2012 27th Annual ACM/IEEE Symposium on Logic in Computer Science

Decidable Problems for Probabilistic Automata on Infinite Words

Krishnendu Chatterjee
IST Austria
Klosterneuburg, Austria

Mathieu Tracol
IST Austria
Klosterneuburg, Austria

***Abstract*—We consider probabilistic automata on infinite words with acceptance defined by parity conditions. We consider three qualitative decision problems: (i) the *positive* decision problem asks whether there is a word that is accepted with positive probability; (ii) the *almost* decision problem asks whether there is a word that is accepted with probability 1; and (iii) the *limit* decision problem asks whether words are accepted with probability arbitrarily close to 1. We unify and generalize several decidability results for probabilistic automata over infinite words, and identify a *robust* (closed under union and intersection) subclass of probabilistic automata for which all the qualitative decision problems are decidable for parity conditions. We also show that if the input words are restricted to lasso shape (regular) words, then the positive and almost problems are decidable for all probabilistic automata with parity conditions. For most decidable problems we show an optimal PSPACE-complete complexity bound.**

***Index Terms*—Automata and formal languages, Probabilistic automata, Parity conditions, Positive, almost and limit decision problems.**

I. INTRODUCTION

Probabilistic automata. The class of probabilistic automata for finite words was introduced in the seminal work of Rabin [18] as an extension of classical finite automata. Probabilistic automata on finite words have been extensively studied (see the book [17] on probabilistic automata and the survey of [6]). Probabilistic automata on infinite words have been studied recently in the context of verification and analysis of reactive systems [2], [1], [7], [8]. We consider probabilistic automata on infinite words with acceptance defined by safety, reachability, Büchi, coBüchi, and parity conditions, as they can express all commonly used specifications (like safety, liveness, fairness) of verification.

Qualitative decision problems. We consider three *qualitative* decision problems for probabilistic automata on infinite words [1], [15]: given a probabilistic automaton with an acceptance condition, (i) the *positive* decision problem asks whether there is a word that is accepted with positive probability (probability > 0); (ii) the *almost-sure* (almost for short in sequel) decision problem asks whether there is a word that is accepted almost-surely (with probability 1); and (iii) the *limit-sure* (limit for short in sequel) decision problem asks whether for every $\epsilon > 0$ there is a word that is accepted with probability at least $1-\epsilon$. The qualitative decision problems for probabilistic automata are the generalization of the emptiness and universality problems for deterministic automata.

Decidability and undecidability results. The decision problems for probabilistic automata on finite words have been extensively studied [17], [6], and the main results establish the undecidability of the quantitative version of the decision problems (where the thresholds are a rational $0 < \lambda < 1$, rather than 0 and 1). The undecidability results for the qualitative decision problems for probabilistic automata on infinite words are quite recent. The results of [1] show that the positive (resp. almost) decision problem is undecidable for probabilistic automata with Büchi (resp. coBüchi) acceptance condition, and as a corollary both the positive and almost decision problems are undecidable for parity acceptance conditions (as both Büchi and coBüchi conditions are special cases of parity conditions). The results of [1] also show that the positive (resp. almost) decision problem is decidable for probabilistic automata with coBüchi (resp. Büchi) acceptance condition, and these results have been extended to the more general case of stochastic games with imperfect information in [3] and [9]. The positive and almost problems are decidable for safety and reachability conditions, and also for probabilistic automata over finite words. For all the decidable almost and positive problems for probabilistic automata PSPACE-complete bounds were established in [8], [7]. It was shown in [15] that the limit decision problem is undecidable even for probabilistic finite automata, and the proof can be easily adapted to show that the limit decision problem is undecidable for reachability, Büchi, coBüchi and parity conditions (see [10] for details).

Decidable subclasses. The root cause of the undecidability results is that for arbitrary probabilistic automata and arbitrary input words the resulting probabilistic process is complicated. As a consequence several researchers have focused on identifying subclasses of probabilistic automata where the qualitative decision problems are decidable. The work of [7] presents a subclass of probabilistic automata, namely hierarchical probabilistic automata (HPA), and show that the positive and almost problems are decidable for Büchi and coBüchi conditions on HPAs. The work of [15] presents a subclass of probabilistic automata, namely $\#$-acyclic automata, and show that the limit reachability problem is decidable for this class of automata over finite words. The two subclasses HPA and $\#$-acyclic automata are incomparable in expressive power.

Our contributions. In this work we unify and generalize several decidability results for probabilistic automata over

1043-6871/12 $26.00 © 2012 IEEE
DOI 10.1109/LICS.2012.29

infinite words, and identify a robust subclass of probabilistic automata for which all the qualitative decision problems are decidable for parity acceptance conditions. For the first time, we study the problem of restricting the structure of input words, as compared to the probabilistic automata, and show that if the input words are restricted to *lasso shape* words, then the positive and almost problems are decidable for all probabilistic automata with parity acceptance conditions. The details of our contributions are as follows.

1) We first present a very general result that would be the basic foundation of the decidability results. We introduce a notion of *simple* probabilistic process: the non-homogeneous Markov chain induced on the state space of a probabilistic automaton by an infinite word is simple if the tail σ-field of the process has a particular structure. The structure of the tail σ-field is derived from Blackwell-Freedman-Cohn-Sonin *decomposition-separation theorem* [5], [12], [19] on finite non-homogeneous Markov chains which generalizes the classical results on homogeneous Markov chains.
2) We then show that if we restrict the input words of a probabilistic automaton to those which induce simple processes, then the positive and almost decision problems are decidable for parity conditions. We establish that these problems are PSPACE-complete.
3) We study for the first time the effect of restricting the structure of input words for probabilistic automata, rather than restricting the structure of probabilistic automata. We show that for all ultimately periodic (regular or lasso shape) words and for all probabilistic automata, the probabilistic process induced is a simple one. Hence as a corollary of our first result, we obtain that if we restrict to lasso shape words, then the positive and almost decision problems are decidable (PSPACE-complete) for all probabilistic automata with parity conditions. However, the limit decision problem for the reachability condition is still undecidable for lasso shape words, as well as for the Büchi and coBüchi conditions.
4) We then introduce the class of *simple probabilistic automata* (for short simple automata): a probabilistic automaton is simple if every input infinite words induce simple processes on its state space. This semantic definition of simple automata uses the decomposition-separation theorem. We present a *structural (or syntactic)* subclass of the class of simple automata, called *structurally simple automata*, which relies on the structure of the *support graph* of the automata (the support graph is obtained via subset constructions of the automata). We show that the class of structurally simple automata generalizes both the models of HPA and #-acyclic automata. Since HPA generalizes deterministic automata, it follows that structurally simple automata with parity conditions strictly generalizes ω-regular languages (since every deterministic parity automaton is a HPA with a parity condition, and structurally simple automata strictly subsumes HPA). We show that for structurally simple automata with parity conditions, the positive and almost problems are PSPACE-complete, and the limit problem can be decided in EXPSPACE. Thus our results both unify and generalize two different results for decidability of subclasses of probabilistic automata. Moreover, we show that structurally simple automata are *robust*, i.e., closed under union and intersection. Thus we are able to identify a robust subclass of probabilistic automata for which all the qualitative decision problems are decidable for parity conditions. From our structural characterization it also follows that given a probabilistic automaton, it can be decided in EXPSPACE whether the automaton is structurally simple.

In this paper we use deep results from probability theory (decomposition-separation theorem) to establish general results about the decidability of problems on probabilistic automata. To the best of our knowledge, the decomposition-separation theorem has never been used in the context of probabilistic automata to establish decidability results. We present a sufficient structural condition to ensure that a probabilistic automaton is simple. Note that the notion of a probabilistic automaton being simple is a semantic notion (of the induced probabilistic process being simple) from probability theory. Thus our structural characterization captures the semantic probabilistic notion in the context of probabilistic automata. The detailed proofs omitted due to space restrictions are available as an Arxiv technical report [11]. A very related work on probabilistic automata will also appear [14]. The work of [14] focuses on probabilistic automata on finite words and the limit decision (value 1) problem, and uses novel algebraic techniques (Simon's factorization forest theorem) to identify a new sub-class of probabilistic automata where the limit decision problem is decidable. An interesting direction of future work would be investigate to unify and generalize our work and the results of [14].

II. Preliminaries

Distributions. Given a finite set Q, we denote by $\Delta(Q)$ the set of probability distributions on Q. Given $\alpha \in \Delta(Q)$, we denote by $\mathrm{Supp}(\alpha)$ the support of α, i.e. $\mathrm{Supp}(\alpha) = \{q \in Q \mid \alpha(q) > 0\}$.

