Input: $\mathcal{A}_1, ..., \mathcal{A}_l$ *a family of deterministic automata (on finite words) on the same finite alphabet* Σ.
Question: Do we have $\mathcal{L}(\mathcal{A}_1) \cap ... \cap \mathcal{L}(\mathcal{A}_l) = \emptyset$ *?*

THEOREM 10. *The strong recurrence problem is PSPACE complete.*

We consider now the complexity of the co-Büchi problems.

PROPOSITION 11. *Let* $\mathcal{T} = (S, \Sigma, \delta, \alpha)$ *be an FPT, and* $F \subseteq S$. *Then the following are equivalent:*

1. $\exists w \in \Sigma^\omega$ *s.t.* $\mathbb{P}^{\alpha}_{w}[\{r | F \cap Inf(r) = \emptyset\}] > 0$.
2. $\exists w \in \Sigma^\omega$ *s.t.* $\mathbb{P}^{\alpha}_{w}[\{r | F \cap Supp(r) = \emptyset\}] > 0$.
3. *There exists an accessible probabilistic loop on* S *whose support does not contain any state in* F

THEOREM 12. *The universal weak and strong transience problems (problem 5) are PSPACE complete.*

PROOF. As for the strong recurrence problem, we can build a nondeterministic Turing machine which finds a relevant probabilistic loop in PSPACE. For the PSPACE hardness, we can also reduce the finite intersection of regular languages problem to these problems.

PROPOSITION 13. *Let* $\mathcal{T} = (S, \Sigma, \delta, \alpha)$ *be an FPT, and* $F \subseteq S$. *Then the following are equivalent:*

1. $\exists w \in \Sigma^\omega$ *s.t.* $\mathbb{P}_{w}[\{r | F \cap Inf(r) \neq \emptyset\}] = 1$.
2. $\exists w \in \Sigma^\omega$ *s.t.* $\mathbb{P}_{w}[\{r | F \cap Supp(r) \neq \emptyset\}] = 1$.
3. *There exists* ρ_0 *and* ρ *in* Σ^* *such that* $(\delta(\alpha, \rho_0), \rho)$ *is a probabilistic loop around* F.

PROOF. $3 \Rightarrow 2$ and $2 \Rightarrow 1$ are simple. Suppose 1: $\exists w \in \Sigma^\omega$ s.t. $\mathbb{P}^{\alpha}_{w}[\{r | F \cap Inf(r) \neq \emptyset\}] > 0$. Write $w = a_1 a_2 ....$ For $i \in \mathbb{N}$, let $H_i = \delta(\alpha, w_{|i}) = \bigcup_{s | \alpha(s) > 0} \delta(s, w_{|i})$.

Since S is finite, there exists $H \subseteq S$ such that infinitely often, $H_i = H$. Let $i_0 \in \mathbb{N}$ such that $H_{i_0} = H$. Let $t \in H$. Then $\mathbb{P}^{\alpha}_{w}[\{r | X_{i_0}(r) = t\}] > 0$. Since $\mathbb{P}^{\alpha}_{w}[\{r | F \cap Inf(r) \neq \emptyset\}] = 1$, F must be reachable from t after a finite number of steps. That is, there exists $l_t \in \mathbb{N}$ such that $\delta(t, a_{i_0+1} a_{i_0+2} ... a_{i_0+l_t})(F) > 0$. Let $l_0 = Max_{t \in H} l_t$, and $l \geq l_0$ such that $\delta(s, w_{|i_0+l}) = H$. Then $\rho_0 = w_{|i_0}$ and $\rho = a_{i_0+1}, ..., a_{i_0+l}$ satisfy the conditions of 3.

THEOREM 14. *The weak transience and strong transience problems (problem 4) are PSPACE complete.*

PROOF. The proof of the fact that these problems are in PSPACE is the same as for the strong recurrence problem: a nondeterministic Turing machine can guess ρ_0 and ρ and verify in PSPACE the requirements. Concerning the PSPACE hardness, we point out that the exact same reduction as for the strong recurrence problem is also a reduction for the Intersection of Regular Languages problem to our problem.

We can reduce the emptiness problem for an automaton in $PBA^{>0}$ to problem 2:

THEOREM 15. *Problem 2 is undecidable.*

4 Probabilistic automata

In this section we study our new classes of constrained probabilistic automata using the results from the previous section. We start our discussion with the class $CPBA^{>0}$. As a CPBA is structurally an FPT plus a set of final states, we can use the results of the last section, and the notion of probabilistic loop. A probabilistic loop on a CPBA will be *accepting* if its support contains an accepting state. For the following, we fix a CPBA $\mathcal{A} = (\mathcal{T}, F)$, where $\mathcal{T} = (S, \Sigma, \delta, \alpha)$ is an FPT and $F \subseteq S$. The past section yields the following theorem.

