Randomization in Automata on Infinite Trees

ARNAUD CARAYOL, LIGM (CNRS & Université Paris Est)
AXEL HADDAD, LIAFA (CNRS & Université Paris Diderot - Paris 7) & LIGM (CNRS & Université Paris Est)
OLIVIER SERRE, LIAFA (CNRS & Université Paris Diderot - Paris 7)

We study finite automata running over infinite binary trees. A run of such an automaton over an input tree is a tree labeled by control states of the automaton: the labeling is built in a top-down fashion and should be consistent with the transitions of the automaton. A branch in a run is accepting if the ω-word obtained by reading the states along the branch satisfies some acceptance condition (typically an ω-regular condition such as a Büchi or a parity condition). Finally, a tree is accepted by the automaton if there exists a run over this tree in which *every* branch is accepting.

In this article, we consider two relaxations of this definition, introducing a qualitative aspect. First, we relax the notion of accepting run by allowing a negligible set (in the sense of measure theory) of nonaccepting branches. In this qualitative setting, a tree is accepted by the automaton if there exists a run over this tree in which *almost every* branch is accepting. This leads to a new class of tree languages, *qualitative tree languages*. This class enjoys many good properties: closure under union and intersection (but not under complement), and emptiness is decidable in polynomial time. A dual class, *positive tree languages*, is defined by requiring that an accepting run contains a non-negligeable set of branches.

The second relaxation is to replace the existential quantification (a tree is accepted if there exists some accepting run over the input tree) with a probabilistic quantification (a tree is accepted if almost every run over the input tree is accepting). For the run, we may use either classical acceptance or qualitative acceptance. In particular, for the latter, we exhibit a tight connection with partial observation Markov decision processes. Moreover, if we additionally restrict operation to the Büchi condition, we show that it leads to a class of probabilistic automata on infinite trees enjoying a decidable emptiness problem. To our knowledge, this is the first positive result for a class of probabilistic automaton over infinite trees.

Categories and Subject Descriptors: F.4.3 [**Mathematical Logic and Formal Languages**]: Formal Languages—*Classes defined by grammars or automata; Decision problems*; F.1.1 [**Computation by Abstract Devices**]: Models of Computation—*Automata*

General Terms: Theory, Automata, Games

Additional Key Words and Phrases: Finite automata on infinite trees, measure theory, (partial observation) Markov decision processes, probabilistic automata

ACM Reference Format:
Arnaud Carayol, Axel Haddad, and Olivier Serre. 2014. Randomization in automata on infinite trees. ACM Trans. Comput. Logic 15, 3, Article 24 (June 2014), 33 pages.
DOI: http://dx.doi.org/10.1145/2629336

24

Authors' addresses: Arnaud Carayol, LIGM, 5 boulevard Descartes, Champs-sur-Marne, 77454 Marne-la-Vallée Cedex 2, France; Axel Haddad, LIAFA, Université Paris Diderot - Paris 7, Case 7014, 75205 Paris Cedex 13, France; Olivier Serre, LIAFA, Université Paris Diderot - Paris 7, Case 7014, 75205 Paris Cedex 13, France; email: opserre@gmail.com.

DOI: http://dx.doi.org/10.1145/2629336

1. INTRODUCTION

Roughly speaking, a finite automaton on infinite trees is a finite memory machine that takes as input an infinite node-labeled binary tree and processes it in a top-down fashion as follows. It starts at the root of the tree in its initial state and picks (possibly nondeterministically) two successor states, one per child, according to the current control state, the letter at the current node and the transition relation. Then, the computation proceeds in parallel from both sons, and so on. Hence, a run of the automaton on an input tree is a labeling of this tree by control states of the automaton that should satisfy the local constraints imposed by the transition relation. A branch in a run is accepting if the ω-word obtained by reading the states along the branch satisfies some acceptance condition (typically, an ω-regular condition such as a Büchi or a parity condition). Finally, a tree is accepted by the automaton if there exists a run over this tree in which *every* branch is accepting. An ω-regular tree language is a tree language accepted by some tree automaton equipped with a parity condition.