Words and prefixes. Let Σ be a finite *alphabet* of letters. A *word* w is a finite or infinite sequence of letters from Σ, i.e., $w \in \Sigma^*$ or $w \in \Sigma^\omega$. Given a word $w = a_1, a_2 ... \in \Sigma^\omega$ and $i \in \mathbb{N}$, we define $w(i) = a_i$, and we denote by $w[1..i] = a_1, ..., a_i$ the prefix of length i of w. Given $j \geq i$, we denote by $w[i..j] = a_i, ..., a_j$ the subword of w from index i to j. An infinite word $w \in \Sigma^\omega$ is a *lasso shape word* if there exist two finite words ρ_1 and ρ_2 in Σ^* such that $w = \rho_1 \cdot \rho_2^\omega$.

Definition 1 (Finite Probabilistic Table (see [17]))**.** *A* Finite Probabilistic Table *(FPT) is a tuple $\mathcal{T} = (Q, \Sigma, \{M_a\}_{a\in\Sigma}, \alpha)$ where Q is a finite set of states, Σ is a finite alphabet, α is an initial distribution on Q, and the M_a, for $a \in \Sigma$, are Markov matrices of size $|Q|$, i.e., for all $q, q' \in Q$ we have $M_a(q, q') \geq 0$ and for all $q \in Q$ we have $\sum_{q'\in Q} M_a(q, q') = 1$.*

Distribution generated by words. For a letter $a \in \Sigma$, let $\delta(q,a)(q') = M_a(q,q')$ denote the transition probability from q to q' given the input letter a. Given $\beta \in \Delta(Q)$, $q \in Q$ and $\rho \in \Sigma^*$, let $\delta(\beta,\rho)(q)$ be the probability, starting from a state sampled accordingly to β and reading the input word ρ, to go to state q. Formally, given $\rho = a_1, ..., a_n \in \Sigma^*$, let $M_\rho = M_{a_1} \cdot M_{a_2} \cdot ... \cdot M_{a_n}$. Then $\delta(\beta,\rho)(q) = \sum_{q' \in Q} \beta(q') \cdot M_\rho(q', q)$. We often write $\delta(\beta,\rho)$ instead of $\mathrm{Supp}(\delta(\beta,\rho))$, for simplicity: $\delta(\beta,\rho)$ is the set of states reachable with positive probability when starting from distribution β and reading ρ. As well, given $H \subseteq Q$, we write $\delta(H,\rho)$ for the the set of states reachable with positive probability when starting from a state in H sampled uniformly at random, and reading ρ.

Homogeneous and non-homogeneous Markov chains. A Markov chain is a sequence of random variables $X_0, X_1, X_2, ...$, taking values in a (finite) set Q, with the Markov property: $\mathbb{P}(X_{n+1} = x | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \mathbb{P}(X_{n+1} = x | X_n = x_n)$. Given $n \in \mathbb{N}$, the matrix M_n of size $|Q|$ such that for all $q, q' \in Q$ we have $M_n(q,q') = \mathbb{P}(X_{n+1} = q' | X_n = q)$ is the *transition matrix at time* n of the chain. The Markov chain is *homogeneous* if M_n does not depend on n. In the general case, we call the chain *non-homogeneous*.

Induced Markov chains. Given a FPT with state space Q, given $G \subseteq Q$ and $\rho = a_0, ..., a_{m-1} \in \Sigma^*$ such that $\delta(G,\rho) \subseteq G$, we define the Markov chain $\{X_n\}_{n \in \mathbb{N}}$ induced by (G,ρ) as follows: the initial distribution, i.e. the distribution of X_0, is uniform on G; given $i \in \mathbb{N}$, X_{i+1} is distributed according to $\delta(X_i, a_{i \bmod m})(-)$. Intuitively, $\{X_n\}_{n \in \mathbb{N}}$ is the Markov chain induced on the FPT when reading the word ρ^ω.

Probability space and σ-field. A word $w \in \Sigma^\omega$ induces a probability space $(\Omega, \mathcal{F}, \mathbb{P}^w)$: $\Omega = Q^\omega$ is the set of *runs*, $\mathcal{F}$ is the σ-field generated by cones of the type $C_\rho = \{r \in Q^\omega \mid r[1..|\rho|] = \rho\}$ where $\rho \in Q^*$, and $\mathbb{P}^w$ is the associated probability distribution on Ω. See [21] for the standard results on this topic. We write $\{X_n^w\}_{n \in \mathbb{N}}$ for the *non-homogeneous* Markov chain induced on Q by w, and given $n \in \mathbb{N}$ let μ_n^w be the distribution of X_n^w on Q:

$$\text{Given } q \in Q, \quad \mu_n^w(q) = \mathbb{P}^w[\{r \in \Omega \mid r(n) = q\}]$$

The σ-field $\mathcal{F}$ is also the smallest σ-field on Ω with respect to which all the X_n^w, $n \in \mathbb{N}$, are measurable. For all $n \in \mathbb{N}$, let $\mathcal{F}_n = \mathcal{B}(X_n^w, X_{n+1}^w, ...)$ be the smallest σ-field on Ω with respect to which all the X_i^w, $i \geq n$, are measurable. We define $\mathcal{F}_\infty = \bigcap_{n \in \mathbb{N}} \mathcal{F}_n$, called the *tail σ-field of* $\{X_n^w\}$. Intuitively, an event Γ is in $\mathcal{F}_\infty$ if changing a finite number of states of a run r does not affect the occurrence of the run r in Γ.

Atomic events. Given a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and $\Gamma \in \mathcal{F}$, we say that Γ is $\mathcal{F}$*-atomic* if $\mathbb{P}(\Gamma) > 0$, and for all $\Gamma' \in \mathcal{F}$ such that $\Gamma' \subseteq \Gamma$ we have either $\mathbb{P}(\Gamma') = 0$ or $\mathbb{P}(\Gamma') = \mathbb{P}(\Gamma)$. In this paper we will use atomic events in relation to the tail σ-field of Markov chains.

Acceptance conditions. Given a FPT, let $F \subseteq Q$ be a set of accepting (or target) states. Given a run r, we denote by $\mathrm{Inf}(r)$ the set of states that appear infinitely often in r. We consider the following acceptance conditions.

1) *Safety condition.* The safety condition $\mathrm{Safe}(F)$ defines the set of paths that only visit states in F; i.e., $\mathrm{Safe}(F) = \{ (q_0, q_1, \ldots) \mid \forall i \geq 0.\ q_i \in F \}$.
2) *Reachability condition.* The reachability condition $\mathrm{Reach}(F)$ defines the set of paths that visit states in F at least once; i.e., $\mathrm{Reach}(F) = \{ (q_0, q_1, \ldots) \mid \exists i \geq 0.\ q_i \in F \}$.
3) *Büchi condition.* The Büchi condition $\text{Büchi}(F)$ defines the set of paths that visit states in F infinitely often; i.e., $\text{Büchi}(F) = \{ r \mid \mathrm{Inf}(r) \cap F \neq \emptyset \}$.
4) *coBüchi condition.* The coBüchi condition $\text{coBüchi}(F)$ defines the set of paths that visit states outside F finitely often; i.e., $\text{coBüchi}(F) = \{ r \mid \mathrm{Inf}(r) \subseteq F \}$.
5) *Parity condition.* The parity condition consists of a priority function $p : Q \to \mathbb{N}$ and defines the set of paths such that the minimum priority visited infinitely often is even, i.e., $\mathrm{Parity}(p) = \{ r \mid \min(p(\mathrm{Inf}(r))) \text{ is even} \}$. Büchi and coBüchi conditions are special cases of parity conditions with two priorities (priority set $\{ 0, 1 \}$ for Büchi and $\{ 1, 2 \}$ for coBüchi).

Probabilistic automata. A *Probabilistic Automaton* (PA) is a tuple $\mathcal{A} = (\mathcal{T}, \Phi)$ where $\mathcal{T}$ is a FPT and Φ is an acceptance condition.

Decision problems. Let $\mathcal{A}$ be a PA with acceptance condition $\Phi : \Omega \to \{0,1\}$. We consider the following decision problems.

1) *Almost problem:* Whether there exists $w \in \Sigma^\omega$ such that $\mathbb{P}_\mathcal{A}^w(\Phi) = 1$.
2) *Positive problem:* Whether there exists $w \in \Sigma^\omega$ such that $\mathbb{P}_\mathcal{A}^w(\Phi) > 0$.
3) *Limit problem:* Whether for all $\epsilon > 0$, there exists $w \in \Sigma^\omega$ such that $\mathbb{P}_\mathcal{A}^w(\Phi) > 1 - \epsilon$. The limit problem is also known as the value 1 problem.

Proposition 1 summarizes the known results from [1], [10], [15], [8], [7].

Proposition 1. *Given a PA with an acceptance condition Φ, the following assertions hold:*

1) *The almost problem is decidable (PSPACE-complete) for $\Phi =$ safety, reachability, Büchi, and undecidable for $\Phi =$ co-Büchi and parity.*
2) *The positive problem is decidable (PSPACE-complete) for $\Phi =$ safety, reachability, co-Büchi, and undecidable for $\Phi =$ Büchi and parity.*
3) *The limit problem is decidable (PSPACE-complete) for $\Phi =$ safety, and undecidable for $\Phi =$ reachability, Büchi, co-Büchi, and parity.*

III. Simple Processes

In this section we first recall the decomposition-separation theorem, then use it to decompose the tail σ-field of stochastic processes into atomic events. We then introduce the notion of simple processes, which are stochastic processes where the atomic events obtained using the decomposition-separation theorem are *non-communicating*.

