Probabilistic Automata on Finite words: Decidable and Undecidable Problems

Hugo Gimbert, Youssouf Oualhadj

Probabilistic Automata on Finite Words: Decidable and Undecidable Problems

Hugo Gimbert[1] and Youssouf Oualhadj[2]

[1] LaBRI, CNRS, France
hugo.gimbert@labri.fr
[2] LaBRI, Université Bordeaux 1, France
youssouf.oualhadj@labri.fr

Abstract. This paper tackles three algorithmic problems for probabilistic automata on finite words: the Emptiness Problem, the Isolation Problem and the Value 1 Problem. The *Emptiness Problem* asks, given some probability $0 \leq \lambda \leq 1$, whether there exists a word accepted with probability greater than λ, and the *Isolation Problem* asks whether there exist words whose acceptance probability is arbitrarily close to λ. Both these problems are known to be undecidable [11, 4, 3]. About the Emptiness problem, we provide a new simple undecidability proof and prove that it is undecidable for automata with as few as two probabilistic transitions. The *Value* 1 *Problem* is the special case of the Isolation Problem when $\lambda = 1$ or $\lambda = 0$. The decidability of the Value 1 Problem was an open question. We show that the Value 1 Problem is undecidable. Moreover, we introduce a new class of probabilistic automata, $\sharp$-*acyclic automata*, for which the Value 1 Problem is decidable.

Introduction

Probabilistic automata on finite words are a computation model introduced by Rabin [12]. Like deterministic automata on finite words, a probabilistic automaton reads finite words from a finite alphabet A. Each time a new letter $a \in A$ is read, a transition from the current state $s \in Q$ to a new state $t \in Q$ occur. In a deterministic automaton, t is a function of s and a. In a probabilistic automaton, a lottery determines the new state, according to transition probabilities which depend on the current state s and letter a.

Since the seminal paper of Rabin, probabilistic automata on finite words have been extensively studied, see [6] for a survey of 416 papers and books about probabilistic automata published in the 60s and 70s.

Quite surprisingly, relatively few *algorithmic* results are known about probabilistic automata on finite words and almost all of them are undecidability results. There are two main algorithmic problems for probabilistic automata on finite words: the Emptiness Problem and the Isolation Problem. The *Emptiness Problem* asks, given some probability $0 \leq \lambda \leq 1$, whether there exists a word accepted with probability greater than λ, while the *Isolation Problem* asks whether there exist words whose acceptance probability is arbitrarily close to

λ. Both these problems were shown undecidable, respectively by Paz [11] and Bertoni [4, 3]. To our knowledge, most decidability results known for probabilistic automata on *finite* words are rather straightforward: they either apply to one-letter probabilistic automata, in other words Markov chains, or to problems where the probabilistic nature of the automaton is not taken into account. A notable exception is the decidability of language equality [13].

In contrast, several algorithmic results were proved for probabilistic automata on *infinite* words. The existence of an infinite word accepted with probability 1 is decidable [1]. For probabilistic Büchi automata [2], the emptiness problem may be decidable or not depending on the acceptance condition. A class of probabilistic Büchi automata which recognize exactly ω-regular languages was presented in [7]. In this paper, we only consider automata on finite words but several of our results seem to be extendable to probabilistic automata on infinite words.

Our contributions are the following.

First, we provide in Section 2.1 a new proof for the undecidability of the Emptiness Problem.

Second, we strengthen the result of Paz: the Emptiness Problem is undecidable even for automata with as few as two probabilistic transitions (Proposition 4).

Third, we solve an open problem: Bertoni's result shows that for any fixed cut-point $0 < \lambda < 1$, the Isolation Problem is undecidable. However, as stated by Bertoni himself, the proof seems hardly adaptable to the symmetric cases $\lambda = 0$ and $\lambda = 1$. We show that both these cases are undecidable as well, in other words the Value 1 Problem is undecidable (Theorem 4).

Fourth, we introduce a new class of probabilistic automata, $\sharp$-*acyclic automata*, for which the Value 1 Problem is decidable (Theorem 5 in Section 4). To our opinion, this is the main contribution of the paper. Moreover, this result is a first step towards the design of classes of stochastic games with partial observation for which the value 1 problem is decidable.

These undecidability results show once again that probabilistic automata are very different from deterministic and non-deterministic automata on finite of infinite words, for which many algorithmic problems are known to be decidable (e.g. emptiness, universality, equivalence). Surprisingly maybe, we remark that several natural decision problems about deterministic and non-deterministic automata are undecidable as well (Corollaries 1 and 2).

Due to space restrictions most proofs are omitted, they can be found in [**?**].

1 Probabilistic Automata

A probability distribution on Q is a mapping $\delta \in [0,1]^Q$ such that $\sum_{s \in S} \delta(s) = 1$. The set $\{s \in Q \mid \delta(s) > 0\}$ is called the support of δ and denoted $\mathrm{Supp}(\delta)$. For every non-empty subset $S \subseteq Q$, we denote δ_S the uniform distribution on S defined by $\delta(q) = 0$ if $q \notin S$ and $\delta(q) = \frac{1}{|S|}$ if $q \in S$. We denote $\mathcal{D}(Q)$ the set of probability distributions on Q.

Formally, a probabilistic automaton is a tuple $\mathcal{A} = (Q, A, (M_a)_{a\in A}, q_0, F)$, where Q is a finite set of states, A is the finite input alphabet, $(M_a)_{a\in A}$ are the transition matrices, q_0 is the initial state and F is the set of accepting states. For each letter $a \in A$, $M_a \in [0,1]^{Q\times Q}$ defines transition probabilities: $0 \leq M_a(s,t) \leq 1$ is the probability to go from state s to state t when reading letter a. Of course, for every $s \in S$ and $a \in A$, $\sum_{t\in S} M_a(s,t) = 1$, in other word in the matrix M_a, the line with index s is a probability distribution on Q.

Transition matrices define a natural action of A^* on $\mathcal{D}(Q)$. For every word $a \in A$ and $\delta \in \mathcal{D}(S)$, we denote $\delta \cdot a$ the probability distribution in $\mathcal{D}(Q)$ defined by $(\delta \cdot a)(t) = \sum_{s\in Q} \delta(s) \cdot M_a(s,t)$. This action extends naturally to words of A_*: $\forall w \in A^*, \forall a \in A, \delta \cdot (wa) = (\delta \cdot w) \cdot a$.

The computation of $\mathcal{A}$ on an input word $w = a_0 \ldots a_n \in A^*$ is the sequence $(\delta_0, \delta_1, \ldots, \delta_n) \in \mathcal{D}(Q)^{n+1}$ of probability distributions over Q such that $\delta_0 = \delta_{\{q_0\}}$ and for $0 \leq i < n, \delta_{i+1} = \delta_i \cdot a_i$.

For every state $q \in Q$ and for every set of states $R \subseteq Q$, we denote $\mathbb{P}_{\mathcal{A}}(q \xrightarrow{w} R) = \sum_{r\in R}(\delta_q \cdot w)(r)$ the probability to reach the set R from state q when reading the word w.

Definition 1 (Value and acceptance probability). *The* acceptance probability *of a word $w \in A^*$ by $\mathcal{A}$ is $\mathbb{P}_{\mathcal{A}}(w) = \mathbb{P}_{\mathcal{A}}(q_0 \xrightarrow{w} F)$. The* value *of $\mathcal{A}$, denoted $val(\mathcal{A})$, is the supremum acceptance probability: $val(\mathcal{A}) = \sup_{w\in A^*} \mathbb{P}_{\mathcal{A}}(w)$.*

2 The Emptiness Problem

Rabin defined the language recognized by a probabilistic automaton as $\mathcal{L}_{\mathcal{A}}(\lambda) = \{w \in A^* \mid \mathbb{P}_{\mathcal{A}}(w) \geq \lambda\}$, where $0 \leq \lambda \leq 1$ is called the cut-point. Hence, a canonical decision problem for probabilistic automata is:

Problem 1 (Emptiness Problem) *Given a probabilistic automaton $\mathcal{A}$ and $0 \leq \lambda \leq 1$, decide whether there exists a word w such that $\mathbb{P}_{\mathcal{A}}(w) \geq \lambda$.*

The *Strict Emptiness Problem* is defined the same way except the large inequality $\mathbb{P}_{\mathcal{A}}(w) \geq \lambda$ is replaced by a strict inequality $\mathbb{P}_{\mathcal{A}}(w) > \lambda$.

