Infinite Synchronizing Words for Probabilistic Automata

Laurent Doyen[1], Thierry Massart[2], and Mahsa Shirmohammadi[2]

[1] LSV, ENS Cachan & CNRS, France
doyen@lsv.ens-cachan.fr
[2] Université Libre de Bruxelles, Brussels, Belgium *
thierry.massart@ulb.ac.be mahsa.shirmohammadi@ulb.ac.be

Abstract. Probabilistic automata are finite-state automata where the transitions are chosen according to fixed probability distributions. We consider a semantics where on an input word the automaton produces a sequence of probability distributions over states. An infinite word is accepted if the produced sequence is synchronizing, i.e. the sequence of the highest probability in the distributions tends to 1. We show that this semantics generalizes the classical notion of synchronizing words for deterministic automata. We consider the emptiness problem, which asks whether some word is accepted by a given probabilistic automaton, and the universality problem, which asks whether all words are accepted. We provide reductions to establish the PSPACE-completeness of the two problems.

1 Introduction

Probabilistic automata (PA) are finite-state automata where the transitions are chosen according to fixed probability distributions. In the traditional semantics, a run of a probabilistic automaton over an input word is a path (i.e., a sequence of states and transitions), and the *classical acceptance conditions* over runs (such as in finite automata, Büchi automata, etc.) are used to define the probability to accept a word as the measure of its accepting runs [11, 2]. Over finite and infinite words, several undecidability results are known about probabilistic automata in the traditional semantics [10, 1].

Recently, an alternative semantics for probabilistic automata has been proposed, with applications in sensor networks, queuing theory, and dynamical systems [9, 8, 5]. In this new semantics, a run over an input word is the sequence of probability distributions produced by the automaton. For an example, consider the probabilistic automaton with alphabet $\Sigma = \{a, b\}$ on Fig. 1 and the sequence of probability distributions produced by the input word $a(aba)^\omega$.

Previous works have considered *qualitative* conditions on this semantics. The space of probability distributions (which is a subset of $[0, 1]^n$) is partitioned into regions defined by linear predicates, and classical acceptance conditions are used to define accepting sequences of regions. It is known that reachability of a region is undecidable

* This work has been partly supported by the MoVES project (P6/39) which is part of the IAP-Phase VI Interuniversity Attraction Poles Programme funded by the Belgian State, Belgian Science Policy.

for linear predicates, and that it becomes decidable for a class of qualitative predicates which essentially constrain only the support of the probability distributions [8].

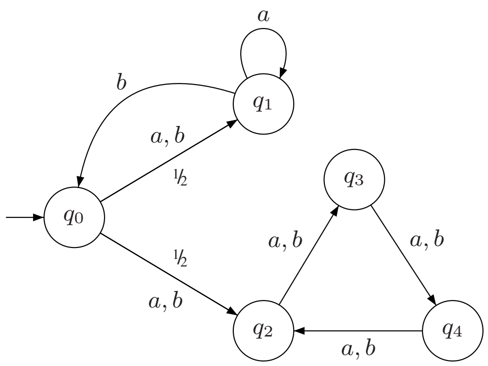
In this paper, we consider a *quantitative* semantics which has decidable properties, defined as follows [5]. A sequence $\bar{X} = X_0 X_1 \ldots$ of probability distributions over a set of states Q is *synchronizing* if in the long run, the probability mass tends to accumulate in a single state. More precisely, we consider two definitions: the sequence $\bar{X}$ is *strongly synchronizing* if $\liminf_{i\to\infty} \|X_i\| = 1$ where $\|X_i\| = \max_{q\in Q} X_i(q)$ is the highest probability in X_i; it is *weakly synchronizing* if $\limsup_{i\to\infty} \|X_i\| = 1$. Intuitively, strongly synchronizing means that the probabilistic automaton behaves in the long run like a deterministic system: eventually, at every step i (or at infinitely many steps for weakly synchronizing) there is a state $\hat{q}_i$ which accumulates almost all the probability, and therefore the sequence $\hat{q}_i \hat{q}_{i+1} \ldots$ is almost deterministic. Note that the state $\hat{q}_i$ needs not be the same at every step i. For instance, in the sequence in Fig. 1, the maximal probability in a state tends to 1, but it alternates between the three states q_2, q_3, and q_4. We define the synchronizing language $L(\mathcal{A})$ of a probabilistic automaton $\mathcal{A}$ as the set of words[3] which induce a synchronizing sequence of probability distributions. In this paper, we consider the decision problems of emptiness and universality for synchronizing language, i.e. deciding whether $L(\mathcal{A}) = \emptyset$, and $L(\mathcal{A}) = \mathcal{D}(\Sigma)^\omega$ respectively.

Synchronizing words have applications in planning, control of discrete event systems, biocomputing, and robotics [3, 15]. For deterministic finite automata (DFA), a (finite) word w is synchronizing if reading w from any state of the automaton always leads to the same state. Note that DFA are a special case of probabilistic automata. A previous generalization of synchronizing words to probabilistic automata was proposed by Kfoury, but the associated decision problem is undecidable [7]. By contrast, the results of this paper show that the definition of strongly and weakly synchronizing words is a decidable generalization of synchronized words for DFA. More precisely, we show that there exists a (finite) synchronizing word for a DFA $\mathcal{A}$ if and only if there exists an (infinite) synchronizing word for $\mathcal{A}$ viewed as a probabilistic automaton with uniform initial distribution over all states.

We show that the emptiness and universality problems for synchronizing languages is PSPACE-complete, for both strongly and weakly synchronizing semantics. For emptiness, the PSPACE upper bound follows from a reduction to the emptiness problem of an exponential-size Büchi automaton. The construction relies on an extension of the classical subset construction. The PSPACE lower bound is obtained by a reduction from the universality problem for nondeterministic finite automata.