A. The Decomposition Separation Theorem and tail σ-fields

The structure of the tail σ-field of a general non-homogeneous Markov chain has been deeply studied by mathematicians. Blackwell and Freedman, in [5], presented a generalization of the classical *decomposition theorem* for homogeneous Markov chains, in the context of non-homogeneous Markov chains with finite state spaces. The work of Blackwell and Freedman has been deepened by Cohn [12] and Sonin [19], who gave a more complete picture. We present the results of [5], [12], [19] in the framework of *jet decompositions* presented in [19].

Jets and partition into jets. A *jet* is a sequence $J = \{J_i\}_{i\in\mathbb{N}}$, where each $J_i \subseteq Q$. A tuple of jets $(J^0, J^1, ..., J^c)$ is called a *partition of Q^ω into jets* if for every $n \in \mathbb{N}$, we have that $J^0_n, J^1_n, ..., J^c_n$ is a partition of Q. The *Decomposition-Separation Theorem*, in short *DS-Theorem*, proved by Cohn [12] and Sonin [19] using results of [5], is given in Theorem 1. We will use mainly the decomposition part of the DS-theorem. We first define the notion of *mixing* property of jets.

Mixing property of jets. Given a FPT $\mathcal{A}$, a jet $J = \{J_i\}_{i\in\mathbb{N}}$ is *mixing* for a word w if: given $X^w_n, n \geq 0$ the process induced on Q by w, given $q, q' \in Q$, and a sequence of states $\{q_i\}_{i\in\mathbb{N}}$ such that for all $i \geq 0$ we have $q_i \in J_i$, given $m \in \mathbb{N}$, if $\lim_n \mathbb{P}^w[X^w_n = q_n \mid X^w_m = q] > 0$ and $\lim_n \mathbb{P}^w[X^w_n = q_n \mid X^w_m = q'] > 0$, then we have:

$$\lim_{n\to\infty} \frac{\mathbb{P}^w[X^w_n = q_n \mid X^w_n \in J^k_n \wedge X^w_m = q]}{\mathbb{P}^w[X^w_n = q_n \mid X^w_n \in J^k_n \wedge X^w_m = q']} = 1$$

Intuitively, a jet is mixing if the probability distribution of a state of the process, conditioned to the fact that this state belongs to the jet, is ultimately independent of the initial state. This extends the notion of mixing process on homogeneous ergodic Markov chains, on which the distribution of a state of the process after a number of steps is close to the stationary distribution, irrespective of the initial state.

Theorem 1 (The Decomposition-Separation (DS) Theorem [5], [12], [19]). *Given a FPT $\mathcal{A} = (Q, \Sigma, \{M_a\}_{a\in\Sigma}, \alpha)$, for all $w \in \Sigma^\omega$ there exists $c \in \{1, 2, \ldots, |Q|\}$ and a partition $(J^0, J^1, ..., J^c)$ of Q^ω into jets such that:*

1) *With probability one, after a finite number of steps, a run $r \in \Omega$ enters into one of the jets J^k, $k \in \{1, 2, \ldots, c\}$ and stays there forever.*
2) *For all $k \in \{1, 2, \ldots, c\}$ the jet J^k is mixing.*

Theorem 1 holds even if Σ is infinite: it is valid for any non-homogeneous Markov chain on a finite state space. In this paper we will focus on finite alphabets only. Also the DS theorem is more general (see [19], page 3), and we only used the decomposition part of the theorem (which is used in proof of Proposition 2). In fact we mostly use the result from [4].

Remark. We note that for all $i \in \{1, 2, \ldots, c\}$, either $\mu^w_n(J^i_n) \to_{n\to\infty} 0$ or there exists $\lambda_i > 0$ such that for n large enough $\mu^w_n(J^i_n) > \lambda_i$. Indeed, if $\mu^w_n(J^i_n) \not\to_{n\to\infty} 0$ but there exists a subsequence of $\{\mu^w_n(J^i_n)\}_{n\in\mathbb{N}}$ which goes to zero, then a non zero probability of runs enter J^i_n and leave it afterward infinitely often, which contradicts the first point of Theorem 1. Thus, we can always assume that there exists $\lambda > 0$ such that for all $i \in \{1, 2, \ldots, c\}$, for n large enough, we have $\mu^w_n(J^i_n) > \lambda$. If this is not the case, we just merge the jets J^i such that $\mu^w_n(J^i_n) \to_{n\to\infty} 0$ with J^0, which does not invalidate the properties of the jet decomposition stated by Theorem 1.

For the following of the section, we fix $w \in \Sigma^\omega$ and a partition $J^0, J^1, ..., J^c$ of Q^ω as in the DS Theorem. Given $i \in \{1, 2, \ldots, c\}$ and $n \in \mathbb{N}$, let:

$$\tau^i_n = \{r \in \Omega \mid r(i) \in J^i_n\}, \quad \text{and} \quad \tau^i_\infty = \cup_{N\in\mathbb{N}} \cap_{n\geq N} \tau^i_n$$

We now present a result directly from our formulation of the DS Theorem (the result can also be proved using more general results of [12]).

Proposition 2. *For all $i \in \{1, 2, \ldots, c\}$, the following assertions hold: (1) $\tau^i_\infty \in \mathcal{F}_\infty$, i.e., τ^∞_i is a tail σ-field event; (2) τ^i_∞ is $\mathcal{F}_\infty$-atomic; i.e., τ^∞_i is an atomic tail event; and (3) $\mathbb{P}^w(\bigcup_{i=1}^c \tau^i_\infty) = 1$.*

The fact that the τ^i_∞ are atomic sets of $\mathcal{F}_\infty$ means that all the runs which belong to the same τ^i_∞ will satisfy the same *tail* properties. Intuitively, a tail property only depends on the asymptotic behaviour of the process. Several important classes of properties are tail properties, as presented in [13]: in particular any parity condition is a tail property.

B. Simple processes and characterization with jets

Definition 2. *Let $\{X^w_n\}_{n\in\mathbb{N}}$ be a process induced on Q by a word $w \in \Sigma^\omega$, and let μ^w_n be its probability distribution on Q at time n. We say that $\{\mu^w_n\}_{n\in\mathbb{N}}$ is* simple *if there exist $\lambda > 0$ and two sequences $\{A_n\}_{n\in\mathbb{N}}$ and $\{B_n\}_{n\in\mathbb{N}}$ of subsets of Q such that:*

- *$\forall n \in \mathbb{N}$, A_n, B_n is a partition of Q*
- *$\forall n \in \mathbb{N}$, $\forall q \in A_n$, $\mu^w_n(q) > \lambda$*
- *$\mu^w_n(B_n) \to_{n\to\infty} 0$*

The second point of the following proposition shows that the tail σ-field of a simple process can be decomposed as a set of "non-communicating" jets. Intuitively, a jet is non-communicating if there exists a bound $N \in \mathbb{N}$ such that after time N, if a run belongs to the jet, it will stay in it for ever with probability one. The following proposition is a reformulation of the notion of simple process in the framework of jets decomposition.

Proposition 3. *Let $w \in \Sigma^\omega$, and suppose that the process $\{\mu^w_n\}_{n\in\mathbb{N}}$ induced on Q is simple. Then there exists a decomposition of Q^ω into jets, $J^0, J^1, ..., J^c$, and $N \in \mathbb{N}$, which satisfy the following properties:*

1) *For all $n \geq N$, all $i \in \{1, 2, \ldots, c\}$ and all $q \in J^i_n$, we have $\mu_n(q) > \lambda$.*
2) *For all $i \in \{1, 2, \ldots, c\}$ and all $n_2 > n_1 \geq N$ we have $\delta(J^i_{n_1}, w^{n_2}_{n_1+1}) \subseteq J^i_{n_2}$.*
3) *$\mu^w_n(J^0_n) \to_{n\to\infty} 0$.*

4) Each jet $J^i,\ i \in \{1, 2, \ldots, c\}$ is mixing.

Remark about simple processes. We now remark on the intuitive and natural interpretation of simple processes. First, in Proposition 3 we have already established the characterization of simple processes through elegant notion of jet decomposition and mixing property (which uses the very well studied notion of communicating and non-communicating processes). Second, the notion of simple processes has another natural interpretation in terms of convergence of probabilities which is as follows: for a simple process, if the probability to be at a state s is bounded away from zero (above a threshold), then a state t reachable from s in the process also has probability bounded away from zero. In other words, simple processes rule out complicated convergence behaviour of probabilistic processes. We also note that in the deep undecidability results of [1], the processes that arise in the analysis are not simple.