The special cases where $\lambda = 0$ and $\lambda = 1$ provide a link between probabilistic and non-deterministic automata on finite words. First, the Strict Emptiness Problem for $\lambda = 0$ reduces to the emptiness problem of non-deterministic automata, which is decidable in non-deterministic logarithmic space. Second, the Emptiness Problem for $\lambda = 1$ reduces to the universality problem for non-deterministic automata, which is PSPACE-complete [9]. The two other cases are trivial: the answer to the Emptiness Problem for $\lambda = 0$ is always yes and the answer to the Strict Emptiness Problem for $\lambda = 1$ is always no.

In the case where $0 < \lambda < 1$, both the Emptiness and the Strict Emptiness Problems are undecidable, which was proved by Paz [11]. The proof of Paz is a reduction from an undecidable problem about free context grammars. An alternative proof was given by Madani, Hanks and Condon [10], based on a reduction from the emptiness problem for two counter machines. Since Paz was focusing on expressiveness aspects of probabilistic automata rather than on algorithmic questions, his undecidability proof is spread on the whole book [11], which makes it arguably hard to read. The proof of Madani et al. is easier to read but quite long and technical.

In the next section, we present a new simple undecidability proof of the Emptiness Problem.

2.1 New proof of undecidability

In this section we show the undecidability of the (Strict) Emptiness Problem for the cut-point $\frac{1}{2}$ and for a restricted class of probabilistic automata called *simple* probabilistic automata:

Definition 2 (Simple automata). *A probabilistic automaton is called* simple *if every transition probability is in* $\left\{0, \frac{1}{2}, 1\right\}$.

The proof is based on a result of Bertoni [4]: the undecidability of the Equality Problem.

Problem 2 (Equality problem) *Given a simple probabilistic automaton* $\mathcal{A}$*, decide whether there exists a word* $w \in A^*$ *such that* $\mathbb{P}_{\mathcal{A}}(w) = \frac{1}{2}$.

Proposition 1 (Bertoni). *The equality problem is undecidable.*

The short and elegant proof of Bertoni is a reduction of the Post Correspondence Problem (PCP) to the Equality Problem.

Problem 3 (PCP) *Let* $\varphi_1 : A \to \{0,1\}^*$ *and* $\varphi_2 : A \to \{0,1\}^*$ *two functions, naturally extended to* A^**. Is there a word* $w \in A^*$ *such that* $\varphi_1(w) = \varphi_2(w)$*?*

Roughly speaking, the proof of Proposition 1 consists in encoding the equality of two words in the decimals of transition probabilities of a well-chosen probabilistic automaton. While the reduction of PCP to the Equality problem is relatively well-known, it may be less known that there exists a simple reduction of the Equality problem to the Emptiness and Strict Emptiness problems:

Proposition 2. *Given a simple probabilistic automaton* $\mathcal{A}$*, one can compute probabilistic automata* $\mathcal{B}$ *and* $\mathcal{C}$ *whose transition probabilities are multiple of* $\frac{1}{4}$ *and such that:*

$$\left(\exists w \in A^+, \mathbb{P}_{\mathcal{A}}(w) = \frac{1}{2}\right) \iff \left(\exists w \in A^+, \mathbb{P}_{\mathcal{B}}(w) \geq \frac{1}{4}\right) \tag{1}$$

$$\iff \left(\exists w \in A^+, \mathbb{P}_{\mathcal{C}}(w) > \frac{1}{8}\right) \ . \tag{2}$$

Proof. The construction of $\mathcal{B}$ such that (1) holds is based on a very simple fact: a real number x is equal to $\frac{1}{2}$ if and only if $x(1-x) \geq \frac{1}{4}$. Consider the automaton $\mathcal{B}$ which is the cartesian product of $\mathcal{A}$ with a copy of $\mathcal{A}$ whose accepting states are the non accepting states of $\mathcal{A}$. Then for every word $w \in A^*$, $\mathbb{P}_{\mathcal{A}_1}(w) = \mathbb{P}_{\mathcal{A}}(w)(1 - \mathbb{P}_{\mathcal{A}}(w))$, thus (1) holds.

The construction of $\mathcal{C}$ such that (2) holds is based on the following idea. Since $\mathcal{A}$ is simple, transition probabilities of $\mathcal{B}$ are multiples of $\frac{1}{4}$, thus for every word w of length $|w|$, $\mathbb{P}_{\mathcal{B}}(w)$ is a multiple of $\frac{1}{4^{|w|}}$. As a consequence, $\mathbb{P}_{\mathcal{B}}(w) \geq \frac{1}{4}$ if and only if $\mathbb{P}_{\mathcal{B}}(w) > \frac{1}{4} - \frac{1}{4^{|w|}}$. Adding three states to $\mathcal{B}$, one obtains easily a probabilistic automaton $\mathcal{C}$ such that for every non-empty word $w \in A^*$ and letter $a \in A$, $\mathbb{P}_{\mathcal{C}}(aw) = \frac{1}{2} \cdot \mathbb{P}_{\mathcal{B}}(w) + \frac{1}{2} \cdot \frac{1}{4^{|w|}}$, thus (2) holds. To build $\mathcal{C}$, simply add a new initial state that goes with equal probability $\frac{1}{2}$ either to the initial state of $\mathcal{B}$ or to a new accepting state q_f. From q_f, whatever letter is read, next state is q_f with probability $\frac{1}{4}$ and with probability $\frac{3}{4}$ it is a new non-accepting absorbing sink state q_*. □

As a consequence:

Theorem 1 (Paz). *The emptiness and the strict emptiness problems are undecidable for probabilistic automata. These problems are undecidable even for simple probabilistic automata and cut-point* $\lambda = \frac{1}{2}$.

To conclude this section, we present another connection between probabilistic and non-probabilistic automata on finite words.

Corollary 1. *The following problem is undecidable. Given a* non-deterministic *automaton on finite words, does there exists a word such that at least half of the computations on this word are accepting?*

We do not know a simple undecidability proof for this problem which does not make of use of probabilistic automata.

2.2 Probabilistic automata with few probabilistic transitions

Hirvensalo [8] showed that the emptiness problem is undecidable for probabilistic automata which have as few as 2 input letters and 25 states, see also [5] for similar result about the isolation problem.

On the other hand, the emptiness problem is decidable for *deterministic* automata. This holds whatever the number of states, as long as there are no probabilistic transition in the automaton. Formally, a probabilistic transition is a couple (s, a) of a state $s \in S$ and a letter $a \in A$ such that for at least one state $t \in S$, $0 < M_a(s, t) < 1$.

This motivates the following question: *what is the minimal number of probabilistic transitions for which the emptiness problem is undecidable?*

The following undecidability result is a rather surprising answer:

Proposition 3. *The emptiness problem is undecidable for probabilistic automata with two probabilistic transitions.*

Moreover, a slight variant of the emptiness problem for probabilistic automata with *one* probabilistic transition is undecidable:

Proposition 4. *The following problem is undecidable: given a simple probabilistic automaton over an alphabet A with one probabilistic transition and given a rational language of finite words $L \subseteq A^*$, decide whether $\mathbb{P}_{\mathcal{A}}(w) \geq \frac{1}{2}$ for some word $w \in L$.*

For probabilistic automata with a unique probabilistic transition, we do not know whether the emptiness problem is decidable or not.