For universality, the upper bound follows from a reduction to the emptiness problem of an exponential-size coBüchi automaton, and the lower bound is obtained by a reduction from the emptiness problem of traditional probabilistic coBüchi automata in positive semantics [4, 14].

[3] Words can be randomized, i.e. their letters can be probability distributions over the alphabet Σ. We denote by $\mathcal{D}(\Sigma)$ the set of all probability distributions over Σ.



$$\begin{array}{c} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \end{array} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \xrightarrow{a} \begin{pmatrix} 0 \\ 1/2 \\ 1/2 \\ 0 \\ 0 \end{pmatrix} \xrightarrow{a} \begin{pmatrix} 0 \\ 1/2 \\ 0 \\ 1/2 \\ 0 \end{pmatrix} \xrightarrow{b} \begin{pmatrix} 1/2 \\ 0 \\ 0 \\ 0 \\ 1/2 \end{pmatrix} \xrightarrow{a} \begin{pmatrix} 0 \\ 1/4 \\ 3/4 \\ 0 \\ 0 \end{pmatrix} \xrightarrow{aba} \begin{pmatrix} 0 \\ 1/8 \\ 7/8 \\ 0 \\ 0 \end{pmatrix} \xrightarrow{(aba)^{n-3}} \begin{pmatrix} 0 \\ 1/2^n \\ 1 - 1/2^n \\ 0 \\ 0 \end{pmatrix}$$

Fig. 1. The word $a(aba)^\omega$ is strongly synchronizing.

The PSPACE-completeness bounds improve the results of [5] where it is shown that the emptiness and universality problems for synchronizing languages are decidable[4] using a characterization which yields doubly exponential algorithms.

Due to lack of space, the details of some proofs are omitted and can be found in [6].

2 Automata and Synchronizing Words

A *probability distribution* over a finite set S is a function $d : S \to [0,1]$ such that $\sum_{s \in S} d(s) = 1$. The *support* of d is the set $\mathsf{Supp}(d) = \{s \in S \mid d(s) > 0\}$. We denote by $\mathcal{D}(S)$ the set of all probability distributions over S.

Given a finite alphabet Σ, we denote by Σ^* the set of all finite words over Σ, and by Σ^ω the set of all infinite words over Σ. The length of a word w is denoted by $|w|$ (where $|w| = \infty$ for infinite words). An infinite *randomized word* over Σ is a sequence $w = d_0 d_1 \dots$ of probability distributions over Σ. We denote by $\mathcal{D}(\Sigma)^\omega$ the set of all infinite randomized words over Σ. A word $w \in \Sigma^\omega$ can be viewed as a randomized word $d_0 d_1 \dots$ in which the support of all probability distributions d_i is a singleton. We sometimes call $w \in \Sigma^\omega$ a *pure word* to emphasize this.

Finite Automata. A nondeterministic finite automaton (NFA) $\mathcal{A} = \langle L, \ell_0, \Sigma, \delta, \mathcal{F} \rangle$ consists of a finite set L of states, an initial state $\ell_0 \in L$, a finite alphabet Σ, a transition relation $\delta : L \times \Sigma \to 2^L$, and an acceptance condition $\mathcal{F}$ which can be either finite, Büchi, or coBüchi (and then $\mathcal{F} \subseteq L$), or generalized Büchi (and then $\mathcal{F} \subseteq 2^L$).

[4] Probabilistic automata are equivalent to Markov decision processes with blind strategies.

Finite acceptance conditions define languages of finite words, other acceptance conditions define languages of infinite words. Automata with Büchi, coBüchi, and generalized Büchi condition are called ω-automata. A *run* over a (finite or infinite) word $w = \sigma_0\sigma_1 \dots$ is a sequence $\rho = r_0 r_1 \dots$ such that $r_0 = \ell_0$ and $r_{i+1} \in \delta(r_i, \sigma_i)$ for all $0 \leq i < |w|$. A finite run $r_0 \dots r_k$ is *accepting* if $r_k \in \mathcal{F}$, and an infinite run $r_0 r_1 \dots$ is *accepting* for a Büchi condition if $r_j \in \mathcal{F}$ for infinitely many j, for a coBüchi condition if $r_j \notin \mathcal{F}$ for finitely many j, for a generalized Büchi condition if for all $s \in \mathcal{F}$, we have $r_j \in s$ for infinitely many j.

The *language* of a (finite- or ω-) automaton is the set $L_{\mathsf{f}}(\mathcal{A})$ (resp., $L_\omega(\mathcal{A})$) of finite (resp., infinite) words over which there exists an accepting run. The *emptiness problem* for (finite- or ω-) automata is to decide, given an automaton $\mathcal{A}$, whether $L_{\mathsf{f}}(\mathcal{A}) = \emptyset$ (resp., $L_\omega(\mathcal{A}) = \emptyset$), and the *universality problem* is to decide whether $L_{\mathsf{f}}(\mathcal{A}) = \Sigma^*$ (resp., $L_\omega(\mathcal{A}) = \Sigma^\omega$). For both finite and Büchi automata, the emptiness problem is NLOGSPACE-complete, and the universality problem is PSPACE-complete [13, 12].

A *deterministic* finite automaton (DFA) is a special case of NFA where the transition relation is such that $\delta(\ell, \sigma)$ is a singleton for all $\ell \in L$ and $\sigma \in \Sigma$, which can be viewed as a function $\delta : L \times \Sigma \to L$, and can be extended to a function $\delta : L \times \Sigma^* \to L$ defined inductively as follows: $\delta(\ell, \epsilon) = \ell$ with ϵ the empty word and $\delta(\ell, \sigma \cdot w) = \delta(\delta(\ell, \sigma), w)$ for all $w \in \Sigma^*$. A *synchronizing* word for a DFA is a word $w \in \Sigma^*$ such that $\delta(\ell, w) = \delta(\ell', w)$ for all $\ell, \ell' \in L$, i.e. such that from all states, a unique state is reached after reading w. Synchronizing words have applications in several areas from planning to robotics and system biology, and they gave rise to the famous Černý's conjecture [3, 15].