Finite automata on infinite trees were originally introduced by Rabin [Rabin 1969] to prove the decidability of the Monadic Second-Order Logic (MSOL) over the full binary tree. Indeed, Rabin proved that for any MSOL formula, one can construct a tree automaton that accepts a nonempty language if and only if the original formula holds at the root of the full binary tree. These automata were also successfully used by Rabin [Rabin 1972] to solve Church's synthesis problem [Church 1962], which seeks to construct a circuit based on a formal specification (typically expressed in MSOL) describing the desired input/output behavior. His approach was to represent the set of all possible behaviors of a circuit by an infinite tree (directions are used to code the input while node labels along a branch code the output) and to reduce the synthesis problem to emptiness of a tree automaton accepting all those trees coding circuits satisfying the specification.

Since then, automata on infinite trees and their variants have been intensively studied and have found many applications, particularly in logic. Connections between automata on infinite trees and logic are discussed in two excellent surveys [Thomas 1997; Vardi and Wilke 2007].

A fundamental result of Rabin is that ω-regular tree languages form a Boolean algebra [Rabin 1969]. The hard part in this proof is the complementation, and, since the publication of this result in 1969, it has been a challenging problem to simplify the proof. A much simpler one was obtained in Gurevich and Harrington [1982] making use of two-player games on graphs for checking membership of a tree in the language accepted by the automaton: the first player (called *Automaton*) builds a run on the input tree while the second player (called *Pathfinder*) tries to exhibit a rejecting branch in the run. Beyond this result, the tight connection between automata and games is one of the main tools in automata theory (see e.g., Grädel et al. [2002] and Löding [2011]).

In this article, we consider variations of the classical model of tree automata over infinite trees. These variations involve probabilities in two different ways and preserve the fruitful connection with game theory.

In the first part of this article, we consider a relaxed notion of an accepting run. While the usual definition requires for a run to be accepting is that *all* branches in it satisfy the acceptance condition, we allow a negligible set of nonaccepting branches. In this *qualitative* setting, a tree is accepted by the automaton if there exists a run over this tree in which *almost every* (in the usual sense from measure theory) branch is accepting. With the parity condition, this leads to a new class of tree languages that we call *qualitative tree languages*. We show that this class enjoys many desirable properties including closure under union and intersection (but not under complement) and emptiness that is decidable in polynomial time (contrasting with the fact that no polynomial algorithm is known for the emptiness test of standard parity tree automata). We also prove that there exists a strong connection between automata accepting qualitative tree languages and Markov decision processes, which play here a similar role as do two-player games for usual tree automata. We also discuss the *positive* setting, in which a run is accepting if the set of accepting branches in it has a strictly positive measure.

The idea of allowing a certain amount of rejecting branches in a run was already considered in Beauquier et al. [1991] and Beauquier and Niwiński [1995], where it was required that the number of accepting branches in a run belong to a specified set of cardinals Γ. In particular, they proved that if Γ consists of all cardinals greater than some γ, then one obtains a regular tree language. Qualitative tree languages, as defined in this article, are not captured by the work of Beauquier and Niwiński [1995]. Indeed, our classes are incomparable with regular tree languages.

In the second part of this article, we investigate probabilistic automata on infinite trees. Acceptance by an automaton is based on existential quantification: an input is accepted if *there exists* an accepting run over it. Probabilistic automata are an alternative way to define acceptance. On finite words, they have been introduced by Rabin [1963]. Compared with the standard setting, the nondeterministic guesses are replaced by random choices (according to some probabilistic distribution depending on the control state and the input letter). Hence, the set of transitions is replaced by a probability distribution over the set of all transitions, which induces a probability measure on the set of runs of the automaton and acceptation is defined using a threshold $0 < \lambda < 1$ on the probability of a run to be accepting. In contrast to the nondeterministic setting, the emptiness problem for probabilistic automata on finite words is undecidable [Paz 1971].

The probabilistic model was recently extended to infinite words Baier and Größer [2005][1] and studied in more details in Baier et al. [2008], Chadha et al. [2009], and Chatterjee et al. [2009]. In addition to the threshold criterion, two additional semantics were considered: ***almost-sure*** and ***probable***, which respectively correspond to a probability 1 or >0 for a run to be accepting.[2] Surprisingly, the class of languages defined by Büchi automata with the probable semantics is closed under complement, which implies that it coincides with the class of languages defined by co-Büchi automata with the almost-sure semantics.[3] The emptiness problem for Büchi automata with the almost-sure semantics, as well as for co-Büchi automata with the probable semantics, are decidable. However, the emptiness problem for Büchi automata with the probable semantics, as well as for co-Büchi automata with the almost-sure semantics, are undecidable. Of course, emptiness is undecidable when considering a threshold semantics. See Baier et al. [2012] for a very rich overview of this topic.