3 Undecidability of the Value 1 problem

In his seminal paper about probabilistic automata [12], Rabin introduced the notion of *isolated cut-points*.

Definition 3. *A real number $0 \leq \lambda \leq 1$ is an isolated cut-point with respect to a probabilistic automaton $\mathcal{A}$ if:*

$$\exists \varepsilon > 0,\ \ \forall w \in A^*,\ \ |\mathbb{P}_{\mathcal{A}}(w) - \lambda| \geq \varepsilon\ \ .$$

Rabin motivates the introduction of this notion by the following theorem:

Theorem 2 (Rabin). *Let $\mathcal{A}$ a probabilistic automaton and $0 \leq \lambda \leq 1$ a cut-point. If λ is isolated then the language $\mathcal{L}_{\mathcal{A}}(\lambda) = \{u \in A^* \mid \mathbb{P}_{\mathcal{A}}(u) \geq \lambda\}$ is rational.*

This result suggests the following decision problem.

Problem 4 (Isolation Problem) *Given a probabilistic automaton $\mathcal{A}$ and a cut-point $0 \leq \lambda \leq 1$, decide whether λ is isolated with respect to $\mathcal{A}$.*

Bertoni [4] proved that the Isolation Problem is undecidable in general:

Theorem 3 (Bertoni). *The Isolation Problem is undecidable for probabilistic automata with five states.*

A closer look at the proof of Bertoni shows that the Isolation Problem is undecidable for a fixed λ, provided that $0 < \lambda < 1$.

However the same proof does not seem to be extendable to the cases $\lambda = 0$ and $\lambda = 1$. This was pointed out by Bertoni in the conclusion of [4]:

> "Is the following problem solvable: $\exists \delta > 0, \forall x, (p(x) > \delta)$? For automata with 1-symbol alphabet, there is a decision algorithm bound with the concept of transient state [11]. We believe it might be extended but have no proof for it".

The open question mentioned by Bertoni is the Isolation Problem for $\lambda = 0$. The case $\lambda = 1$ is essentially the same, since 0 is isolated in an automaton $\mathcal{A}$ if and only if 1 is isolated in the automaton obtained from $\mathcal{A}$ by turning final states to non-final states and vice-versa. When $\lambda = 1$, the Isolation Problem asks whether there exists some word accepted by the automaton with probability arbitrarily close to 1. We use the game-theoretic terminology and call this problem the Value 1 Problem.

The open question of Bertoni can be rephrased as the decidability of the following problem:

Problem 5 (Value 1 Problem) *Given a probabilistic automaton $\mathcal{A}$, decide whether $\mathcal{A}$ has value 1.*

Unfortunately,

Theorem 4. *The Value 1 Problem is undecidable.*

The proof of Theorem 4 is a reduction of the Strict Emptiness Problem to the Value 1 Problem. It is similar to the proof of undecidability of the Emptiness Problem for probabilistic Büchi automata of Baier et al. [2]. The core of the proof is the following proposition.

Proposition 5. *Let $0 < x < 1$ and $\mathcal{A}_x$ be the probabilistic automaton depicted on Fig. 1. Then $\mathcal{A}_x$ has value 1 if and only if $x > \frac{1}{2}$.*

The proof of Theorem 4 relies on the fact that there is a natural way to combine $\mathcal{A}_x$ with an arbitrary automaton $\mathcal{B}$ so that the resulting automaton has value 1 if and only if some word is accepted by $\mathcal{B}$ with probability strictly greater than $\frac{1}{2}$.

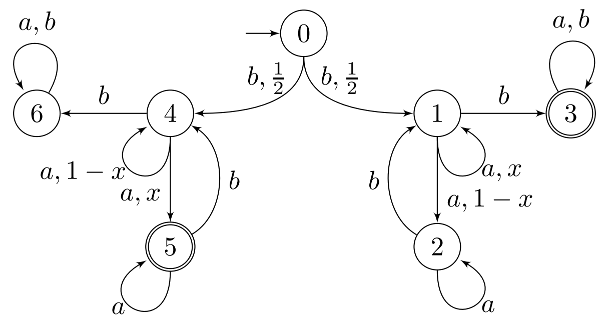

Fig. 1. This automaton has value 1 if and only if $x > \frac{1}{2}$.

The value 1 problem for simple probabilistic automata can be straigntforwardly rephrased as a "quantitative" decision problem about non-deterministic automaton on finite words, which shows that:

Corollary 2. *This decision problem is undecidable: given a* non-deterministic *automaton on finite words, does there exists words such that the proportion of of non-accepting computation pathes among all computation pathes is arbitrarily small?*

4 The class of $\sharp$-acyclic probabilistic automata

In this section, we introduce a new class of probabilistic automata, $\sharp$-*acyclic probabilistic automata*, for which the value 1 problem is decidable.

To get a decision algorithm for the value 1 problem, our starting point is the usual subset construction for non-deterministic automata, defined by mean of the natural action of letters on subsets of Q. However the quantitative aspect of the Value 1 Problem stressed in Corollary 2 suggests that the subset construction needs to be customized. Precisely, we use not only the usual action $S \cdot a$ of a letter a on a subset $S \subseteq Q$ of states but consider also another action $a^\sharp$. Roughly speaking, $a^\sharp$ deletes states that are transient when reading letter a forever.

Definition 4 (Actions of letters and $\sharp$-reachability). *Let $\mathcal{A}$ a probabilistic automaton with alphabet A and set of states Q. Given $S \subseteq Q$ and $a \in A$, we denote:*

$$S \cdot a = \{s \in Q \mid \exists s \in S, M_a(s,t) > 0\} \ .$$

A state $t \in Q$ is a-reachable from $s \in Q$ if for some $n \in \mathbb{N}$, $\mathbb{P}_\mathcal{A}(s \xrightarrow{a^n} t) > 0$. A state $s \in Q$ is a-recurrent if it is in a bottom strongly connected component of the graph of states and a-transitions i.e. if for any state $t \in Q$,

$$(t \text{ is } a\text{-reachable from } s) \implies (s \text{ is } a\text{-reachable from } t) \ .$$

A set $S \subseteq Q$ is a-stable if $S = S \cdot a$. If S is a-stable, we denote:

$$S \cdot a^\sharp = \{s \in S \mid s \text{ is } a\text{-recurrent}\} \ .$$

The support graph $\mathcal{G}_\mathcal{A}$ *of a probabilistic automaton $\mathcal{A}$ with alphabet A and set of states Q is the directed graph whose vertices are the non-empty subsets of Q and whose edges are the pairs (S,T) such that for some letter $a \in A$, either $(S \cdot a = T)$ or $(S \cdot a = S$ and $S \cdot a^\sharp = T)$.*

Reachability in the support graph of $\mathcal{A}$ is called $\sharp$-reachability in $\mathcal{A}$.

The class of $\sharp$-acyclic probabilistic automata is defined as follows.

Definition 5 ($\sharp$-acyclic probabilistic automata). *A probabilistic automaton is* $\sharp$-acyclic *if the only cycles in its support graph are self-loops.*

Obviously, this acyclicity condition is quite strong. However, it does not forbid the existence of cycles in the transition table, see for example the automaton depicted on Fig. 2. Note also that the class of $\sharp$-acyclic automata enjoys good properties: it is closed under cartesian product and parallel composition.