THEOREM 16. *The following are equivalent:*
1. $\mathcal{L}^{>0}(\mathcal{A}) \neq \emptyset$.
2. *$\mathcal{A}$ accepts a lasso shape word.*
3. *There exists an accessible and accepting probabilistic loop on $\mathcal{A}$.*

PROOF. $1 \Leftrightarrow 3$ comes from theorem 9. $2 \Rightarrow 1$ is direct. Suppose 3. If $x \in \Sigma^*$ is such that $\delta(\alpha, x)(s) > 0$ and $y \in \Sigma^*$ is the trace of the loop, the word $x \cdot y^\omega$ is a lasso shape word and belongs to the language of the automaton.

COROLLARY 17. *The emptiness problem of the language of a CPBA with the probable semantics is PSPACE complete.*

PROPOSITION 18. *$\mathcal{C}_{CPBA^{>0}}$ is not closed under complementation.*

In particular, the proof shows that the set of ω-regular languages is not a subset of the set of languages definable by automata in $CPBA^{>0}$. The following proposition, using a construction of [10], shows that the class of the complements of languages of automata in $PBA^{=1}$ is a subset of the class of languages recognized by automata in $CPBA^{>0}$.

PROPOSITION 19. *If $\mathcal{A} \in PBA$, there exists $\mathcal{A}' \in CPBA$ such that $|\mathcal{A}'| \leq |\mathcal{A}| + 1$ and $\mathcal{L}^{>0}(\mathcal{A}') = \mathcal{L}^{=1}(\mathcal{A})^c$. Moreover, the inclusion $\{\mathcal{L}^{=1}(\mathcal{A})^c | \mathcal{A} \in PBA\} \subsetneq \mathcal{C}_{CPBA^{>0}}$ is strict.*

As a corollary, since the emptiness of the language of a CPBA$^{>0}$ can be decided in PSPACE, this proves that the universality problem of the language of an automaton in $PBA^{=1}$ can be decided in PSPACE. The following proposition shows that the emptiness problems on the classes $PBA^{=1}$ and $CPBA^{=1}$ are equivalent. Given a probabilistic automaton $\mathcal{A}$ with final states set F, we can consider the language $\mathcal{L}^{PBA^{=1}}(\mathcal{A})$ of the set of words accepted by $\mathcal{A}$ when $\mathcal{A}$ is considered in $PBA^{=1}$, and also the language $\mathcal{L}^{CPBA^{=1}}(\mathcal{A})$ of the set of words accepted by $\mathcal{A}$ when $\mathcal{A}$ is considered in $CPBA^{=1}$.

PROPOSITION 20. *Let $\mathcal{A}$ be a probabilistic automaton with final states set F. Then:*

- $\mathcal{L}^{PBA^{=1}}(\mathcal{A}) = \emptyset$ *iff* $\mathcal{L}^{CPBA^{=1}}(\mathcal{A}) = \emptyset$.
- $\mathcal{L}^{PBA^{=1}}(\mathcal{A}) = \Sigma^\omega$ *iff* $\mathcal{L}^{CPBA^{=1}}(\mathcal{A}) = \Sigma^\omega$.

PROOF. Follows directly from propositions 11 and 13, as the condition on the probabilistic loop around the final state set is a structural condition, which does not depend on considerations on Inf or Supports sets of the runs.

By theorem 12 and theorem 14, the complexity of the emptiness problem and the universality problem of the language of an automaton in $CPBA^{=1}$ or in $PBA^{=1}$ is in PSPACE. This improves the previous results of [1] which showed using different tools that the emptiness problem for the language of an automaton in $PBA^{=1}$ is in EXPTIME. Note that the upcoming paper [5] shows PSPACE-completeness for the emptiness problem and the universality problem of the language of an automaton in $PBA^{=1}$.

PROPOSITION 21. *The class of languages $\mathcal{C}_{PBA^{=1}}$ is a subclass of $\mathcal{C}_{CPBA^{=1}}$, and the inclusion is strict.*

PROOF. The inclusion follows from the construction of a layered automaton, as in proposition 19. We can show that $\{w | a \in Supp(w)\} \in \mathcal{C}_{CPBA^{=1}} - \mathcal{C}_{PBA^{=1}}$.

We have seen that in contrast to (classical) probabilistic automata, for constrained probabilistic automata, the emptiness problem for Büchi acceptance under the probable semantics becomes decidable. However, for Street, resp. Muller acceptance condition, the emptiness problem for the probable semantics is undecidable. Surprisingly, for Rabin (and thus parity) acceptance, we can prove as for theorem 10 that the problem is decidable.