IV. Decidable Problems for Simple Processes and Lasso shape Words

In this section we will present decidable algorithms (with optimal complexity) for the decision problems with the restriction of simple processes, and for lasso shape words.

A. Decidable problems for simple processes

We first define the *simple decision problems* that impose the simple process restriction. Given an acceptance condition Φ, we consider the following problems:

1) *Simple almost (resp. positive) problems:* Does there exist $w \in \Sigma^\omega$ such that $\{\mu_n^w\}_{n\in\mathbb{N}}$ is simple and $\mathbb{P}_{\mathcal{A}}^w(\Phi) = 1$ (resp. $\mathbb{P}_{\mathcal{A}}^w(\Phi) > 0$)?
2) *Simple limit problem:* For all $\epsilon > 0$, is there $w \in \Sigma^\omega$ such that $\{\mu_n^w\}_{n\in\mathbb{N}}$ is simple and $\mathbb{P}_{\mathcal{A}}^w(\Phi) > 1 - \epsilon$?

Proposition 4 shows that the decidability and undecidability results of Proposition 1 concerning the positive, almost, and limit safety and reachability problems still hold when we consider their "simple process" version. Proposition 5 and Proposition 6 are more interesting as they show that the almost and positive parity problem become decidable when restricted to simple processes. Finally, Proposition 7 shows that the "limit" decision problems remain undecidable even when restricted to simple processes.

Proposition 4. *The simple almost (resp. positive) safety and reachability problems are PSPACE-complete, as well as the simple limit safety problem. The simple limit reachability problem is undecidable.*

Proposition 5. *The simple almost parity problem is PSPACE-complete.*

Proof: The proof is in three parts: first we present an equivalent formulation of the problem. Then we show that the equivalent formulation gives a problem which we can solve in PSPACE. Finally we give the PSPACE lower bound.

Equivalent formulation. In the following, $p : Q \to \mathbb{N}$ is a parity function on Q, and $\Phi = \mathrm{Parity}(p)$. We prove that: **(1)** There exists $w \in \Sigma^\omega$ such that the induced process is simple and $\mathbb{P}_{\mathcal{A}}^w(\Phi) = 1$ if and only if **(2)** There exists $G \subseteq Q$ and $\rho_1, \rho_2 \in \Sigma^*$ such that $G = \delta(\alpha, \rho_1)$, $\delta(G, \rho_2) \subseteq G$, and the runs on the Markov chain induced by (G, ρ_2) satisfy Φ with probability one. We then show that the properties can be verified in PSPACE and also present a PSPACE lower bound.

We show the equivalence **(2)**$\Leftrightarrow$**(1)**. The way **(2)**$\Rightarrow$**(1)** is direct, since we will show in Section IV-B that the process induced by a lasso shape word on any automaton is always simple. We prove that **(1)**$\Rightarrow$**(2)**. Let $w = a_1, ..., a_i, ...$ be such that the induced process is simple and $\mathbb{P}_{\mathcal{A}}^w(\Phi) = 1$. Using Proposition 3, let $J^0, J^1, ..., J^m$ be the decomposition of Q^ω into jets and let $N_0 \in \mathbb{N},\ \lambda > 0$ be such that:

- $\forall n \geq N_0$, $\forall i \in \{1, 2, \ldots, c\}$, $\forall q \in J_n^i$: $\mu_n(q) > \lambda$.
- $\forall i \in \{1, 2, \ldots, c\}$, for all $n_2 > n_1 \geq N_0$, we have $\delta(J_{n_1}^i, w_{n_1+1}^{n_2}) \subseteq J_{n_2}^i$.
- $\mu_n^w(J_n^0) \to_{n\to\infty} 0$
- Each jet $J^i,\ i \in \{1, 2, \ldots, c\}$ is mixing.

Without loss on generality, since Q is finite, taking N_0 large enough, we can assume that the vector of sets of states $(J_{N_0}^0, ..., J_{N_0}^c)$ appears infinitely often in the sequence $\{(J_n^0, ..., J_n^c)\}_{n\in\mathbb{N}}$. As well, without loss on generality, we can assume that for all $n \geq N_0$ and all $i \in \{1, 2, \ldots, c\}$, all the states in J_n^i appear infinitely often among the sets J_m^i, for $m \geq N_0$. Let $i \in \{1, 2, \ldots, c\}$. Given $q \in Q$, let

$$\Phi_q = \{r \in \Omega \mid q \in \mathrm{Inf}(r) \text{ and } p(q) = \min_{q' \in \mathrm{Inf}(r)} p(q')\}$$

Clearly, for all $q \in Q$, $\Phi_q \in \mathcal{F}_\infty$. Since Q is finite, there exists $q_i \in Q$ such that $\mathbb{P}(\tau_\infty^i \cap \Phi_{q_i}) > 0$. By Proposition 2, τ_∞^i is atomic, hence $\tau_\infty^i \subseteq \Phi_{q_i}$. Since the runs of the process satisfy the parity condition with probability one, $p(q_i)$ must be even. Moreover, for all $n \geq N_0$ and all $q \in J_n^i$, we must have $p(q) \geq p(q_i)$. Indeed, such a q appears an infinite number of times in the sequence J_n^i, by hypothesis, and always with probability at least λ.

Since $\tau_\infty^i \subseteq \Phi_{q_i}$, there exists $m_i \in \mathbb{N}$ such that for all $q \in J_i^{N_0}$, there exists $m < m_i$ such that $\delta(q, w[N_0+1..m])(q_i) > 0$. We define $m = \max_{i\in\{1,2,\ldots,c\}}\ m_i$, and $m' \geq m$ such that

$$(J_{N_0}^0, ..., J_{N_0}^c) = (J_{N_0+m'}^0, ..., J_{N_0+m'}^c)$$

Taking $\rho_1 = w[0..N_0]$ and $\rho_2 = w[N_0+1..N_0+m']$ completes the proof. Indeed, when starting from the initial distribution, after reading ρ_1, we arrive by construction in one of the sets $J_{N_0}^i$, with $i \in \{0, \ldots, c\}$. Starting from this state q, if the word ρ_2 is taken as input, we go to set $J_{N_0+m'}^i$ with probability one, visit q_i with positive probability, and do not visit any state with probability smaller that $p(q_i)$. This implies that when starting from q and reading ρ_2^ω, we visit q_i with probability one, hence the result.

Now, we argue the PSPACE upper and lower bounds.

PSPACE upper bound. First, we show that we can verify the second property in NPSPACE, hence in PSPACE. The proof is in two steps. In a first step, we show that we can decide in NPSPACE whether, given $G \subseteq Q$, there exists $\rho_1 \in \Sigma^*$

such that $G = \delta(\alpha, \rho_1)$. For this notice that, given $G \subseteq Q$, if there exists $\rho_1 \in \Sigma^*$ such that $G = \delta(\alpha, \rho_1)$, then there exists $\rho_1' \in \Sigma^*$ such that $G = \delta(\alpha, \rho_1')$ and $|\rho_1'| \leq 2^{|Q|}$. Thus, we can restrict the search to words ρ_1 of length at most $2^{|Q|}$. By guessing the letters $a_1, a_2, \dots$ of ρ_1 one by one, and by keeping in memory the set $A_i = \delta(\alpha, a_1, \dots, a_i)$ at each step, we can check at each step whether $A_i = G$, and thus we can decide whether there exists such a ρ_1 in NPSPACE.

In a second step, we show that, given $G \subseteq Q$, we can decide in NPSPACE whether there exists $\rho_2 \in \Sigma^*$ such that the runs on the periodic non-homogeneous Markov chain induced by (G, ρ_2) satisfy Φ with probability one. For this, we refine the previous argument. Notice that this is equivalent to find $\rho_2 = a_1, \dots, a_k \in \Sigma^*$ and $A, B \subseteq Q$ such that:

- ρ_2 has length at most $2^{2 \cdot |Q|}$
- $\delta(G, \rho_2) \subseteq G$
- A, B partition G
- A is the set of recurrent states for the homogeneous Markov chain induced by ρ_2 on G
- B is the set of transient states for the homogeneous Markov chain induced by ρ_2 on G
- For all $q_0 \in A$, for all the finite runs $q_0, a_1, q_1, a_2, q_2, \dots, a_k$ generated with positive probability when initiated on q and when reading ρ_2, the minimal value of the $p(q_i),\ i \in \{\, 0, k-1 \,\}$ is even.

This can be checked in NPSPACE. Indeed, we can guess A, B, and the letters of ρ_2 one by one, and at each step keep in memory the following sets:

- The set of states visited at time i, i.e. $E_i = \delta(A \cup B, a_1, \dots, a_i)$
- For all $q \in A$ and all $q' \in \delta(q, a_1, \dots, a_i)$, the minimal p value of the paths visited between q and q'. Notice that this set has size at most $|Q|$.
- For all $q \in A \cup B$ and all $q' \in \delta(A \cup B, a_1, \dots, a_i)$, a boolean value $v_i(q, q')$ which is equal to one if there exists a path between q and q' between the first step and step i, and which is null if not.