4.1 The value 1 problem is decidable for $\sharp$-acyclic automata

For $\sharp$-acyclic probabilistic automata, the value 1 problem is decidable:

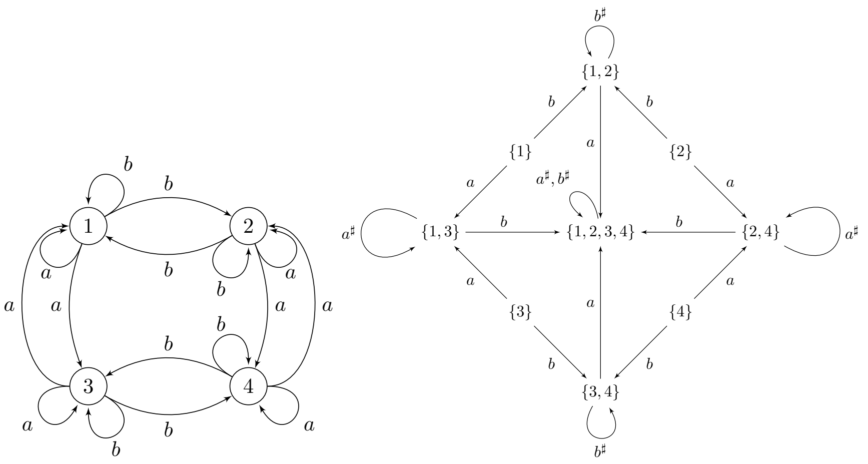

Fig. 2. A $\sharp$-acyclic automaton (on the left) and its support graph (on the right). All transition probabilities are equal to $\frac{1}{2}$.

Theorem 5. *Let $\mathcal{A}$ be a probabilistic automaton with initial state q_0 and final states F. Suppose that $\mathcal{A}$ is $\sharp$-acyclic . Then $\mathcal{A}$ has value 1 if and only if F is $\sharp$-reachable from $\{q_0\}$ in $\mathcal{A}$.*

The support graph can be computed on the fly in polynomial space thus deciding whether a probabilistic automaton is $\sharp$-acyclic and whether an $\sharp$-acyclic automaton has value 1 are PSPACE decision problems.

The rest of this section is dedicated to the proof of Theorem 5. This proof relies on the notion of limit-paths.

Definition 6 (Limit paths and limit-reachability). *Let $\mathcal{A}$ be a probabilistic automaton with states Q and alphabet A. Given two subsets S, T of Q, we say that T is limit-reachable from S in $\mathcal{A}$ if there exists a sequence $w_0, w_1, w_2, \ldots \in A^*$ of finite words such that for every state $s \in S$:*

$$\mathbb{P}_{\mathcal{A}}(s \xrightarrow{w_n} T) \xrightarrow[n \to \infty]{} 1 \ .$$

The sequence $w_0, w_1, w_2, \ldots$ is called a limit path *from S to T, and T is said to be limit-reachable from S in $\mathcal{A}$.*

In particular, an automaton has value 1 is and only if F is limit-reachable from $\{q_0\}$.

To prove Theorem 5, we show that in $\sharp$-acyclic automata, $\sharp$-reachability and limit-reachability coincide. The following proposition shows that $\sharp$-reachability always imply limit-reachability, may the automaton be $\sharp$-acyclic or not.

Proposition 6. *Let $\mathcal{A}$ be a probabilistic automaton with states Q and $S, T \subseteq Q$. If T is $\sharp$-reachable from S in $\mathcal{A}$ then T is limit-reachable from S in $\mathcal{A}$.*

The converse implication is not true in general. For example, consider the automaton depicted on Fig. 3. There is only one final state, state 3. The initial state is not represented, it leads with equal probability to states 1, 2 and 3. The transitions from states 1, 2 and 3 are either deterministic or have probability $\frac{1}{2}$.

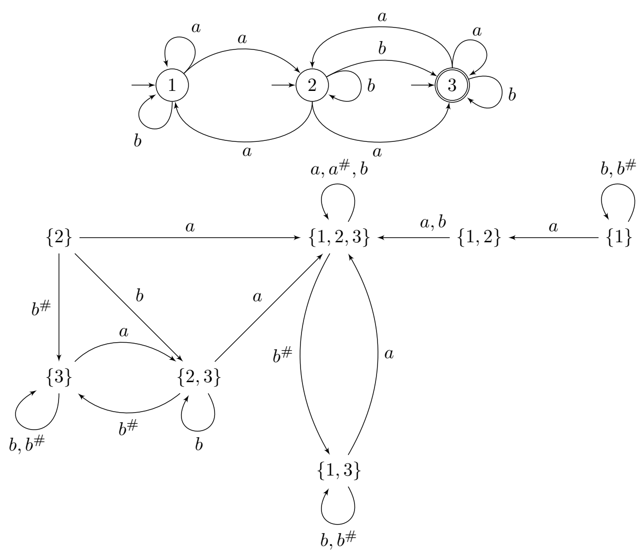

Fig. 3. This automaton has value 1 and is not $\sharp$-acyclic .

It turns out that the automaton on Fig. 3 has value 1, because $((b^n a)^n)_{n\in\mathbb{N}}$ is a limit-path from $\{1,2,3\}$ to $\{3\}$. However, $\{3\}$ is *not* reachable from $\{1,2,3\}$ in the support graph. Thus, limit-reachability does not imply $\sharp$-reachability in general. This automaton is *not* $\sharp$-acyclic , because his support graph contains a cycle of length 2 between $\{1,2,3\}$ and $\{1,3\}$. It is quite tempting to add an edge labelled $(ab^\sharp)^\sharp$ between $\{1,3\}$ and $\{3\}$.

Now we prove that for $\sharp$-acyclic automata, limit-reachability implies $\sharp$-reachability.

Definition 7 (Stability and $\sharp$-stability). *Let $\mathcal{A}$ be a probabilistic automaton with states Q. The automaton $\mathcal{A}$ is* stable *if for every letter $a \in A$, Q is a-stable. A stable automaton $\mathcal{A}$ is* $\sharp$-stable *if for every letter $a \in A$ $Q \cdot a^\sharp = Q$.*

The proof relies on the three following lemmatas.

Lemma 1 (Blowing lemma). *Let $\mathcal{A}$ be a $\sharp$-acyclic probabilistic automaton with states Q and $S \subseteq Q$. Suppose that $\mathcal{A}$ is $\sharp$-acyclic and $\sharp$-stable. If Q is limit-reachable from S in $\mathcal{A}$, then Q is $\sharp$-reachable from S as well.*

Proof (of the blowing lemma). If $S = Q$ there is nothing to prove. If $S \neq Q$, we prove that there exists $S_1 \subseteq Q$ such that (i) S_1 is $\sharp$-reachable from S, (ii)

$S \subsetneq S_1$, and (iii) Q is limit-reachable from S_1. Since $S \subsetneq Q$ and since there exists a limit-path from S to Q there exists at least one letter a such that S is not a-stable, i.e. $S \cdot a \not\subseteq S$. Since $\mathcal{A}$ is subset-acyclic, there exists $n \in \mathbb{N}$ such that $S \cdot a^{n+1} = S \cdot a^n$ i.e. $S \cdot a^n$ is a-stable. We choose $S_1 = (S \cdot a^n) \cdot a^\sharp$ and prove (i),(ii) and (iii) First, (i) is obvious.

To prove (ii), we prove that S_1 contains both S and $S \cdot a$. Let $s \in S$. By definition, every state t of $S \cdot a^n$ is a-accessible from s. Since $\mathcal{A}$ is $\sharp$-stable, state s is a-recurrent and by definition of a-recurrence, s is a-accessible from t. Since $t \in S \cdot a^n$ and $S \cdot a^n$ is a-stable, $s \in S \cdot a^n$ and since s is a-recurrent $s \in (S \cdot a^n) \cdot a^\sharp = S_1$. The proof that $S \cdot a \subseteq S_1$ is similar.