Probabilistic Automata. A *probabilistic automaton* (PA) $\mathcal{A} = \langle Q, \mu_0, \Sigma, \delta \rangle$ consists of a finite set Q of states, an initial probability distribution $\mu_0 \in \mathcal{D}(Q)$, a finite alphabet Σ, and a probabilistic transition function $\delta : Q \times \Sigma \to \mathcal{D}(Q)$. In a state $q \in Q$, the probability to go to a state $q' \in Q$ after reading a letter $\sigma \in \Sigma$ is $\delta(q, \sigma)(q')$. Define $\mathsf{Post}(q, \sigma) = \mathsf{Supp}(\delta(q, \sigma))$, and for a set $s \subseteq Q$ and $\Sigma' \subseteq \Sigma$, let $\mathsf{Post}(s, \Sigma') = \bigcup_{q \in s} \bigcup_{\sigma \in \Sigma'} \mathsf{Post}(q, \sigma)$.

The *outcome* of an infinite randomized word $w = d_0 d_1 \dots$ is the infinite sequence $X_0 X_1 \dots$ of probability distributions $X_i \in \mathcal{D}(Q)$ such that $X_0 = \mu_0$ is the initial distribution, and for all $n > 0$ and $q \in Q$,

$$X_n(q) = \sum_{\sigma \in \Sigma} \sum_{q' \in Q} X_{n-1}(q') \cdot d_{n-1}(\sigma) \cdot \delta(q', \sigma)(q)$$

The *norm* of a probability distribution X over Q is $\|X\| = \max_{q \in Q} X(q)$. We say that w is a *strongly synchronizing* word if

$$\liminf_{n \to \infty} \|X_n\| = 1, \tag{1}$$

and that it is a *weakly synchronizing* word if

$$\limsup_{n \to \infty} \|X_n\| = 1. \tag{2}$$

Intuitively, a word is synchronizing if in the outcome the probability mass tends to concentrate in a single state, either at every step from some point on (for strongly synchronizing), or at infinitely many steps (for weakly synchronizing). Note that equivalently, the randomized word w is strongly synchronizing if the limit $\lim_{n\to\infty}\|X_n\|$ exists and equals 1. We denote by $\mathcal{L}_S(\mathcal{A})$ (resp., $\mathcal{L}_W(\mathcal{A})$) the set of strongly (resp., weakly) synchronizing words of $\mathcal{A}$.

In this paper, we are interested in the *emptiness problem* for strongly (resp., weakly) synchronizing languages which is to decide, given a probabilistic automaton $\mathcal{A}$, whether $\mathcal{L}_S(\mathcal{A}) = \emptyset$ (resp., $\mathcal{L}_W(\mathcal{A}) = \emptyset$), and in the *universality problem* which is to decide, whether $\mathcal{L}_S(\mathcal{A}) = \mathcal{D}(\Sigma)^\omega$ (resp., $\mathcal{L}_W(\mathcal{A}) = \mathcal{D}(\Sigma)^\omega$).

Synchronizing sequences of probability distributions have been first introduced for Markov decision processes (MDP) [5]. A probabilistic automaton can be viewed as an MDP where a word corresponds to a blind strategy (in the terminology of [5]) which chooses letters (or actions) independently of the sequence of states visited by the automaton and it only depends on the number of rounds that have been played so far. It is known that the problem of deciding the existence of a blind synchronizing strategy for MDPs is decidable[5] [5, Theorem 5]. In Section 3 we provide a solution in PSPACE to this problem, as well as a matching PSPACE lower bound.

Remark 1. From the results of [5], it follows that if there exists a (strongly or weakly) synchronizing word, then there exists a pure one.

A deterministic finite automaton is also a special case of probabilistic automaton where the probabilistic transition function is such that $\mathsf{Post}(q, \sigma)$ is a singleton for all $q \in Q$ and $\sigma \in \Sigma$ (and disregarding the initial distribution μ_0). We show that the definition of strongly (and weakly) synchronizing word generalizes to probabilistic automata the notion of synchronizing words for DFA, in the following sense.

Theorem 1. *Given a deterministic finite automaton $\mathcal{A}$, the following statements are equivalent:*

1. *There exists a (finite) synchronizing word for $\mathcal{A}$.*
2. *There exists an (infinite) strongly (or weakly) synchronizing word for $\mathcal{A}$ (viewed as a probabilistic automaton) with uniform initial distribution.*

Proof. First, if $w \in \Sigma^*$ is a synchronizing word for the DFA $\mathcal{A}$, there is a state q which is reached from all states of A by reading w. This implies that $X_{|w|}(q) = 1$ in the PA $\mathcal{A}$ (no matter the initial distribution) and since the transition function of $\mathcal{A}$ is deterministic, any infinite word with prefix w is both strongly (and thus also weakly) synchronizing for $\mathcal{A}$.

Second, assume that w is a strongly (or weakly) synchronizing word for the PA $\mathcal{A}$ with initial distribution μ_0 such that $\mu_0(q) = \frac{1}{m}$ where $m = |Q|$ is the number of states of $\mathcal{A}$. By Remark 1, we assume that $w = \sigma_0\sigma_1 \cdots \in \Sigma^\omega$ is pure. Let $X_0X_1 \ldots$ be the outcome of w in $\mathcal{A}$. Since the transitions in $\mathcal{A}$ are deterministic, all probabilities $X_i(q)$ for $i \geq 0$ and $q \in Q$ are multiples of $\frac{1}{m}$, i.e. $X_i(q) = \frac{c}{m}$ for some $0 \leq c \leq m$. Therefore, the fact that $\liminf_{n\to\infty} \|X_n\| = 1$ (or $\limsup_{n\to\infty} \|X_n\| = 1$) implies that $X_i(q) = 1$ for some $i \geq 0$ and $q \in Q$. Then, the finite word $\sigma_0\sigma_1 \ldots \sigma_{i-1}$ is synchronizing for $\mathcal{A}$. $\square$

[5] The results in [5] suggest a doubly exponential algorithm for solving this problem.