In this article, we consider probabilistic automata on infinite trees. We first focus on the *almost-sure* semantics, that is, a tree is accepted if almost every run over it is accepting; and we later discuss the *probable* semantics, that is, a tree is accepted if the set of accepting runs on it has a (strictly) positive measure. Of course, each semantics can be used in combination with the acceptance criteria on runs: the classical one (all

[1]A previous attempt by Reisz [1999] should be mentioned, but this approach does not make real use of probabilities because, in this setting, an input word is accepted if, after some time, the behavior in the run becomes deterministic.

[2]In the finite word case, the almost-sure and probable acceptance are trivial because the set of runs for a given word is finite.

[3]Indeed, let L be accepted by a Büchi automaton with the probable acceptance. As one can complement, there is a Büchi automaton with the probable semantics $\mathcal{A}$ such that $\overline{L}$ is the language accepted by $\mathcal{A}$. If one sees $\mathcal{A}$ as a co-Buchi automaton $\mathcal{B}$ (final states becoming the forbidden ones) with an almost-sure semantics, $\mathcal{B}$ accepts a word if and only if $\mathcal{A}$ does not. Hence, $\mathcal{B}$ recognizes the complement of $\overline{L}$; namely, L.

branches are accepting), the qualitative one (almost all branches are accepting), and the positive one (there is a non-negligible set of accepting branches). For all these combinations, we establish that the definition makes sense (that is, we prove measurability of the set of accepting runs).

For the qualitative criterion on runs combined with the almost-sure semantics, as well as for the probable criterion on runs combined with the positive semantics, we prove that there exists a strong connection with partial observation Markov decision processes. This condition is independent of the acceptance condition on branches (Büchi, co-Büchi, parity...). In particular, for the Büchi (*resp.* co-Büchi) acceptance condition on branches, probabilistic automata on infinite trees with the qualitative criterion on runs combined with the almost-sure semantics (*resp.* with the positive criterion on runs combined with the probable semantics) enjoy a decidable emptiness problem. To our knowledge, this is the first positive result for a class of probabilistic automata over infinite trees. On the negative side, we derive from undecidability results on probabilistic automata on infinite words similar results for probabilistic automata on infinite trees. Note that although being immediate for the classical criterion on runs, such a reduction is not as simple for the qualitative and positive criteria.

The article is organized as follows. In Section 2, we introduce general notations and definitions. In Section 3, we define the class of qualitative tree languages, present their properties (closure properties, decidability properties) and their tight connections with Markov decision processes, and establish that they are incomparable with regular tree languages. We also discuss possible variants of the model (leading to the notion of positive tree languages). In Section 4, we focus on the probabilistic model. We first justify the validity of the definition and give some examples. Then we provide connections with partial observation Markov decision processes from which we derive positive results for the emptiness problem on a subclass. Finally, we discuss variants of the probabilistic model. Section 5 summarizes the contributions of the article.

2. DEFINITIONS

2.1. Words and Trees

An ***alphabet*** A is a finite set of letters. In the sequel, A^* denotes the set of finite words over A, and A^ω the set of infinite words over A. The ***empty word*** is written ε; the length of a word u is denoted by $|u|$. Let u be a finite word and v be a (possibly infinite) word. Then, $u \cdot v$ denotes the ***concatenation*** of u and v; the word u is a ***prefix*** of v, denoted $u \sqsubseteq v$, if there exists a word w such that $v = u \cdot w$. We denote by $u \sqsubset v$ the fact that u is a strict prefix of v (*i.e.*, $u \sqsubseteq v$ and $u \neq v$). For some word u and some integer $k \geq 0$, we denote by u^k the word obtained by concatenating k copies of u (with the convention that $u^0 = \varepsilon$).

In this article, we consider full binary node-labeled trees. An ***A-labeled tree*** t is a mapping from $\{0, 1\}^*$ to A. In this context, an element $u \in \{0, 1\}^*$ is called a ***node***, and the node $u \cdot 0$ (*resp.* $u \cdot 1$) is the ***left son*** (*resp.* ***right son***) of u. The node ε is called the ***root***. We shall refer to $|u|$ as the ***depth*** of u. The letter $t(u)$ is called the ***label*** of u in t.