If $S_1 = Q$ the proof is complete, because (i) holds. If $S_1 \subsetneq Q$, then (iii) holds because $S \subseteq S_1$ thus Q is limit-reachable not only from S but from S_1 as well, using the same limit-path. As long as $S_n \neq Q$, we use (iii) to build inductively an increasing sequence $S \subsetneq S_1 \subsetneq S_2 \subsetneq \ldots \subsetneq S_n = Q$ such that for every $1 \leq k < n$, S_{k+1} is $\sharp$-reachable from S_k. Since $\sharp$-reachability is transitive this completes the proof of the blowing lemma. □

Next lemma shows that in a $\sharp$-stable and $\sharp$-acyclic automata, once a computation has flooded the whole state space, it cannot shrink back.

Lemma 2 (Flooding lemma). *Let $\mathcal{A}$ be a probabilistic automaton with states Q. Suppose that $\mathcal{A}$ is $\sharp$-acyclic and $\sharp$-stable. Then Q is the only set of states limit-reachable from Q in $\mathcal{A}$.*

Now, we turn our attention to leaves of the acyclic support graph.

Definition 8. *Let $\mathcal{A}$ be a probabilistic automaton with states Q. A non-empty subset $R \subseteq Q$ is called a leaf if for every letter $a \in A$, $R \cdot a = R$ and $R \cdot a^\sharp = R$.*

In a stable $\sharp$-acyclic automaton, there is a unique leaf:

Lemma 3 (Leaf lemma). *Let $\mathcal{A}$ be a probabilistic automaton with states Q. Suppose that $\mathcal{A}$ is $\sharp$-acyclic .Then there exists a unique leaf $\sharp$-accessible from Q. Every set limit-reachable from Q contains this leaf.*

To prove that limit-reachability implies $\sharp$-reachability, we proceed by induction on the depth in the support graph. The inductive step is:

Lemma 4 (Inductive step). *Let $\mathcal{A}$ be a probabilistic automaton with states Q and $S_0, T \subseteq Q$. Suppose that $\mathcal{A}$ is $\sharp$-acyclic and T is limit-reachable from S_0. Then either $S_0 = T$ or there exists $S_1 \neq S_0$ such that S_1 is $\sharp$-reachable from S_0 in $\mathcal{A}$ and T is limit-reachable from S_1 in $\mathcal{A}$.*

Repeated use of Lemma 4 gives:

Proposition 7. *Let $\mathcal{A}$ be a probabilistic automaton with states Q and $S_0, T \subseteq Q$. Suppose that $\mathcal{A}$ is $\sharp$-acyclic. If T is limit-reachable from S_0 in $\mathcal{A}$, then T is $\sharp$-reachable from S_0 as well.*

Thus, limit-reachability and $\sharp$-reachability coincide in $\sharp$-acyclic automata and Theorem 5 holds.

Is the maximal distance between two $\sharp$-reachable sets in the support graph bounded by a polynomial function of $|A|$ and $|Q|$? The answer to this question could lead to a simpler proof and/or algorithm.

Conclusion Whether the emptiness problem is decidable for probabilistic automata with a unique probabilistic transition is an open question.

The class of $\sharp$-acyclic automata can be probably extended to a larger class of automata for which the value 1 problem is still decidable.

Acknowledgments We thank the anonymous referees as well as A. Muscholl and I. Walukiewicz for finding a flaw in the previous version of this manuscript. We thank Benedikt Bollig for his valuable comments during the final preparation of this manuscript.

References

1. R. Alur, C. Courcoubetis, and M. Yannakakis. Distinguishing tests for nondeterministic and probabilistic machines. In *STOC*, pages 363–372. ACM, 1995.
2. Christel Baier, Nathalie Bertrand, and Marcus Größer. On decision problems for probabilistic büchi automata. In *FoSSaCS*, pages 287–301, 2008.
3. A. Bertoni, G. Mauri, and M. Torelli. Some recursive unsolvable problems relating to isolated cutpoints in probabilistic automata. In *Proceedings of ICALP*, pages 87–94, London, UK, 1977. Springer-Verlag.
4. Alberto Bertoni. The solution of problems relative to probabilistic automata in the frame of the formal languages theory. In *Proc. of the 4th GI Jahrestagung*, volume 26 of *LNCS*, pages 107–112. Springer, 1974.
5. Vincent Blondel and Vincent Canterini. Undecidable problems for probabilistic automata of fixed dimension. *Theory of Computing Systems*, 36:231–245, 2003.
6. R. G. Bukharaev. Probabilistic automata. *Journal of Mathematical Sciences*, 13(3):359–386, 1980.
7. R. Chadha, A. Prasad Sistla, and M. Viswanathan. Power of randomization in automata on infinite strings. In *CONCUR*, volume 5710 of *Lecture Notes in Computer Science*, pages 229–243. Springer, 2009.
8. Mika Hirvensalo. Improved undecidability results on the emptiness problem of probabilistic and quantum cut-point languages. In *SOFSEM '07*, pages 309–319, Berlin, Heidelberg, 2007. Springer-Verlag.
9. Dexter Kozen. Lower bounds for natural proofs systems. In *Proc. of 18th Symp. Foundations of Comp Sci.*, pages 254–266, 1977.
10. O. Madani, S. Hanks, and A. Condon. On the undecidability of probabilistic planning and related stochastic optimization problems. *Artificial Intelligence*, 147:5–34, 2003.
11. Azaria Paz. *Introduction to probabilistic automata (Computer science and applied mathematics)*. Academic Press, Inc., Orlando, FL, USA, 1971.
12. Michael O. Rabin. Probabilistic automata. *Information and Control*, 6(3):230–245, 1963.
13. Wen-Guey Tzeng. A polynomial-time algorithm for the equivalence of probabilistic automata. *SIAM J. Comput.*, 21(2):216–227, 1992.

Appendix

Proof (of Proposition 1). Given any instance $\varphi_1, \varphi_2 : A \to \{0,1\}^*$ of the PCP problem, we build an automaton $\mathcal{A}$ which accepts some word with probability $\frac{1}{2}$ if and only if PCP has a solution. Let $\psi : \{0,1\}^* \to [0,1]$ the mapping defined by:

$$\psi(a_0 \dots a_n) = \frac{a_n}{2} + \dots + \frac{a_0}{2^n} \ ,$$

and let $\theta_1 = \psi \circ \varphi_1$ and $\theta_2 = \psi \circ \varphi_2$. Without loss of generality, by insertion of 1's in words of the PCP instance, we can suppose that $\varphi_1, \varphi_2 : A \to \{10, 11\}^*$. That way, since Φ is injective on $1\{0,1\}^*$:

$$\forall w \in A^*, (\theta_1(w) = \theta_2(w)) \iff (\varphi_1(w) = \varphi_2(w)) \ . \tag{3}$$

Let $\mathcal{A}_1 = (Q, A, M, q_0^1, q_F^1)$ the probabilistic automaton with two states $Q = \{q_0^1, q_F^1\}$ and transitions:

$$\forall a \in A, M(a) = \begin{bmatrix} 1 - \theta_1(a) & \theta_1(a) \\ 1 - \theta_1(a) - 2^{-|\varphi_1(a)|} & \theta(a) + 2^{-|\varphi_1(a)|} \end{bmatrix} \ .$$

A simple computation shows that:

$$\forall w \in A^*, \ \mathbb{P}_{\mathcal{A}_1}(w) = \theta_1(w) \ . \tag{4}$$

A very similar construction produces a two-states automaton $\mathcal{A}_2$ such that:

$$\forall w \in A^*, \ \mathbb{P}_{\mathcal{A}_2}(w) = 1 - \theta_2(w) \ . \tag{5}$$