At the end, we just have to check that $E_k = G$, that the minimal p-values of all the paths issued from A is even, that the set of states in A are recurrent for the chain, and that the states in B are transient. This can be done easily since we can recover the graph of the Markov chain on G from the values given by $v_{|\rho_2|}$.

PSPACE lower bound. We prove now that the simple almost Büchi problem is PSPACE-hard. For this, we reduce the problem of checking the emptiness of a finite intersection of regular languages, which is known to be PSPACE complete by [16], to the *simple almost Büchi problem*, which is a particular case of the simple almost parity problem. The size of the input of Problem 1 is the sum of the number of states of the automata.

Problem 1 (Finite Intersection of Regular Languages)**.**
Input: $\mathcal{A}_1, ..., \mathcal{A}_l$ a family of regular deterministic automata (on finite words) on the same finite alphabet Σ.
Question: Do we have $\mathcal{L}(\mathcal{A}_1) \cap ... \cap \mathcal{L}(\mathcal{A}_l) = \emptyset$?

Let $\mathcal{A}_1, ..., \mathcal{A}_l$ be a family of regular automata on the same finite alphabet Σ, with respective state spaces Q_i and transition functions δ_i (where $\delta_i(s, a)(t) = 1$ if there exist a transition from s to t with label $a \in \Sigma$ in $\mathcal{A}_i$). We build a probabilistic automaton $\mathcal{A} = (Q, \Sigma', \delta, \alpha, F)$ such that the simple almost Büchi(F) problem is satisfied on $\mathcal{A}$ iff $\mathcal{L}(\mathcal{A}_1) \cap ... \cap \mathcal{L}(\mathcal{A}_l) \neq \emptyset$.

Let x be a new letter, not in Σ, and let $\Sigma' = \Sigma \cup \{x\}$.

- Q is the union of the state spaces of the $\mathcal{A}_i$, plus two extra states s and $\perp$. That is $Q = \bigcup_{i=1}^{l} Q_i' \cup \{s, \perp\}$, where the Q_i' are disjoint copies of the Q_i.
- The state $\perp$ is a sink: for all $a \in \Sigma'$, $\delta(\perp, a)(\perp) = 1$.
- If u' is the copy of a non accepting state u of $\mathcal{A}_i$, we allow in $\mathcal{A}$ the same transitions from u' as in $\mathcal{A}_i$ for u: if $a \in \Sigma$, let $\delta(u', a)(v') = 1$ iff v' is the copy of a state $v \in Q_i$ such that $\delta_i(u, a)(v) = 1$. Moreover we add a transition from u with label x: $\delta(u, x)(\perp) = 1$.
- If u' is the copy of an accepting state u of $\mathcal{A}_i, i \in [1; l]$, the transitions from u' in $\mathcal{A}$ are the same as in $\mathcal{A}_i$, plus an extra transition $\delta(u', x)(s) = 1$.
- From state s in $\mathcal{A}$, with uniform probability on $i \in [1; l]$, when reading x, the system goes to one of the copies of an initial state of the $\mathcal{A}_i$'s.
- For the transitions which have not been precised, for instance if $a \in \Sigma$ is read in state s, the system goes with probability one to the sink $\perp$.
- The initial distribution α is the Dirac distribution on s.
- $F = \{s\}$

Given $\rho \in \mathcal{L}(\mathcal{A}_1) \cap ... \cap \mathcal{L}(\mathcal{A}_l)$, the input word $(x \cdot \rho \cdot x)^\omega$ satisfies clearly the simple almost Büchi(F) problem since a run visits s after each occurrence of $x \cdot \rho \cdot x$ (the generated process is simple since we see in Section IV-B that any process generated on a probabilistic automaton by a lasso shape word is simple).

Conversely, suppose that there exists $\rho \in \Sigma^\omega$ such that the induced process is simple and satisfies almost surely the Büchi(F) condition.

- Since the only transition from s which does not goes to the sink has label x, the word ρ must start with letter x.
- Since with probability one the runs induced by ρ visit infinitely often s, the letter x must appear infinitely often in ρ. Let $\rho = x \cdot \rho' \cdot x$ where $\rho' \in \Sigma$ is non empty and does not contain the letter x. After reading $x \cdot \rho' \cdot x$, since the process cannot be in the sink $\perp$ with positive probability, it has to be on s with probability one. This implies that $\rho' \in \mathcal{L}(\mathcal{A}_1) \cap ... \cap \mathcal{L}(\mathcal{A}_l)$, hence $\mathcal{L}(\mathcal{A}_1) \cap ... \cap \mathcal{L}(\mathcal{A}_l) \neq \emptyset$.

This concludes the proof of the PSPACE completeness of our problem. ■

Proposition 6. *The simple positive parity problem is PSPACE-complete.*

A corollary of the proofs of Proposition 5 and Proposition 6 is that if the simple almost (resp. positive) parity problem is satisfied by a word, then there is also a witness lasso shape word that satisfies the almost (resp. positive) parity problem.

Proposition 7. *The simple limit Büchi and coBüchi problems are undecidable.*

From the propositions of this section we obtain the following theorem. In the theorem below the PSPACE-completeness of the limit safety problem follows as for safety conditions the limit and almost problem coincides.

Theorem 2. *The simple almost and positive problems are PSPACE-complete for parity conditions. The simple limit problem is PSPACE-complete for safety conditions, and the simple limit problem is undecidable for reachability, Büchi, coBüchi and parity conditions.*

B. Decidable problems for lasso shape words

In this sub-section we consider the decision problems where, instead of restricting the probabilistic automata, we restrict the set of input words to lasso shape words. The set of lasso shape words is quite natural, as they represent regular words, and has often been used as counter-examples in many problems related to verification. This provides the motivation to study probabilistic automata with lasso shape (or periodic or regular) words as input. First, the processes induced by such words are simple:

Proposition 8. *Let $\mathcal{A}$ be a PA, let w be a lasso shape word, and let $\alpha \in \Delta(Q)$. Then the process induced by w and α on Q is simple.*

Proof: We just have to show that for any $\alpha \in \Delta(Q)$ and $\rho \in \Sigma^*$, the process induced by ρ^ω and α on Q is simple. Let $\{X_n\}_{n\in\mathbb{N}}$ be the non-homogeneous Markov chain induced on Q by α and ρ^ω. Then for all $i \in \{\ 0, 1, \dots, |\rho| - 1\ \}$, the chain $\{X_{n\cdot|\rho|+i}\}_{n\in\mathbb{N}}$ is homogeneous. The result follows from the classical decomposition Theorem of the state space of an homogeneous Markov chain into periodic components of recursive classes, and transient states. ■

Corollary 1. *Let $\mathcal{M}$ be a finite state machine. Then for any $w \in \Sigma^\omega$ generated by $\mathcal{M}$, the process induced by w and α on Q is simple.*

The results of this section along with the results of the previous sub-section give us the following theorem.

Theorem 3. *Given a probabilistic automaton with parity acceptance condition, the question whether there is lasso shape word that is accepted with probability 1 (or positive probability) is PSPACE-complete.*

V. Structurally Simple Automata

In this section we introduce the class of structurally simple automata, which is a structurally defined subclass of probabilistic automata on which every word induce a simple process. We show that the problems associated to this class of automata are decidable (the almost and positive problems are PSPACE-complete and limit problem is in EXPSPACE). We then show that this subclass of simple automata is closed under union and intersection, and finally show that structurally simple automata strictly generalizes HPA and $\#$-acyclic automata. Also note that since every deterministic automaton is HPA, it follows that the class of deterministic automata is strictly contained in structurally simple automata.

A. Simple automata and structural characterization

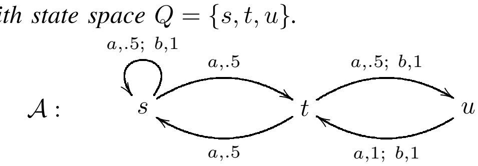
Definition 3 (Simple Automata)**.** *A probabilistic automaton is* simple *if for all $w \in \Sigma^\omega$, the process $\{\mu_n^w\}_{n\in\mathbb{N}}$ induced on its state space by w is simple.*

In [15], given $S \subseteq Q$ and $a \in \Sigma$, the authors define the set $S \cdot a$ as the support of $\delta(S, a)$, and in the case where $S \cdot a = S$, the set $S \cdot a^\#$ as the set of states which are recurrent for the homogeneous Markov chain induced on S by the transition matrix M_a. Next, they define the *support graph* $\mathcal{G}_\mathcal{A}$ of the automaton $\mathcal{A}$ as the graph whose nodes are the subsets of Q, and such that, given $S, T \subseteq Q$, the couple (S, T) is an edge in $\mathcal{G}_\mathcal{A}$ if there exists $a \in \Sigma$ such that $S \cdot a = T$ or $S \cdot a = S$ and $S \cdot a^\# = T$. They present the class of $\#$-acyclic automata as the class of probabilistic automata whose support graph is acyclic.