Note that the problem of deciding whether there exists a synchronizing word for a given DFA can be solved in polynomial time, while the emptiness problem for synchronizing languages (for probabilistic automata) is PSPACE-complete (see Theorem 2).

End-Components. A set $C \subseteq Q$ is *closed* if for every state $q \in C$, there exists $\sigma \in \Sigma$ such that $\mathsf{Post}(q, \sigma) \subseteq C$. For each $q \in C$, let $D_C(q) = \{\sigma \in \Sigma \mid \mathsf{Post}(q, \sigma) \subseteq C\}$. The graph induced by C is $\mathcal{A} \restriction C = (C, E)$ where E is the set of edges $(q, q') \in C \times C$ such that $\delta(q, \sigma)(q') > 0$ for some $\sigma \in D_C(q)$. An *end-component* is a closed set U such that the graph $\mathcal{A} \restriction C$ is strongly connected.

3 The Emptiness Problem is PSPACE-complete

In this section, we present constructions to reduce the emptiness problem for synchronizing languages of probabilistic automata to the emptiness problem for ω-automata, with Büchi condition for strongly synchronizing language, and with generalized Büchi condition for weakly synchronizing language. The constructions are exponential and therefore provide a PSPACE upper bound for the problems. We also prove a matching lower bound.

Lemma 1. *The emptiness problem for strongly synchronizing language of probabilistic automata is decidable in PSPACE.*

We give the main idea of the proof of Lemma 1. The details can be found in [6].

Given a PA $\mathcal{A} = \langle Q, \mu_0, \Sigma, \delta \rangle$, we construct a Büchi automaton $\mathcal{B} = \langle L, \ell_0, \Sigma, \delta_{\mathcal{B}}, \mathcal{F}_{\mathcal{B}} \rangle$ such that $\mathcal{L}_S(\mathcal{A}) = \emptyset$ iff $L(\mathcal{B}) = \emptyset$. The automaton $\mathcal{B}$ is exponential in the size of $\mathcal{A}$, and thus the PSPACE bound follows from the NLOGSPACE-completeness of the emptiness problem for Büchi automata.

The construction of $\mathcal{B}$ relies on the following intuition. A strongly synchronizing word induces a sequence of probability distributions X_i in which the probability mass tends to accumulate in a single state $\hat{q}_i$ at step i. It can be shown that for all sufficiently large i, there exists a deterministic transition from $\hat{q}_i$ to $\hat{q}_{i+1}$, i.e. there exists $\sigma_i \in \Sigma$ such that $\mathsf{Post}(\hat{q}_i, \sigma_i) = \{\hat{q}_{i+1}\}$. The Büchi automaton $\mathcal{B}$ will guess the *witness sequence* $\hat{q}_i \hat{q}_{i+1} \dots$ and check that the probability mass is 'injected' into this sequence. The state of $\mathcal{B}$ keeps track of the support $s_i = \mathsf{Supp}(X_i)$ of the outcome sequence on the input word, and at some point guesses that the witness sequence $\hat{q}_i \hat{q}_{i+1} \dots$ starts. Then, using an *obligation* set $o_i \subseteq s_i$, it checks that every state in s_i eventually 'injects' some probability mass in the witness sequence.

The construction of $\mathcal{B} = \langle L, \ell_0, \Sigma, \delta_{\mathcal{B}}, \mathcal{F}_{\mathcal{B}} \rangle$ is as follows:

- $L = 2^Q \cup (2^Q \times 2^Q \times Q)$ is the set of states. A state $s \subseteq Q$ is the support of the current probability distribution. A state $(s, o, \hat{q}) \in 2^Q \times 2^Q \times Q$ consists of the support s, the obligation set $o \subseteq s$, and a state $\hat{q} \in s$ of the witness sequence.
- $\ell_0 = \mathsf{Supp}(\mu_0)$ is the initial state.
- $\delta_{\mathcal{B}} : L \times \Sigma \to 2^L$ is defined as follows. For all $s \in 2^Q$ and $\sigma \in \Sigma$, let $s' = \mathsf{Post}(s, \sigma)$, and define $\delta_{\mathcal{B}}(s, \sigma) = \{s'\} \cup \{(s', s', \hat{q}) \mid \hat{q} \in s'\}$. For all $(s, o, \hat{q}) \in 2^Q \times 2^Q \times Q$ and $\sigma \in \Sigma$, let $s' = \mathsf{Post}(s, \sigma)$. If $\mathsf{Post}(\hat{q}, \sigma)$ is not a singleton, then $\delta_{\mathcal{B}}((s, o, \hat{q}), \sigma) = \emptyset$, otherwise let $\{\hat{q}'\} = \mathsf{Post}(\hat{q}, \sigma)$, and

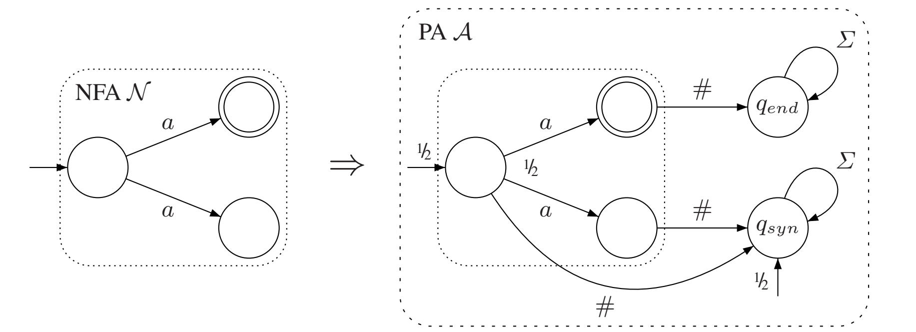


Fig. 2. Sketch of the reduction for PSPACE-hardness of the emptiness problem.