Let $\mathcal{A}$ be the disjoint union of these two automata $\mathcal{A}_1$ and $\mathcal{A}_2$ plus a new initial state that leads with equal probability $\frac{1}{2}$ to one of the initial states q_0^1 and q_0^2 of $\mathcal{A}_1$ and $\mathcal{A}_2$. Then for every word $w \in A^*$ and every letter $a \in A$,

$$\begin{aligned}\left(\exists w \in A^*, \ \mathbb{P}_{\mathcal{A}}(aw) = \frac{1}{2}\right) &\iff \left(\exists w \in A^*, \ \frac{1}{2}\mathbb{P}_{\mathcal{A}_1}(w) + \frac{1}{2}\mathbb{P}_{\mathcal{A}_2}(w) = \frac{1}{2}\right) \\ &\iff (\exists w \in A^*, \ \theta_1(w) = \theta_2(w)) \\ &\iff (\exists w \in A^*, \ \varphi_1(w) = \varphi_2(w)) \\ &\iff \ \text{PCP has a solution,}\end{aligned}$$

where the first equivalence is by definition of $\mathcal{A}$, the second is by (4) and (5), the third holds by (3) and the fourth is by definition of PCP. This completes the proof of Proposition 1. □

Proof (of Theorem 1). According to Proposition 1 and Proposition 2 the emptiness and the strict emptiness problems are undecidable for cut-point $\frac{1}{2}$ and automata whose transition probabilities are multiples of $\frac{1}{8}$. The transformation of such automata into simple automata is easy.

5 Proof of Proposition 4

Proof. To prove Proposition 4, we provide an algorithm which takes as input a simple probabilistic automaton $\mathcal{A}$ on an alphabet A (with an arbitrary number of probabilistic transitions) and outputs two objects:

- a simple automaton $\mathcal{A}'$ over an alphabet A' with only one probabilistic transition and
- $|A|$ words $\{u_a, a \in A\}$ on the alphabet A', each word u_a of length $3|Q|$,

such that:

$$\left(\exists w \in \mathcal{A}^*, \mathbb{P}_{\mathcal{A}}(w) \geq \frac{1}{2}\right) \iff \left(\exists w \in \{u_a, a \in A\}^*, \mathbb{P}_{\mathcal{A}'}(w') \geq \frac{1}{2}\right) \quad . \tag{6}$$

According to Theorem 1, the Emptiness Problem for simple probabilistic automata and cut-point $\frac{1}{2}$ is undecidable, hence (6) proves Proposition 4.

The alphabet A', the automaton $\mathcal{A}'$ and the words $(u_a)_{a \in A}$ are defined as follows. If $\mathcal{A} = (Q, A, \mathcal{M}, q_0, F)$ then $\mathcal{A}' = (Q', A', \mathcal{M}', q_0, F)$ is defined by:

1. The alphabet A' is made of a new letter a_* plus, for each letter $a \in A$ and state $s \in Q$, two new letters $\alpha(a, s)$ and $\beta(a, s)$ so that:
$$A' = \{a_*\} \cup \bigcup_{a \in A, s \in Q} \{\alpha(a, s), \beta(a, s)\} \quad .$$
2. The set of states of $\mathcal{A}'$ is obtained from Q by addition of three new states s_*, s_0, s_1, so that:
$$Q' = Q \cup \{s_*, s_0, s_1\} \quad .$$
3. The initial state q_0 and the set of final states F are left unchanged.
4. The transitions of $\mathcal{A}'$ are as follows. For every letter $a \in A$ and state $s \in Q$, the new letter $\alpha(a, s)$ has no effect on states $u \neq s$ (i.e. $M_{\alpha(a,s)}(u) = u$), while from state s the transition is deterministic to state s_* i.e. $M_{\alpha(a,s)}(s) = s_*$.
5. The new letter a_* has no effect on states $u \neq s_*$, while from state s_* this is the only probabilistic transition of $\mathcal{A}'$, defined by $M_{a_*}(s_*) = \frac{1}{2}s_0 + \frac{1}{2}s_1$.
6. For every letter $a \in A$ and state $s \in Q$, the new letter $\beta(a, s)$ has no effect on states $u \notin \{s_0, s_1\}$. Transitions on letter $\beta(a, s)$ from states s_0 and s_1 are deterministic and depend on $M_a(s)$. If the transition $M_a(s)$ is deterministic, i.e. if $M_a(s, r) = 1$ for some state r then $M_{\beta(a,s)}(s_0) = r$ and $M_{\beta(a,s)}(s_1) = r$. If the transition $M_a(s)$ is probabilistic i.e. if $M_a(s) = \frac{1}{2}r + \frac{1}{2}r'$ for some states r, r' then $M_{\beta(a,s)}(s_0) = r$ and $M_{\beta(a,s)}(s_1) = r'$.

Now we define the words $u_a, a \in A$. Choose some enumeration $\{s_0, s_1, \ldots, s_n\} = Q$ of states of $\mathcal{A}$ and for each letter a, define the word

$$u_a = \alpha(a, s_0)a_*\beta(a, s_0)\alpha(a, s_1)a_*\beta(a, s_1) \cdots \alpha(a, s_n)a_*\beta(a, s_n) \in A'^* \quad .$$

Then (6) holds because for every word $a_0a_1a_2 \cdots \in A^*$,

$$\mathbb{P}_{\mathcal{A}}(a_0a_1a_2 \cdots) = \mathbb{P}_{\mathcal{A}'}(u_{a_0}u_{a_1}u_{a_2} \ldots) \quad .$$

Since automaton $\mathcal{A}'$ has only one probabilistic transition, this completes the proof of Proposition 4. □

6 Proof of Proposition 3

Proof. The undecidable problem described in Proposition 4 reduces to the emptiness problem for simple probabilistic automata with two probabilistic transitions: given $\mathcal{A}$ and L, add a new initial state to $\mathcal{A}$ and from this new initial state, proceed with probability $\frac{1}{2}$ either to the original initial state of $\mathcal{A}$ to the initial state of a deterministic automaton that checks whether the input word is in L. This new automaton accepts a word with probability more than $\frac{3}{4}$ if and only if the original automaton accepts a word with probability more than $\frac{1}{2}$. □

7 Proof of Proposition 5

Proof (of Proposition 5). We shall prove:

$$\left(x > \frac{1}{2}\right) \iff (\forall \varepsilon > 0, \exists w \in A^*, \mathbb{P}_{\mathcal{A}_x}(w) \geq 1 - \varepsilon) \quad . \tag{7}$$

In order to prove this equivalence we notice that: $\mathbb{P}_{\mathcal{A}_x}(1 \xrightarrow{a^n b} 3) = x^n$ and $\mathbb{P}_{\mathcal{A}_x}(4 \xrightarrow{a^n b} 6) = (1-x)^n$. Let $(n_k)_{k\in\mathbb{N}}$ an increasing sequence of integers. By reading the word $w = a^{n_0} b a^{n_1} b \ldots a^{n_i} b$, we get:

$$\begin{cases} \mathbb{P}_{\mathcal{A}_x}(1 \xrightarrow{w} 3) = 1 - \prod_{k\geq 0} \left(1 - x^{n_k}\right) \\ \mathbb{P}_{\mathcal{A}_x}(4 \xrightarrow{w} 6) = (1-x)^{n_1} + (1-(1-x)^{n_1})(1-x)^{n_2} + \ldots \\ \qquad\qquad\qquad = 1 - \prod_{k\geq 0}(1-(1-x)^{n_k}) \leq \sum_{k\geq 0}(1-x)^{n_k} \end{cases} \tag{8}$$