Definition 4 ([15])**.** *A probabilistic automaton $\mathcal{A}$ is $\#$-acyclic if $\mathcal{G}_\mathcal{A}$ is acyclic.*

We now present a natural generalization of this approach. Given $S \subseteq Q$ and a finite word $\rho \in \Sigma^*$, let $S \cdot \rho = \mathrm{Supp}(\delta(S, \rho))$. If $S \cdot \rho = S$, we define $S \cdot \rho^\#$ as the set of states which are recurrent for the homogeneous Markov chain induced on S by ρ (i.e. by the transition matrix $\{\delta(q, \rho)(q')\}_{q,q'\in S}$).

Example 1. *Consider the following probabilistic automaton $\mathcal{A}$, with state space $Q = \{s, t, u\}$.*



We have $Q \cdot a = Q \cdot a^\# = Q$, and $Q \cdot b = Q \cdot b^\# = Q$. However, $Q \cdot (ab)^\# = \{u\}$.

Given a probabilistic automaton $\mathcal{A}$, an *execution tree* is given by an initial distribution $\alpha \in \Delta(Q)$, or a set of states $A \subseteq Q$, and a finite or infinite word ρ. We use the term execution tree informally for the set of execution runs on $\mathcal{A}$ which can be probabilistically generated when the system is initiated in one of the states of $\mathrm{Supp}(\alpha)$ (or A), and when the word ρ is taken as input.

Definition 5 ($\#$-reductions)**.** *A* $\#$-reduction *is a tuple (A, B, ρ) where $A, B \subseteq Q$ and $\rho \in \Sigma^*$ are such that: (i) $A \neq \emptyset$, (ii) $B \neq \emptyset$, (iii) $A \cap B = \emptyset$, (iv) $(A \cup B) \cdot \rho = A \cup B$, and (v) $(A \cup B) \cdot \rho^\# = B$.*

For simplicity, we may use the term $\#$-reduction for a couple (A, ρ) where $A \subseteq Q$ and $\rho \in \Sigma^*$ are such that $A \cdot \rho = A$ and $A \cdot \rho^\# \neq A$.

Definition 6. *An execution tree (α, ρ) is said to be* chain recurrent *for a probabilistic automaton $\mathcal{A}$ if it does not contain* *a #-reduction. That is, for all $\rho_1, \rho_2 \in \Sigma^*$ such that $\rho_1 \cdot \rho_2$ is a prefix of ρ, $(\delta(\alpha, \rho_1), \rho_2)$ is not a #-reduction. We write* $\mathrm{CRec}(\alpha)$ *for the set of $\rho \in \Sigma^*$ such that (α, ρ) is a chain recurrent execution tree for $\mathcal{A}$.*

The following *key lemma* shows that for any probabilistic automaton $\mathcal{A}$ there exists a constant $\gamma(\mathcal{A}) > 0$ such that the probability to reach any state on a chain recurrent execution tree is either 0 or greater than $\gamma(\mathcal{A})$. Given a probabilistic automaton $\mathcal{A}$, let $\epsilon(\mathcal{A})$ be the smallest non zero probability which appears among the $\delta(q, a)(q')$, where $q, q' \in Q$ and $a \in \Sigma$.

Lemma 1. *Let $\mathcal{A}$ be a probabilistic automaton. For all $q \in Q$, all $\rho \in \mathrm{CRec}(q)$ and all $q' \in \mathrm{Supp}(\delta(q, \rho))$ we have $\delta(q, \rho)(q') \geq \epsilon^{2^{2 \cdot |Q|}}$ where $\epsilon = \epsilon(\mathcal{A})$.*

Proof: Given $U \subseteq Q$ and $\rho \in \Sigma^*$, let

$$\delta^{-1}(\rho)(U) = \{q \in Q \mid \delta(q, \rho)(U) > 0\}.$$

The following remark will be useful: given $\rho = a_1, ..., a_n \in \Sigma^*$, given $U \subseteq Q$ and $i \in \{ 0, 1, 2, \ldots, n-1 \}$, let $S_i = \delta^{-1}(a_{i+1}, ..., a_n)(U)$. Then we have:

1) For all $i \in \{ 0, 1, 2, \ldots, n-1 \}$, $\delta(Q \setminus S_i, a_{i+1}, ..., a_n) \subseteq Q \setminus U$
2) For all $i \in \{ 0, 1, 2, \ldots, n-2 \}$, if $\delta(S_i, a_{i+1}) \subseteq S_{i+1}$, then $\delta(\alpha, a_1, ..., a_i)(S_i) = \delta(\alpha, a_1, ..., a_{i+1})(S_{i+1})$
3) Given $i \in \{ 0, 1, 2, \ldots, n-2 \}$, let k_i be the number of integers $j \in [i; n-2]$ such that $\delta(S_j, a_{j+1}) \not\subseteq S_{j+1}$. Then, for all $i \in \{ 0, 1, 2, \ldots, n-2 \}$,
$$\delta(\alpha, a_1, ..., a_n)(U) \geq \delta(\alpha, a_1, ..., a_i)(S_i) \cdot \epsilon^{k_i}$$

The only non trivial point is the last one. It follows from the fact that for all $\rho \in \Sigma^*$ and $q, q' \in Q$, if $\delta(q, \rho)(q') > 0$, then by definition of ϵ we have $\delta(q, \rho)(q') > \epsilon^{|\rho|}$.

By contradiction, suppose that there exists $\rho \in \mathrm{CRec}(q)$ and $U \subseteq Q$ such that $U \subseteq \mathrm{Supp}(\delta(q, \rho))$ and

$$\delta(q, \rho)(U) < \epsilon^{2^{2 \cdot |Q|}}$$

We show that then we can write $\rho = \rho_1 \cdot \rho_2 \cdot \rho_3$ where ρ_1, ρ_2, ρ_3 are such that $\delta(q, \rho_1)$ can be partitioned into two subsets A and B such that (A, B, ρ_2) is a #-reduction. This contradicts the definition of $\mathrm{CRec}(\mathcal{A})$.

Let $\rho = a_1, ..., a_l$. Given $i \in \{ 0, 1, 2, \ldots, l-1 \}$, let:

- $V_i^n = \delta^{-1}(a_{i+1}, a_{i+2}, ...a_l)(U) \cap \delta(q, a_1, \ldots, a_i)$
- $W_i^n = (Q \setminus V_i^n) \cap \delta(q, a_1, \ldots, a_i)$

Using the third point of the previous remark, since $\delta(q, \rho)(U) < \epsilon^{2^{2 \cdot |Q|}}$, there exists a least k integers i in $\{ 1, 2, \ldots, l-2 \}$ such that $\delta(V_i^n, a_{i+1}) \not\subseteq V_{i+1}^n$, where k satisfies $\epsilon^k < \epsilon^{2^{2 \cdot |Q|}}$. Thus, $k \geq 2^{2 \cdot |Q|}$. Let $n_1, ..., n_{2^{2 \cdot |Q|}}$ be the $2^{2 \cdot |Q|}$ largest integers in $\{ 1, 2, \ldots, l \}$ such that $\delta(V_i^n, a_{i+1}) \not\subseteq V_{i+1}^n$.

By a simple cardinality argument, there exist $i < j$ in $\{ 1, 2, \ldots, 2^{2 \cdot |Q|} \}$ such that $V_{n_i}^n = V_{n_j}^n$ and $W_{n_i}^n = W_{n_j}^n$. Let $\rho_1 = a_1, \ldots, a_{n_i - 1}$, $\rho_2 = a_{n_i}, \ldots, a_{n_j - 1}$ and $\rho_3 = a_{n_j}, \ldots, a_n$. Then we are in the following situation:

$$\begin{tikzcd}
q \arrow[r, "\rho_1"] \arrow[dr, "\rho_1"'] & W^n_{n_i} \arrow[r, "\rho_2"] & W^n_{n_i} \\
& V^n_{n_i} \arrow[r, "\rho_2"] \arrow[ur, "\rho_2"] & V^n_{n_i}
\end{tikzcd}$$

That is, $\delta(q, \rho_1)$ can be partitionned into two subsets $V_{n_i}^n$ and $W_{n_i}^n$ such that $\delta(W_{n_i}^n, \rho_2) \subseteq W_{n_i}^n$, $\delta(V_{n_i}^n, \rho_2) \subseteq V_{n_i}^n \cup W_{n_i}^n$, and $\delta(V_{n_i}^n, \rho_2) \not\subseteq V_{n_i}^n$. This implies that there exists $A \subseteq V_{n_i}^n$ such that $(A, (V_{n_i}^n \setminus A) \cup W_{n_i}^n, \rho_2)$ is a #-reduction. Since A, $(V_{n_i}^n \setminus A) \cup W_{n_i}^n$ is a partition of $\delta(q, \rho_1)$, we get that the execution tree (q, ρ) contains a #-reduction. This is a contradiction since $\rho \in \mathrm{CRec}(q)$. ∎

Definition 7 (Structurally simple automata)**.** *An automaton $\mathcal{A}$ is* structurally simple *if for all $\rho \in \Sigma^*$ and $C \subseteq Q$, if $D \subseteq C$ is minimal among the $D \subseteq Q$ such that $C \overset{\#-\rho}{\to} D$, we have that (D, ρ) is chain recurrent. Here $\# - \rho$ intuitively denotes an iterated #-reachability with the word ρ.*

The notion of iterated #-reachability is the natural extension of #-reachability (see [11] for further details along with illustrating examples). We now prove that all the structurally simple automata are simple. We show that on a structurally simple automaton, given $w \in \Sigma^\omega$, the associated execution tree can be decomposed as a sequence of a bounded number of chain recurrent execution trees. The key Lemma 1 is then used to bound the probabilities which appear.