  - if $o \neq \emptyset$, then $\delta_{\mathcal{B}}((s, o, \hat{q}), \sigma) = \{(s', o' \setminus \{\hat{q}'\}, \hat{q}') \mid \forall q \in o : o' \cap \mathsf{Post}(q, \sigma) \neq \emptyset\}$,
  - if $o = \emptyset$, then $\delta_{\mathcal{B}}((s, o, \hat{q}), \sigma) = \{(s', s', \hat{q}')\}$.
- $\mathcal{F}_{\mathcal{B}} = \{(s, o, \hat{q}) \in 2^Q \times 2^Q \times Q \mid o = \emptyset\}$ is the set of accepting states.

Lemma 2. *The emptiness problem for weakly synchronizing language of probabilistic automata is decidable in PSPACE.*

The proof of Lemma 2 is by a reduction to the emptiness problem of an exponential-size ω-automaton with generalized Büchi condition. It can be found in [6].

Lemma 3. *The emptiness problem for strongly synchronizing language and for weakly synchronizing language of probabilistic automata is PSPACE-hard.*

Proof. We present a proof for strongly synchronizing words using a reduction from the universality problem for nondeterministic finite automata. The proof and the reduction for weakly synchronizing words is analogous.

Given a NFA $\mathcal{N}$, we construct a PA $\mathcal{A}$, such that $L(\mathcal{N}) = \Sigma^*$ iff $\mathcal{L}_S(\mathcal{A}) = \emptyset$. The reduction is illustrated in Fig. 2. The nondeterministic transitions of $\mathcal{N}$ become probabilistic in $\mathcal{A}$ with uniform probability. The initial probability distribution assigns probability $\frac{1}{2}$ to the absorbing state q_{sync}. Therefore, a synchronizing word needs to inject all that probability into q_{sync}. This can be done with the special symbol $\#$ from the non-accepting states of the NFA. From the accepting states, the $\#$ symbol leads to a sink state q_{end} from which there is no way to synchronize the automaton.

Let $\mathcal{N} = \langle L, \ell_0, \Sigma, \delta_{\mathcal{N}}, \mathcal{F}_{\mathcal{N}} \rangle$ be a NFA, we construct the PA $\mathcal{A} = \langle Q, \mu_0, \Sigma', \delta, \mathcal{F} \rangle$ as follows:

- $Q = L \cup \{q_{sync}, q_{end}\}$.
- $\mu_0(\ell_0) = \mu_0(q_{sync}) = \frac{1}{2}$, and $\mu_0(q) = 0$ for all $q \in Q \setminus \{\ell_0, q_{sync}\}$.
- $\Sigma' = \Sigma \cup \{\#\}$.

– $\delta : Q \times \Sigma \to \mathcal{D}(Q)$ is the probabilistic transition function defined as follows. For all $\sigma \in \Sigma'$, $\delta(q_{sync}, \sigma)(q_{sync}) = 1$ and $\delta(q_{end}, \sigma)(q_{end}) = 1$. For all $q \in \mathcal{F}_N$, $\delta(q, \#)(q_{end}) = 1$, and for all $q \notin \mathcal{F}_N$, $\delta(q, \#)(q_{sync}) = 1$. Finally, for all $q, q' \in L$ and $\sigma \in \Sigma$, $\delta(q, \sigma)(q') = \frac{1}{|\delta_N(q,\sigma)|}$ if $q' \in \delta_N(q, \sigma)$, and $\delta(q, \sigma)(q') = 0$ otherwise.

We show that $L(\mathcal{N}) \neq \Sigma^*$ iff $\mathcal{L}_S(\mathcal{A}) \neq \emptyset$. First, assume that $L(\mathcal{N}) \neq \Sigma^*$. Let $w \in \Sigma^*$ such that $w \notin L(\mathcal{N})$. Then all runs of $\mathcal{N}$ over w end in a non-accepting state, and in $\mathcal{A}$ the state q_{sync} is reached with probability 1 on the word $w \cdot \#$. Therefore, $w \cdot (\#)^\omega$ is a strongly synchronizing word for $\mathcal{A}$ and $\mathcal{L}_S(\mathcal{A}) \neq \emptyset$.

Second, assume that $\mathcal{L}_S(\mathcal{A}) \neq \emptyset$. Let $w' \in \mathcal{L}_S(\mathcal{A})$ be a strongly synchronizing word for $\mathcal{A}$, and let $X_0 X_1 \ldots$ be the outcome of w' in $\mathcal{A}$. Since $\mu_0(q_{sync}) = \frac{1}{2}$ and q_{sync} is a sink state, we have $X_k(q_{sync}) \geq \frac{1}{2}$ for all $k \geq 0$ and since w' is strongly synchronizing, it implies that $\lim_{k\to\infty} X_k(q_{sync}) = 1$. Then w' has to contain $\#$, as this is the only letter on a transition from a state in L to q_{sync}. Let $w \in \Sigma^*$ be the prefix of w' before the first occurrence of $\#$. We claim that w is not accepted by $\mathcal{N}$. By contradiction, if there is an accepting run r of $\mathcal{N}$ over w, then positive probability is injected in q_{end} by the finite word $w \cdot \#$ and stays there forever, in contradiction with the fact that $\lim_{k\to\infty} X_k(q_{sync}) = 1$. Therefore $w \notin L(\mathcal{N})$ and $L(\mathcal{N}) \neq \Sigma^*$. □

The following result follows from Lemma 1, Lemma 2, and Lemma 3.