If $x \leq \frac{1}{2}$ then $\mathbb{P}_{\mathcal{A}_x}(1 \xrightarrow{w} 3) \leq \mathbb{P}_{\mathcal{A}_x}(4 \xrightarrow{w} 6)$ therefore if $x \leq \frac{1}{2}$ no word w can be accepted with probability greater than $\frac{1}{2}$, in particular the automaton does not have value 1. This proves the converse implication in (7). Assume that $x > \frac{1}{2}$ and let $\varepsilon > 0$, we exhibit an increasing sequence of integers $(n_k)_{k\in\mathbb{N}}$ such that we have:

$$\begin{cases} \sum_{k\geq 0} x^{n_k} = \infty \\ \sum_{k\geq 0}(1-x)^{n_k} \leq \varepsilon \end{cases} \tag{9}$$

Let $C \in \mathbb{R}$ and $n_k = \ln_x(\frac{1}{k}) + C$, notice that $\sum_{k\geq 0}(x)^{n_k} = x^C . \sum_{k\geq 0} \frac{1}{k} = \infty$. In the other hand we have:

$$\begin{aligned} 1 - x &= x^{\ln_x(1-x)} \\ &= x^{\frac{\ln(1-x)}{\ln x}} \end{aligned}$$

There exists $\beta > 1$ such that: $1-x = x^\beta$, hence $\sum_{k\geq 0}(1-x)^{n_k} = \sum_{k\geq 0} x^{\beta n_k}$. So: $\sum_{k\geq 0} x^{\beta n_k} = x^{\beta C} \sum_{k\geq 0} x^{\beta \ln_x(\frac{1}{k})} = x^{\beta C} \sum_{k\geq 0} \frac{1}{k^\beta}$. Since this series converges, we satisfy (7) by choosing a suitable constant c.

Suppose now that (7) is satisfied for some sequence $n_0, n_1, n_2 \ldots$. Then according to (8), for every $i \in \mathbb{N}$ the word $a^{n_0}ba^{n_1}b \ldots a^{n_i}b$ is accepted by $\mathcal{A}_x$ with probability

$$\frac{1}{2}\left(1 - \prod_{0 \leq k \leq i}(1 - x^{n_k})\right) + \frac{1}{2}\left(\prod_{0 \leq k \leq i}(1 - (1-x)^{n_k})\right) .$$

According to , when i goes to ∞, the left operand converges to $\frac{1}{2}$ and the right operand is asymptotically larger than $\frac{1}{2}(1-\varepsilon)$. This proves the implication in (7). $\square$

8 Proof of Theorem 4

Proof (of Theorem 4). Given a probabilistic automaton $\mathcal{B}$ with alphabet A such that $a, b \notin \mathcal{B}$, we combine $\mathcal{B}$ and the automaton $\mathcal{A}_x$ on Fig.1 to obtain an automaton $\mathcal{C}$ which has value 1 if and only if there exists a word w such that $\mathbb{P}_{\mathcal{A}}(w) > \frac{1}{2}$. The input alphabet of $\mathcal{C}$ is $A \cup \{b\}$ plus a new letter $\sharp$. $\mathcal{C}$ is computed as follows. First, the transitions in $\mathcal{A}_x$ on letter a are deleted. Second, we make two copies $\mathcal{A}_4$ and $\mathcal{A}_1$ of the automaton $\mathcal{B}$, such that the initial state of $\mathcal{A}_4$ is 4 and the initial state of $\mathcal{A}_1$ is 1. From states of $\mathcal{A}_4$ and $\mathcal{A}_1$ other than the initial states, reading letter b leads to the sink state 6. Third, from a state s of $\mathcal{A}_4$ the transition on the new letter $\sharp$ is deterministic and leads to 5 if s is a final state and to 4 if s is not a final state. Fourth, from a state s of $\mathcal{A}_1$ the transition on the new letter $\sharp$ is deterministic and leads to 1 if s is a final state and to 2 if s is not a final state. Fifth, the final states of $\mathcal{C}$ are 5 and 3. Sixth, states $0, 3, 6, 5$ and 2 are absorbing for letters in A.

Then suppose there exists w such that $\mathbb{P}_{\mathcal{A}}(w) > \frac{1}{2}$ and let us show that $\mathcal{C}$ has value 1. Let $\epsilon > 0$ and let $u_\epsilon = ba^{i_0}ba^{i_1}ba^{i_2}b \cdots a^{i_k}$ be a word accepted by $\mathcal{B}$ with probability $1 - \epsilon$. Then by construction of $\mathcal{C}$,

$$\mathbb{P}_{\mathcal{C}}(b(w\sharp)^{i_0}b(w\sharp)^{i_1}b(w\sharp)^{i_2}b \cdots (w\sharp)^{i_k}) \geq \mathbb{P}_{\mathcal{A}}(u_\epsilon) \geq 1 - \epsilon,$$

thus $\mathcal{C}$ has value 1.

Now suppose that for every $w \in A^*, \mathbb{P}_{\mathcal{A}}w \leq \frac{1}{2}$ and let us show that $\mathcal{C}$ has not value 1. Let $w' \in (A \cup \{b, \sharp\})^*$. Factorize w' in $w' = u_0v_0\sharp u_1v_1\sharp u_kv_k \cdots$ such that $u_i \in b^*$ and $v_i \in A^*$. Then by construction of $\mathcal{C}$ and by hypothesis, $\mathbb{P}_{\mathcal{C}}(w') \leq \mathbb{P}_{\mathcal{A}_{\frac{1}{2}}}(u_0au_1au_2a \cdots u_ka) \leq \mathrm{val}(\mathcal{A}_{\frac{1}{2}})$. Thus $\mathrm{val}(\mathcal{C}) \leq \mathrm{val}(\mathcal{A}_{\frac{1}{2}})$ and according to Proposition 5, $\mathrm{val}(\mathcal{C}) < 1$. $\square$

9 Proof of Theorem 5

Proof (of Proposition 6). Proposition 6 is a consequence of the two following facts.

First, if there is an edge from S to T in the support graph of $\mathcal{A}$, then T is limit reachable from S: let $S, T \subseteq Q$ and $a \in A$. If $S \cdot a = T$, then the sequence constant equal to a is a limit path from S to T. If $S \cdot a = S$ and $S \cdot a^\sharp = T$ then by definition of $S \cdot a^\sharp$, $(a^n)_{n\in\mathbb{N}}$ is a limit path from S to T.

Second, limit-reachability is a transitive relation: let $S_0, S_1, S_2 \subseteq Q$ such that S_1 is limit-reachable from S_0 and S_2 is limit-reachable from S_1. Let $(u_n)_{n\in\mathbb{N}}$ a limit-path from S_0 to S_1 and $(v_n)_{n\in\mathbb{N}}$ a limit-path from S_1 to S_2. Then $(u_n v_n)_{n\in\mathbb{N}}$ is a limit-path from S_0 to S_2. □

Proof (of the leaf lemma). Since $\mathcal{A}$ is $\sharp$-acyclic , there exists at least one leaf S which is $\sharp$-reachable from Q.

We prove that every set limit-reachable from Q contains the leaf S. Let R limit-reachable from Q and $(u_n)_{n\in\mathbb{N}}$ a limit-path from Q to R. Since S is a leaf then for every $a \in A$, S is a-stable, hence $(u_n)_{n\in\mathbb{N}}$ is a fortiori a limit-path from S to $R \cap S$. Since S is a leaf, the restriction $\mathcal{A}[S]$ of the automaton $\mathcal{A}$ to S is $\sharp$-stable. According to the flooding lemma applied to $\mathcal{A}[S]$, $S = R \cap S$, thus $S \subseteq R$. Since $\sharp$-reachability imply limit-reachability, this implies unicity of the leaf reachable from Q. □

Proof (of the Flooding lemma). Let $(u_n)_{n\in\mathbb{N}}$ be a limit path from Q to some set of states $T \subsetneq Q$. We shall prove that $T = Q$.

Let $A_T = \{a \in A \mid T \cdot a = T\}$.