Lemma 2. *Let $\{\mu_n^w\}_{n \in \mathbb{N}}$ be the process generated by a word $w = a_1, a_2, ... \in \Sigma^\omega$ on a probabilistic automaton. Given $n \geq 1$ recall that $w[1..n] = a_1, ..., a_n$. Suppose that there exists $\gamma > 0$ and $N \geq 0$ such that for all $n \geq N$ and all $q \in \mathrm{Supp}(\delta(\alpha, w[1..n]))$ we have $\delta(\alpha, w[1..n])(q) > \gamma$. Then the process is simple.*

Proof: For all $n \in \mathbb{N}$, we let $A_n = \mathrm{Supp}(\delta(\alpha, w[1..n]))$ and $B_n = Q \setminus A_n$. By hypothesis, for all all $n \geq N$ and all $q \in A_n$, we have $\mu_n^w(q) = \delta(\alpha, w[1..n])(q) > \gamma$. Moreover, for all n and all $q \in B_n$ we have $\mu_n^w(q) = 0$. This shows that the process is simple. ∎

We introduce the notion of *sequence of recurrent execution trees* in order to represent a process which may not be chain recurrent, but which can be decomposed as a sequence of a finite number of chain recurrent execution trees. The *length* of the sequence measures the number of steps which do not belong to a chain recurrent subsequence, and will be useful to bound the probabilities which appear. Lemma 3 uses the key Lemma 1.

Definition 8 (Sequence of recurrent execution trees)**.** *A* sequence of recurrent execution trees *is a finite sequence $(\alpha_1, \rho_1), \rho'_1, (\alpha_2, \rho_2), \rho'_2, ...(\alpha_k, \rho_k)$ such that:*

- *$\rho_k \in \Sigma^\omega$, and for $i \in [1; k-1]$ we have $\rho_i, \rho'_i \in \Sigma^*$*
- *For all $i \in [2; k]$ we have $\mathrm{Supp}(\alpha_i) \subseteq \mathrm{Supp}(\delta(\alpha_{i-1}, \rho_{i-1} \cdot \rho'_{i-1}))$*
- *All the execution trees (α_i, ρ_i) are chain recurrent*

The length *of the sequence is defined as $\sum_{i=1}^{k-1} |\rho'_i|$.*

Given an execution tree (α, w), a subsequence of recurrent execution trees of (α, w) is a sequence of recurrent execution trees $(\alpha_1, \rho_1), \rho'_1, (\alpha_2, \rho_2), \rho'_2, ...(\alpha_k, \rho_k)$ such that $\alpha = \alpha_1$ and $w = \rho_1 \cdot \rho'_1 \cdot \rho_2 \cdot \rho'_2 \cdot \dots \cdot \rho_k$.

Lemma 3. *Let $\mathcal{A}$ be a probabilistic automaton. Suppose that there exists $K \in \mathbb{N}$ such that for all execution trees (α, ρ), there exists a subsequence of recurrent execution trees of length at most K. Then $\mathcal{A}$ is simple.*

Proof: Let $\rho \in \Sigma^\omega$, and let $\{\mu_n^\rho\}_{n\in\mathbb{N}}$ be the process induced on Q by ρ. By hypothesis, let $(\alpha, \rho_1), \rho'_1, (\alpha_2, \rho_2), \rho'_2, ...(\alpha_k, \rho_k)$ be a sub-sequence of recurrent execution trees of (α, ρ) of length at most K. That is, we have $\sum_{i=1}^{k-1} |\rho'_i| \leq K$. By definition, for all $i \in \{ 1, \dots k-1 \}$ we have $\rho_i \in \Sigma^*$ and $\rho'_i \in \Sigma^*$, and $\rho_k \in \Sigma^\omega$. For all $i \in \{ 1, \dots, k-1 \}$, let $\alpha'_i = \delta(\alpha_i, \rho_i)$. We are in the following situation:

$$\alpha \xrightarrow{\rho_1} \alpha'_1 \xrightarrow{\rho'_1} \alpha_2 \xrightarrow{\rho_2} \alpha'_2 \xrightarrow{\rho'_2} \alpha_3 \dots \xrightarrow{\rho'_{k-1}} \alpha_k \xrightarrow{\rho_k}$$

We know that:

- For all $i \in \{ 1, \dots, k-2 \}$, the execution tree (α_i, ρ_i) is chain recurrent;
- (α_k, ρ_k) is chain recurrent.

We show that the process $\{\mu_n^\rho\}_{n\in\mathbb{N}}$ satisfies the hypothesis of Lemma 2. As before, let $\epsilon = \epsilon(\mathcal{A})$ be the minimal non zero probability which appears among the values $\delta(q, a)(q')$ when $q, q' \in Q$ and $a \in \Sigma$. Let $\lambda = \epsilon^{2^{2\cdot|Q|}}$. By Lemma 1, for all $q \in Q$, all $\rho' \in \mathrm{CRec}(q)$ and all $q' \in \mathrm{Supp}(\delta(q, \rho'))$, we have $\delta(q, \rho')(q') \geq \lambda$. We claim that for all $i \in \{1, \dots, k-1\}$ and all $q \in \mathrm{Supp}(\alpha'_i)$, we have $\alpha'_i(q) \geq (\mathrm{Min}_{q\in\mathrm{Supp}(\alpha)}\alpha(q)) \cdot \lambda^i \cdot \epsilon^{K\cdot i}$. We prove this result by induction on i:

- The case $i = 1$ follows from the use of Lemma 1 on the chain recurrent execution tree (α_1, ρ_1).
- Suppose the proposition true until $i \in \{ 1, \dots, k-2 \}$. Let $q' \in \mathrm{Supp}(\alpha'_{i+1})$. then there exists $q \in \mathrm{Supp}(\alpha'_i)$ such that $\delta(q, \rho_i \cdot \rho'_i)(q') > 0$. Let $q'' \in Q$ be such that $\delta(q, \rho_i)(q'') > 0$, and $\delta(q'', \rho'_i)(q') > 0$. By the use of Lemma 1 on the chain recurrent execution tree (α_i, ρ_i), we know that $\delta(q, \rho_i)(q'') > \lambda$. By definition of ϵ and K, we have that $\delta(q'', \rho'_i)(q') \geq \epsilon^{|\rho'_i|}$, hence $\delta(q'', \rho'_i)(q') \geq \epsilon^K$. We have $\alpha'_{i+1}(q') \geq \alpha_i(q) \cdot \delta(q, \rho_i \cdot \rho'_i)(q')$. Since by induction hypothesis we have that $\alpha_i(q) \geq (\mathrm{Min}_{q\in\mathrm{Supp}(\alpha)}\alpha(q)) \cdot \lambda^i \cdot \epsilon^{K\cdot i}$, we get that $\alpha'_{i+1}(q') \geq (\mathrm{Min}_{q\in\mathrm{Supp}(\alpha)}\alpha(q)) \cdot \lambda^{i+1} \cdot \epsilon^{K\cdot i+1}$, hence the result.

Now, let $N = \sum_{i=1}^{k-1}(|\rho_i| + |\rho'_i|)$, and let $n \geq N$. Since (α_k, ρ_k) is chain recurrent, we can apply the same method for the chain recurrent execution tree (α_k, ρ_k). As a conclusion, we see that the process $\{\mu_n^\rho\}_{n\in\mathbb{N}}$ satisfies the hypothesis of Lemma 2 with the parameters $N = \sum_{i=1}^{k-1}(|\rho_i| + |\rho'_i|)$ and $\gamma = (\mathrm{Min}_{q\in\mathrm{Supp}(\alpha)}\alpha(q)) \cdot \lambda^K \cdot \epsilon^{K\cdot K}$. This completes the proof and we have the desired result. ∎

Lemma 4. *Suppose that $\mathcal{A}$ is structurally simple. Then for all execution trees (α, w), there exists a subsequence of recurrent execution trees of length at most $2^{2\cdot|Q|}$.*

The proof of Lemma 4 is available in [11]. Theorem 4 follows from Lemma 3 and Lemma 4.