THEOREM 22. *The emptiness problem for CPSA, resp. CPMA, under the probable semantics is undecidable.*

PROOF. With $\mathsf{Acc} = \{(\{s\}, S), (\{t\}, S)\}$, resp. $\mathsf{Acc} = \{T : \{s,t\} \subseteq T \subseteq S\}$, problem 2 (two states strong recurrence) reduces to the emptiness problem for CPSA$^{>0}$, resp. CPMA$^{>0}$. As theorem 15 shows the undecidability of problem 2, the claim follows.

THEOREM 23. *The emptiness problem for CPRA under the probable semantics is decidable.*

Remarks: If we use the alternative definition for the support of a run, such that a state s is in the support of a run if the *Inf* limit of the time spent on s is non zero, we get different classes of automata, with different languages. However, the emptiness problem and all the natural problems can still be solved using the same tools. For instance, the language of an associated PCA automaton is still non empty iff there exists an accessible and accepting probabilistic loop. Thus, the complexity of the problems we studied does not change.

5 Conclusion

This paper presents an alternative definition to the the classical "infinitely often" Büchi condition. We presented several notions of recurrence and transience on finite probabilistic tables and gave the precise computational complexity of several of the associated problems. We used these results to prove the decidability of basic problems on new classes of probabilistic automata on infinite words. Several theoretical questions are still open, e.g., the complexity of the universal strong recurrence problem. The possibility to find classes of probabilistic automata on which the basic problems such as the emptiness problem are computable, and which may be used to specify relevant properties in a system verification context, could motivate future work. Another issue is in the context of infinite duration games, where we can change the classical ω-regular condition of [8], or the extensions of [6], by our notion of acceptance.

Bibliography

[1] C. Baier, N. Bertrand, and M. Grösser. On Decision Problems for Probabilistic Büchi Automata. In *Proc. of FOSSACS '08*, volume 4962 of *LNCS*. Springer, 2008.

[2] C. Baier and M. Grösser. Recognizing omega-regular Languages with Probabilistic Automata. In *Proc. of LICS '05*, pages 137–146. IEEE CS Press, 2005.

[3] M. Bojanczyk and T. Colcombet. Bounds in omega-regularity. In *Proc. of LICS '06*, pages 285–296. IEEE CS Press, 2006.

[4] D. Bustan, S. Rubin, and M.Y. Vardi. Verifying omega-Regular Properties of Markov Chains. In *Proc. of CAV '04*, volume 3114 of *LNCS*, pages 189–201. Springer, 2004.

[5] R. Chadha, A.P. Sistla, and M. Viswanathan. Power of randomization in automata on infinite strings. In *to appear in Proc. of the 20th International Conference on Concurrency Theory (CONCUR'09)*, Lecture Notes in Computer Science. Springer, 2009.

[6] K. Chatterjee, T.A. Henzinger, and F. Horn. Stochastic Games with Finitary Objectives.

[7] C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.

[8] L. de Alfaro and TA Henzinger. Concurrent omega-regular games. In *Logic in Computer Science, 2000. Proceedings. 15th Annual IEEE Symposium on*, pages 141–154, 2000.

[9] E. Grädel, W. Thomas, and T. Wilke. Automata, Logics, and Infinite Games: A Guide to Current Research, volume 2500 of LNCS. *Notes in Comp. Sci. Springer*, 2002.

[10] M. Größer. *Reduction Methods for Probabilistic Model Checking*. PhD thesis, Technische Universität Dresden, 2008.

[11] J.G. Kemeny and J.L. Snell. *Finite Markov Chains*. Springer-Verlag, 1983.

[12] D. Kozen. Lower bounds for natural proof systems. In *Foundations of Computer Science, 1977., 18th Annual Symposium on*, pages 254–266, 1977.

[13] V.G. Kulkarni. *Modeling and Analysis of Stochastic Systems*. Chapman & Hall/CRC, 1995.

[14] Z. Manna and A. Pnueli. *The temporal logic of reactive and concurrent systems: specification.* Springer.

[15] A. Paz. Ergodic theorems for infinite probabilistic tables. *Ann. Math. Statist*, 41:539–550, 1970.

[16] A. Paz. *Introduction to probabilistic automata*. Academic Press New York, 1971.

[17] E. Seneta. *Non-Negative Matrices and Markov Chains*. Springer, 2006.

[18] M.Y. Vardi. Automatic verification of probabilistic concurrent finite state programs. In *Foundations of Computer Science, 1984., 26th Annual Symposium on*, pages 327–338, 1985.