Theorem 2. *The emptiness problem for strongly synchronizing language and for weakly synchronizing language of probabilistic automata is PSPACE-complete.*

4 The Universality Problem is PSPACE-complete

In this section, we present necessary and sufficient conditions for probabilistic automata to have a universal strongly (resp., weakly) synchronizing language. We show that the construction can be checked in PSPACE. Unlike for the emptiness problem, it is not sufficient to consider only pure words for universality of strongly (resp., weakly) synchronizing languages. For instance, all infinite pure words for the probabilistic automaton in Fig. 3 are strongly (and weakly) synchronizing, but the uniformly randomized word over $\{a, b\}$ is not strongly (nor weakly) synchronizing. Formally, we say an infinite randomized word is a uniformly randomized word over Σ denoted by w_u, if $d_i(\sigma) = \frac{1}{|\Sigma|}$ for all $\sigma \in \Sigma$ and $i \in \mathbb{N}$.

Lemma 4. *There is a probabilistic automaton for which all pure words are strongly synchronizing, but not all randomized words .*

The reason is that there are two sets ($\{q_1\}$ and $\{q_2\}$) for which the probability can not go out. For a given PA $\mathcal{A} = \langle Q, \mu_0, \Sigma, \delta, \mathcal{F}\rangle$, a maximal end-component $U \subseteq Q$ is *terminal*, if $\mathsf{Post}(U, \Sigma) \subseteq U$. It is easy to see that a terminal end-component keeps probability inside. To have a universal strongly/weakly synchronizing language, the PA $\mathcal{A}$ needs to have only a unique terminal end-component. Otherwise, the uniformly randomized word w_u would reach all terminal end-components and would not

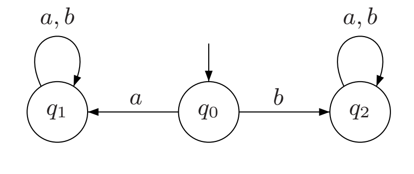


Fig. 3. Randomization is necessary.

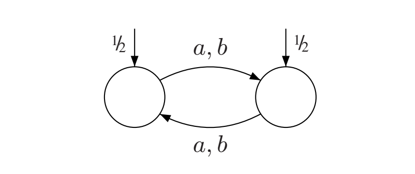


Fig. 4. Randomization is not sufficient.

be strongly synchronizing. Though having only a terminal end-component is necessary, it is not sufficient. For example, the infinite word $(ab)^\omega \notin \mathcal{L}_S(A)$ for the PA $\mathcal{A}$ in Fig. 5 which contains only one terminal end-component. The probabilistic automaton needs to ensure that for all randomized words, all of the probability mass tends to accumulate in the unique terminal end-component. We express this property for a terminal end-component as being absorbing. We say that a terminal end-component U is *absorbing*, if $\lim_{n\to\infty} \sum_{q\in U} X_n(q) = 1$ for the outcome $X_0X_1\ldots$ of all infinite randomized words $w \in D(\Sigma)^\omega$. Fig. 6 shows an automaton where the unique end competent is absorbing and the strongly synchronizing language is universal.

Lemma 5. *For a given PA $\mathcal{A}$, deciding whether a given terminal end-component U is absorbing is decidable in PSPACE.*

We give the main idea of the proof of Lemma 5. The details are provided in [6].

Given a terminal end-component $U \subseteq Q$ of the PA $\mathcal{A} = \langle Q, \mu_0, \Sigma, \delta, \mathcal{F}\rangle$, we construct a coBüchi automaton $\mathcal{C} = \langle L, \ell_0, 2^\Sigma, \delta_\mathcal{C}, \mathcal{F}_\mathcal{C}\rangle$ such that U is absorbing iff $L(\mathcal{C}) = \emptyset$. The coBüchi automaton $\mathcal{C}$ is exponential in the size of $\mathcal{A}$, and as a consequence of NLOGSPACE-completeness of the emptiness problem for coBüchi automata, the PSPACE bound follows.

The automaton $\mathcal{C}$ is constructed to guess an infinite word w as a witness, to prove that the terminal end-component U is not absorbing. This word induces an infinite sequence of supports $s_0s_1s_2\ldots$ produced by its outcome $X_0X_1X_2\ldots$ (i.e., $s_i = \mathsf{Supp}(X_i)$ for all $i \in \mathbb{N}$). At some point n, there is a subset $s \subseteq s_n$ form which U cannot be reached. Therefore, the states of $\mathcal{C}$ keeps track of the support $s_i = \mathsf{Supp}(X_i)$ of the outcome, and at some point guesses the set s and checks that U is never reached from states in s. Then the acceptance condition requires that eventually all the reached states are outside of the end-component U. Since, by Lemma 4, the pure words are not sufficient, the alphabet of $\mathcal{C}$ is 2^Σ. A word over this alphabet is a sequence of subsets of letters which can be viewed as the sequence of supports of the distributions of a randomized word.

The construction of $\mathcal{C} = \langle L, \ell_0, 2^\Sigma, \delta_\mathcal{C}, \mathcal{F}_\mathcal{C}\rangle$ is as follows:

- $L = 2^Q \times \{0, 1\}$.
- $\ell_0 = (\mathsf{Supp}(\mu_0), 0)$ is the initial state.
- $2^\Sigma \setminus \{\emptyset\}$ is the alphabet.
- $\delta_\mathcal{C} : L \times 2^\Sigma \to 2^L$ is the transition function defined as follows. For all $s \subseteq Q$ and $\Sigma' \subseteq \Sigma$, let $s' = \mathsf{Post}(s, \Sigma')$ and define $\delta_\mathcal{C}((s, 0)) = \{(s', 0)\} \cup \{(s'', 1) \mid s'' \neq$

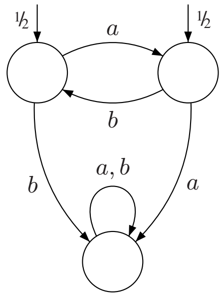


Fig. 5. Non-absorbing end-component.