Theorem 4. *All structurally simple automata are simple.*

B. Decision problems for structurally simple automata

For the following of this sub-section, $\mathcal{A}$ is a structurally simple automaton with state space Q and initial distribution α. We consider the complexity of the decision problems related to infinite words on structurally simple PAs. The upper bound on the complexity in Theorem 5 follows from the results of Section IV, since the process induced on a simple PA by a word $w \in \Sigma$ is always simple. The lower bound follows from the fact that the PA used for the lower bound of Section IV is structurally simple.

Theorem 5. *The almost and positive problems are PSPACE-complete for parity conditions on structurally simple PAs.*

In Proposition 6 of [15], the authors show that if $F \subseteq Q$ is reachable from a state q_0 in the support graph of $\mathcal{A}$, then it is limit reachable from q_0 in $\mathcal{A}$. A generalization of this result to the extended support graph gives half of Proposition 9 (details with complete proof in [11]). The limit problem is in EXPSPACE for reachability conditions (details in [11] for the definition of the extended support graph). Theorem 6 shows that the limit parity problem is also decidable for simple automata.

Proposition 9. *Let $\mathcal{A}$ be a structurally simple automaton, and let $F \subseteq Q$. Then* **(1)** *F is reachable from $\mathrm{Supp}(\alpha)$ in the extended support graph of $\mathcal{A}$ iff* **(2)** *it is limit reachable from α in $\mathcal{A}$.*

Theorem 6. *The limit problem is in EXPSPACE for parity conditions on structurally simple PAs.*

The following theorem establishes the decidability of the problem that given a probabilistic automaton whether the automaton is structurally simple. For both Theorem 6 and Theorem 7 we show an EXPSPACE upper bound, and PSPACE lower bounds follow from our lower bounds of Section IV, and establishing the exact complexity is an open problem.

Theorem 7. *We can decide in EXPSPACE whether a given probabilistic automaton is structurally simple or not.*

C. Closure properties for Structurally Simple Automata

Given $\mathcal{A}_1 = (S_1, \Sigma, \delta_1, \alpha_1)$ and $\mathcal{A}_2 = (S_2, \Sigma, \delta_2, \alpha_2)$ two structurally simple automata on the same alphabet Σ, the construction of the Cartesian product automaton $\mathcal{A}_1 \bowtie \mathcal{A}_2$ is standard. We detail this construction in [11], along with the proof of the following proposition.

Proposition 10. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two structurally simple automata. Then $\mathcal{A} = \mathcal{A}_1 \bowtie \mathcal{A}_2$ is structurally simple.*

We prove that the classes of languages recognized by structurally simple automata under various semantics (positive parity, almost parity) are robust. This property relies on the fact that one can construct the intersection or the union of two parity (non-probabilistic) automata using only product constructions and change in the semantics (going from parity to Streett or Rabin, and back to parity; see [20], [2] for details of Rabin and Streett conditions and the translations). By Proposition 10, such transformations keep the automata simple.

Theorem 8. *The class of languages recognized by structurally simple automata under positive (resp. almost) semantics and parity condition is closed under union and intersection.*

The class of languages recognized by structurally simple automata is also closed under complementation. The proof of this fact is technical and the result is obtained by following the steps of the proof of [2] that shows that probabilistic automata is closed under complementation and verifying that all the steps preserve the structurally simple property.

D. Subclasses of Simple Automata

In this section we show that both $\#$-acyclic automata (recall Definition 4) and hierarchical probabilistic automata are strict subclasses of simple automata.

Proposition 11. *The class of structurally simple automata strictly subsumes the class of $\#$-acyclic automata.*

Another restriction of Probabilistic Automata which has been considered is the model of *Hierarchical PAs*, presented first in [7]. Intuitively, a hierarchical PA is a probabilistic automaton on which a *rank function* must increase on every runs. This condition imposes that the induced processes are ultimately deterministic with probability one.

Definition 9 ([7])**.** *Given $k \in \mathbb{N}$, a PA $\mathcal{B} = (Q, q_s, Q, \delta)$ over an alphabet Σ is said to be a k-*level hierarchical PA *(k-HPA) if there is a function $\mathrm{rk} : Q \to \{0, 1, ..., k\}$ such that the following holds:*

Given $j \in \{0, 1, ..., k\}$, let $Q_j = \{q \in Q \mid \mathrm{rk}(q) = j\}$. For every $q \in Q$ and $a \in \Sigma$, if $j_0 = \mathrm{rk}(q)$ then $post(q, a) \subseteq \cup_{j_0 \le l \le k} Q_l$ and $|post(q, q) \cap Q_{j_0}| \le 1$.

Proposition 12. *The class of structurally simple automata strictly subsumes the class of Hierarchical PAs.*

It follows that our decidability results for structurally simple PAs both unifies and generalizes the decidability results previously known for $\#$-acyclic (for limit reachability) and hierarchical PA (for almost and positive Büchi).

VI. Conclusion

In this work we have used a very general result from stochastic processes, namely the decomposition-separation theorem, to identify simple structure of tail σ-fields, and used them to define simple processes on probabilistic automata. We showed that under the restriction of simple processes the almost and positive decision problems are decidable for all parity conditions. We then characterized structurally a subclass of the class of simple automata on which every process is simple. We showed that this class is decidable, robust, and that it generalizes the previous known subclasses of probabilistic automata for which the decision problems were decidable. Our techniques also show that for lasso shape words the almost and positive decision problems are decidable for all probabilistic automata. We believe that our techniques will be useful in future research for other decidability results related to probabilistic automata and more general probabilistic models (such as partially observable Markov decision processes and partial-observation stochastic games).

Acknowledgements. The research was supported by Austrian Science Fund (FWF) Grant No P 23499-N23 on Modern Graph Algorithmic Techniques in Formal Verification, FWF NFN Grant No S11407-N23 (RiSE), ERC Start grant (279307: Graph Games), and Microsoft faculty fellows award.

References

[1] C. Baier, N. Bertrand, and M. Größer. On decision problems for probabilistic Büchi automata. In *FOSSACS*, pages 287–301. Springer, 2008.
[2] C. Baier and M. Großer. Recognizing ω-regular languages with probabilistic automata. In *LICS*, pages 137–146, 2005.
[3] N. Bertrand, B. Genest, and H. Gimbert. Qualitative determinacy and decidability of stochastic games with signals. In *LICS*, pages 319–328, 2009.
[4] D. Blackwell. Finite non-homogeneous chains. *Ann. Math.*, 46:594–599, 1945.
[5] D. Blackwell and D. Freedman. The tail σ-field of a Markov chain and a theorem of Orey. *The Annals of Mathematical Statistics*, 35(3):1291–1295, 1964.
[6] R. G. Bukharaev. Probabilistic automata. *Journal of Mathematical Sciences*, 13:359–386, 1980.
[7] R. Chadha, A. Sistla, and M. Viswanathan. Power of randomization in automata on infinite strings. In *CONCUR*, pages 229–243. Springer, 2009.
[8] R. Chadha, A.P. Sistla, , and M. Viswanathan. On the expressiveness and complexity of randomization in finite state monitors. *Journal of the ACM*, 56:1–44, 2009.
[9] K. Chatterjee and L. Doyen. Partial-observation stochastic games: How to win when belief fails. In *LICS*, 2012.
[10] K Chatterjee and Thomas A. Henzinger. Probabilistic Automata on Infinite Words: Decidability and Undecidability Results. *ATVA*, 2010.
[11] K. Chatterjee and M. Tracol. Decidable problems for probabilistic automata on infinite words. In *Arxiv Technical Report (CoRR): http://arxiv.org/abs/1104.0127*, 2011.
[12] H. Cohn. Products of stochastic matrices and applications. *International Journal of Mathematics and Mathematical Sciences*, 12(2):209–233, 1989.
[13] M. de Rougemont and M. Tracol. Statistic Analysis for Probabilistic Processes. In *LICS*, pages 299–308, 2009.
[14] N. Fijalkow, H. Gimbert, and Y. Oualhadj. Deciding the value 1 problem of probabilistic leaktight automata. In *LICS*, 2012.
[15] H. Gimbert and Y. Oualhadj. Probabilistic automata on finite words: Decidable and undecidable problems. *ICALP*, pages 527–538, 2010.
[16] D. Kozen. Lower bounds for natural proof systems. In *FOCS*, pages 254–266, 1977.
[17] A. Paz. *Introduction to probabilistic automata*. Academic Press, Inc. Orlando, FL, USA, 1971.
[18] M.O. Rabin. Probabilistic automata. *Information and Control*, 6:230–245, 1963.
[19] I. Sonin. The asymptotic behaviour of a general finite nonhomogeneous Markov chain (the decomposition-separation theorem). *Lecture Notes-Monograph Series*, 30:337–346, 1996.
[20] W. Thomas. Languages, automata, and logic. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 3, Beyond Words, chapter 7, pages 389–455. Springer, 1997.
[21] M.Y. Vardi. Automatic verification of probabilistic concurrent finite state programs. In *FOCS*, pages 327–338, 1985.