First, we prove that for every letter $a \in A_T$, $Q \setminus T$ is a-stable. Otherwise there would be $s \in Q \setminus T$, $a \in A_T$ and $t \in T$ such that t is a-reachable from s. Since $\mathcal{A}$ is $\sharp$-stable, s and t are both a-recurrent, and by definition of a-recurrence, since t is a-reachable from s, s would be a-reachable from t as well. But $s \in Q \setminus T$ and $t \in T$, which contradicts the a-stability of T.

Second, we prove that $u_n \in A_T^*$ for only finitely many $n \in \mathbb{N}$. Since for every $a \in A_T$, $Q \setminus T$ is a-stable, then during the computation $\delta_Q = \delta_0, \delta_1, \ldots, \delta_{|u_n|}$ on the word u_n, $\sum_{s \in Q\setminus T} \delta_k(s)$ is constant. Thus, for every $n \in \mathbb{N}$,

$$\mathbb{P}_{\mathcal{A}}(s \xrightarrow{u_n} T) = \sum_{s\in Q\setminus T} (\delta_Q \cdot u_n)(s) = \sum_{s \in Q\setminus T} \delta_Q(s) = \frac{|Q| - |T|}{|Q|} > 0.$$

Since $(u_n)_{n\in\mathbb{N}}$ is a limit-path from Q to T, $\mathbb{P}_{\mathcal{A}}(s \xrightarrow{u_n} T)$ converges to 0 hence the inequality can hold only for finitely many $n \in \mathbb{N}$.

Now we show that there exists $T_1 \subseteq Q$ such that:

(i) $T_1 \neq T$,
(ii) T is $\sharp$-reachable from T_1 in $\mathcal{A}$,
(iii) and T_1 is limit-reachable from Q in $\mathcal{A}$.

Since any infinite subsequence of a limit-path is a limit-path, and since we proved that $u_n \in A_T^*$ for only finitely many $n \in \mathbb{N}$, we can assume w.l.o.g. that for every $n \in \mathbb{N}$, $u_n \notin A_T^*$. Thus for every $n \in \mathbb{N}$, there exists $v_n \in A^*$, $a_n \in A \setminus A_T$ and $w_n \in A_T^*$ such that $u_n = v_n a_n w_n$. W.l.o.g. again, since A is finite and $\mathcal{D}(Q)$ is compact, we can assume that $(a_n)_{n\in\mathbb{N}}$ is constant equal to a letter $a \in A \setminus A_T$ and that $(\delta_Q \cdot v_n)_{n\in\mathbb{N}}$ converges to a probability distribution $\delta \in \mathcal{D}(Q)$.

The choice of T_1 such that (i), (ii) and (iii) hold depends on $\mathrm{Supp}(\delta) \cdot a$.

If $\mathrm{Supp}(\delta) \cdot a = T$ then we choose $T_1 = \mathrm{Supp}(\delta)$. Then (i) holds because $a \notin A_T$, (ii) holds because $T = T_1 \cdot a$ and (iii) holds because $(v_n)_{n\in\mathbb{N}}$ is a limit-path from Q to T_1.

If $\mathrm{Supp}(\delta)\cdot a \neq T$ then we choose $T_1 = \mathrm{Supp}(\delta)\cdot a$. Then (i) clearly holds and (iii) holds because $(v_n a)_{n\in\mathbb{N}}$ is a limit path from Q to T_1 in $\mathcal{A}$. To prove that (ii) holds, consider the restriction $\mathcal{A}[T, A_T]$ of automaton $\mathcal{A}$ to states T and alphabet A_T. Then $(w_n)_{n\in\mathbb{N}}$ is a limit-path from T_1 to T in $\mathcal{A}[T, A_T]$. Moreover, since $\mathcal{A}$ is $\sharp$-acyclic and $\sharp$-stable, $\mathcal{A}[T, A_T]$ also is. Thus, we can apply the blowing lemma to $\mathcal{A}[T, A_T]$ and T_1, which proves that T is $\sharp$-reachable from T_1 in $\mathcal{A}[T, A_T]$, thus in $\mathcal{A}$ as well.

If $T_1 = Q$, the proof is complete. Otherwise, as long as $T_n \neq Q$, we use condition (iii) to build inductively a sequence $T = T_0, T_1, T_2, \cdots T_n$ such that for every $0 \leq k < n$, $T_k \neq T_{k+1}$ (condition (i))and T_k is $\sharp$-reachable from T_{k+1} in $\mathcal{A}$ (condition (ii)). Since $\mathcal{A}$ is $\sharp$-acyclic , $T_n = Q$ after at most 2^Q inductive steps.

Since $\sharp$-reachability is transitive, this proves that T is $\sharp$-reachable from Q. Since $\mathcal{A}$ is $\sharp$-stable, the only set $\sharp$-reachable from Q is Q thus $T = Q$, which completes the proof of the Flooding lemma. □

Proof (of Lemma 4). Let $(u_n)_{n\in\mathbb{N}}$ be a limit-path from S_0 to T. Let $A_0 = \{a \in A \mid S_0 \cdot a = S_0\}$. For every $n \in \mathbb{N}$, let v_n be the longuest prefix of u_n in A_0^*. Since every infinite subsequence of a limit-path is a limit-path, and since $\mathcal{D}(Q)$ is compact, we can suppose without loss of generality that $(\delta_{S_0} \cdot v_n)_{n\in\mathbb{N}}$ converges to some distribution $\delta \in \mathcal{D}(Q)$.

Suppose first that $\mathrm{Supp}(\delta) = S_0$. If $u_n \in A_0^*$ for infinitely many $n \in \mathbb{N}$ then $T = S_0$. Otherwise, since A is finite we can suppose w.l.o.g. that there exists a letter $a \in A \setminus A_0$ such that for every $n \in \mathbb{N}$, $v_n a$ is a prefix of u_n. Let also w_n such that $u_n = v_n a w_n$. Let $S_1 = S_0 \cdot a$. Then $S_1 \neq S_0$ because $a \notin A_0$ and S_1 is clearly $\sharp$-reachable from S_0. Moreover $(w_n)_{n\in\mathbb{N}}$ is a limit-path from S_1 to T, this completes the proof.

Suppose now that $\mathrm{Supp}(\delta) \neq S_0$. Let $\mathcal{A}[S_0, A_0]$ the probabilistic automaton obtained from $\mathcal{A}$ by restriction to the alphabet A_0 and to the state space S_0. By definition of A_0, $\mathcal{A}[S_0, A_0]$ is stable and it is $\sharp$-acyclic because $\mathcal{A}$ is. According to the leaf lemma, $\mathcal{A}[S_0, A_0]$ has a unique leaf. Let S_1 be this unique leaf. Since $\mathrm{Supp}(\delta)$ is limit-reachable from S_0 in $\mathcal{A}[S_0, A_0]$, according to the leaf lemma again, $S_1 \subseteq \mathrm{Supp}(\delta)$ hence $S_1 \neq S_0$. Moreover, since it is the unique leaf, S_1 is $\sharp$-reachable from S_0 in $\mathcal{A}[S_0, A_0]$ hence in $\mathcal{A}$ as well. For every $n \in \mathbb{N}$, let w_n such that $u_n = v_n w_n$. Then $(w_n)_{n\in\mathbb{N}}$ is a limit-path from S_1 to T. This completes the proof. □

Proof (of Proposition 7). Apply again and again Lemma 4 to build a sequence $S_0, S_1, S_2, \ldots$ such that for every k, $S_k \neq S_{k+1}$, S_{k+1} is $\sharp$-reachable from S_k and T is limit-reachable from S_{k+1}. As long as $S_k \neq T$, Lemma 4 is used to build S_{k+1}. Since $\mathcal{A}$ is subset-acyclic, the sequence has length at most 2^Q thus for some $k, S_k = T$. Since $\sharp$-reachability is transitive, this proves that T is $\sharp$-reachable from S_0. □