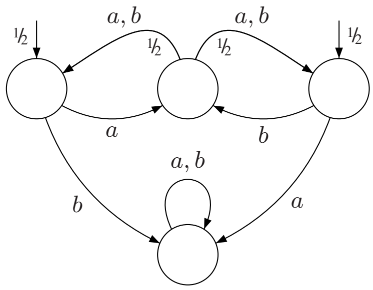


Fig. 6. Absorbing end-component.

$\emptyset \wedge s'' \subseteq s' \setminus U\}$ and define $\delta_{\mathcal{C}}((s,1)) = \{(s',1)\}$ if $s' \cap U = \emptyset$, and $\delta_{\mathcal{C}}((s,1)) = \emptyset$ otherwise.
- and $\mathcal{F}_{\mathcal{C}} = 2^Q \times \{1\}$ is the coBüchi acceptance condition.

Another necessary condition to have a universal strongly (resp., weakly) synchronizing language for a probabilistic automaton is that the uniformly randomized word is synchronizing as well. For instance, the automaton presented in Fig. 4 has an absorbing end-component, but since the uniformly randomized word is not strongly synchronizing, the strongly synchronizing language is not universal.

Lemma 6. *The universality problem for strongly synchronizing language and for weakly synchronizing language of probabilistic automata is decidable in PSPACE.*

We state the main idea of the proof of Lemma 6 for strongly synchronizing languages. The detailed proof can be found in [6]. The proof for weakly synchronizing languages follows an analogous discussion which is left to the reader.

We establish the following characterization. The synchronizing language of a given PA $\mathcal{A}$ is universal iff (I) there is a (then necessarily unique) absorbing end-component in $\mathcal{A}$, and (II) the uniformly randomized word w_u is strongly (resp., weakly) synchronizing. The above arguments show that these conditions are necessary and we now briefly explain why they are also sufficient. Since the uniformly randomized word w_u is strongly synchronizing, it can be shown that the unique terminal end-component U of $\mathcal{A}$ consists of a simple cycle, in the sense that $|\mathsf{Post}(q, \Sigma)| = 1$ for all states $q \in U$. It follows that if word w is not strongly synchronizing, then two different states of U would be reached after the same number of steps. But since all states reachable by w are also reachable by w_u, it would mean that w_u is not strongly synchronizing, a contradiction.

Condition (I) can be checked in PSPACE by Lemma 5, and Condition (II) reduces to check that a Markov chain is synchronizing, which can be done in polynomial time by steady state analysis. The PSPACE bound follows.

Lemma 7. *The universality problem for strongly synchronizing language and for weakly synchronizing language of probabilistic automata is PSPACE-hard.*

Proof. We present a proof using a reduction from a PSPACE-complete problem so called *initial state problem*. Given a nondeterministic finite automaton $\mathcal{N} = \langle Q, q_0, \Sigma, \delta, \mathcal{F}\rangle$ and a state $q \in Q$, we denote by $\mathcal{N}_q$ the automaton $\mathcal{N}$ in which the initial state is q, i.e. $\mathcal{N}_q = \langle Q, q, \Sigma, \delta, \mathcal{F}\rangle$. The *initial state problem* is to decide, given $\mathcal{N}$, whether the exists a state $q \in Q$ and a word $w \in \Sigma^\omega$ such that all runs r of $\mathcal{N}_q$ over w avoid $\mathcal{F}$, i.e. $r_i \notin \mathcal{F}$ for all $i \geq 0$. From the results of [4, 14], it follows that the initial state problem is PSPACE-complete. We present a polynomial-time reduction from the the initial state problem to the universality problem, establishing the PSPACE hardness of the universality problem.

Given an NFA $\mathcal{N} = \langle L, \ell_0, \Sigma, \delta_{\mathcal{N}}, \mathcal{F}_{\mathcal{N}}\rangle$ with $\mathcal{F}_{\mathcal{N}} \neq \emptyset$, we construct a PA $\mathcal{A} = \langle Q, \mu_0, \Sigma, \delta\rangle$ as follows:

- $Q = L \cup \{q_{end}\}$.
- $\mu_0(\ell_0) = 1$, and $\mu_0(q) = 0$ for all $q \in Q \setminus \{\ell_0\}$.
- $\delta : Q \times \Sigma \to \mathcal{D}(Q)$ is the probabilistic transition function defined as follows. For all $q \in Q$ and $\sigma \in \Sigma$, if $q \notin \mathcal{F}$, then $\delta(q, \sigma)$ is the uniform distribution over $\delta_{\mathcal{N}}(q, \sigma)$, and if $q \in \mathcal{F}_{\mathcal{N}}$, $\delta(q, \sigma)(q') = \frac{1}{2|\delta_{\mathcal{N}}(q,\sigma)|}$ for all $q' \in \delta_{\mathcal{N}}(q, \sigma)$ and $\delta(q, \sigma)(q_{end}) = \frac{1}{2}$.

We show that the answer to the initial state problem for $\mathcal{N}$ is YES if and only if $\mathcal{A}$ is not universal. We assume w.l.o.g that all states in $\mathcal{N}$ are reachable. First, if the answer to the initial state problem for $\mathcal{N}$ is YES, then let $\hat{q}$ be an initial state and $w \in \Sigma^\omega$ be a word satisfying the problem. We construct a word that is not (strongly neither weakly) synchronizing for $\mathcal{A}$. First, consider the $|Q|$-times repetition of the uniform distribution d_u over Σ. Then, with positive probability the state q_{end} is reached, and also with positive probability the state $\hat{q}$ is reached, say after k steps. Let $w' \in \Sigma^\omega$ such that $w = v \cdot w'$ and $|v| = |Q| - k$. Note that from state $\hat{q}$ the finite word v is played with positive probability by the repetition of uniform distribution d_u. Therefore, on the word $(d_u)^{|Q|} \cdot w'$, with some positive probability the set q_{end} is never reached, and thus it is not synchronizing, and $\mathcal{A}$ is not universal. Second, if $\mathcal{A}$ is not universal, then the terminal end-component $\{q_{end}\}$ is not absorbing and by the construction in Lemma 5, there exists a state $\hat{q}$ and a pure word $w \in \Sigma^\omega$ such that all runs from $\hat{q}$ on w avoid q_{end}, and therefore also avoid $\mathcal{F}_{\mathcal{N}}$. Hence, the answer to the initial state problem for $\mathcal{N}$ is YES. $\square$

The following result follows from Lemma 6, and Lemma 7.

Theorem 3. *The universality problem for strongly synchronizing language and for weakly synchronizing language of probabilistic automata is PSPACE-complete.*

5 Discussion

The complexity results of this paper show that both the emptiness and the universality problems for synchronizing languages are PSPACE-complete. The results in this paper apply also to a more general definition of synchronizing sequence of probability distribution, where groups of equivalent states are clustered together. A labeling function assigns a color to each group of equivalent states. The definition of synchronizing sequences then corresponds to the requirement that the automaton essentially behaves deterministically according to the sequence of colors produced in the long run. A labeled probabilistic automaton is a PA $\mathcal{A}\langle Q, \mu_0, \Sigma, \delta\rangle$ with a labeling function $\mathsf{L} : Q \to \Gamma$ where Γ is a finite set of colors. The L-norm of a probability distribution $X \in \mathcal{D}(Q)$ is $\|X\|_{\mathsf{L}} = \max_{\gamma \in \Gamma} \sum_{q:\mathsf{L}(q)=\gamma} X(q)$, and a sequence $X_0 X_1 \dots$ is strongly synchronizing (resp., weakly synchronizing) if $\liminf_{n\to\infty} \|X_n\|_{\mathsf{L}} = 1$, (resp., $\limsup_{n\to\infty} \|X_n\|_{\mathsf{L}} = 1$). The constructions of ω-automata in Lemma 1 and Lemma 2 can be adapted to show that the emptiness problem remains in PSPACE for labeled probabilistic automata. Roughly, the ω-automaton will guess the witness sequence $\hat{\gamma}_i \hat{\gamma}_{i+1} \dots$ of colors rather than a witness sequence of states. The solution of the universality problem is adapted analogously.

References

1. Baier, C., Bertrand, N., Größer, M.: On decision problems for probabilistic Büchi automata. In: Proc. of FoSSaCS: Foundations of Software Science and Computational Structures. pp. 287–301. LNCS 4962, Springer (2008)
2. Baier, C., Größer, M.: Recognizing omega-regular languages with probabilistic automata. In: Proc. of LICS: Logic in Comp. Science. pp. 137–146. IEEE (2005)
3. Benenson, Y., Adar, R., Paz-Elizur, T., Livneh, Z., e. Shapiro: DNA molecule provides a computing machine with both data and fuel. Proc. National Acad. Sci. USA 100, 2191–2196 (2003)
4. Chadha, R., Sistla, A.P., Viswanathan, M.: On the expressiveness and complexity of randomization in finite state monitors. In: Proc. of LICS: Logic in Computer Science. pp. 18–29. IEEE Comp. Soc. (2008)
5. Doyen, L., Massart, T., Shirmohammadi, M.: Synchronizing objectives for Markov decision processes. In: Proc. of iWIGP. pp. 61–75 (2011)
6. Doyen, L., Massart, T., Shirmohammadi, M.: Infinite synchronizing words for probabilistic automata. Tech. Rep. 2011.138, Université Libre de Bruxelles (ULB), Belgium (06 - 2011)
7. Kfoury, D.: Synchronizing sequences for probabilistic automata. Studies in Applied Mathematics 29, 101–103 (1970)
8. Korthikanti, V.A., Viswanathan, M., Kwon, Y., Agha, G.: Reasoning about mdps as transformers of probability distributions. In: Proc. of QEST: Quantitative Evaluation of Systems. pp. 199–208. IEEE Computer Society (2009)
9. Kwon, Y., Agha, G.: Linear inequality ltl (iltl): A model checker for discrete time Markov chains. In: ICFEM. pp. 194–208 (2004)
10. Paz, A.: Introduction to Probabilistic Automata. Academic Press, New York (1971)
11. Rabin, M.O.: Probabilistic automata. Information and Control 6(3), 230–245 (1963)
12. Sistla, A.P., Vardi, M.Y., Wolper, P.: The complementation problem for Büchi automata with applications to temporal logic. Theor. Comput. Sci. 49, 217–237 (1987)
13. Stockmeyer, L., Meyer, A.: Word problems requiring exponential time. In: Proceedings of the 5th Annual Symposium on Theory of Computing. pp. 1–9. ACM Press (1973)
14. Tracol, M., Baier, C., Größer, M.: Recurrence and transience for probabilistic automata. In: Proc. of FSTTCS: Foundations of Software Technology and Theoretical Computer Science. LIPIcs, vol. 4, pp. 395–406. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2009)
15. Volkov, M.V.: Synchronizing automata and the cerny conjecture. In: Proc. of LATA: Language and Automata Theory and Applications. pp. 11–27. LNCS 5196, Springer (2008)