A ***branch*** is an infinite word $\pi \in \{0, 1\}^\omega$. We write $\mathrm{Br} = \{0, 1\}^\omega$ for the set of all branches. A node u belongs to a branch π if u is a prefix of π. For an A-labeled tree t and a branch $\pi = \pi_0\pi_1 \cdots$ we define the ***label*** of π as the ω-word $t(\pi) = t(\varepsilon)t(\pi_0)t(\pi_0\pi_1)t(\pi_0\pi_1\pi_2)\cdots$. The ***cone*** going through a node u is the set $\mathrm{Cone}(u) = u \cdot \{0, 1\}^\omega$. A ***sub-cone*** of a cone $\mathrm{Cone}(u)$ is a cone $\mathrm{Cone}(v)$ with $u \sqsubseteq v$.

Given a tree t and a node u, the ***subtree of t rooted at u*** denoted $t[u]$ is the tree defined by $t[u](v) = t(u \cdot v)$. A tree t is said to be ***regular*** if it contains only finitely many different subtrees, that is, the set $\{t[u] \mid u \in \{0, 1\}^*\}$ is finite.

We assume that the reader is familiar with basic notions of measure theory and probability theory, and we use Bauer [1996, 2001] as references for all known results related to this field. Let $\mathcal{F}_{\rm Br}$ be the σ-algebra generated by the set of cones(i.e., the smallest set of subsets of $\{0, 1\}^\omega$ containing the cones and closed under countable union and complementation). Let μ be the unique probability measure on $\mathcal{F}_{\rm Br}$ such that for all $u \in \{0, 1\}^*$, $\mu(\text{Cone}(u)) = 2^{-|u|}$. The existence and uniqueness of μ are guaranteed by Carathéodory's extension theorem [Bauer 2001, Theorem 5.6, p. 24]. For all $0 < p < 1$, a probability measure μ_p is similarly defined by taking $\mu_p(\text{Cone}(u)) = p^{|u|_0}(1-p)^{|u|_1}$ where $|u|_0$ and $|u|_1$ respectively designate the number of occurrences of 0 and 1 in u. In particular, the measure μ corresponds to $\mu_{1/2}$.

2.2. Tree Automata and Regular Tree Languages

A ***tree automaton*** $\mathcal{A}$ is a tuple $\langle A, Q, q_{ini}, \Delta, \text{Acc}\rangle$ where A is the ***input alphabet***, Q is the finite ***set of states***, $q_{\rm ini} \in Q$ is the ***initial state***, $\Delta \subseteq Q \times A \times (Q \times Q)$ is the ***transition relation,*** and $\text{Acc} \subseteq Q^\omega$ is the ***acceptance condition***. In the following, we use the notation $q \overset{a}{\to} (q_0, q_1)$ as a shorthand for $(q, a, (q_0, q_1)) \in \Delta$. An automaton is ***deterministic*** if $q \overset{a}{\to} (q_0, q_1)$ and $q \overset{a}{\to} (q_0', q_1')$ implies $q_0 = q_0'$ and $q_1' = q_1'$. An automaton is ***complete*** if, for all $q \in Q$ and $a \in A$, there is at least one pair $(q_0, q_1) \in Q^2$ such that $q \overset{a}{\to} (q_0, q_1)$.

Given an A-labeled tree t, a *run* of $\mathcal{A}$ over t is a Q-labeled tree ρ such that the root is labeled by the initial state, that is, $\rho(\varepsilon) = q_{\rm ini}$; for all nodes u, $(\rho(u), t(u), \rho(u\cdot 0), \rho(u\cdot 1)) \in \Delta$.

A branch $\pi \in \{0, 1\}^\omega$ is ***accepting*** in the run ρ if $\rho(\pi) \in \text{Acc}$. A run ρ is ***accepting*** if all its branches are accepting. Finally, a tree t is ***accepted*** if there exists an accepting run of $\mathcal{A}$ over t. The set of all trees accepted by $\mathcal{A}$ is denoted $L(\mathcal{A})$.

We consider the following classical acceptance conditions:

—A ***reachability*** condition is given by a subset $F \subseteq Q$ of final states by letting $Reach(F) = Q^* F Q^\omega$, that is, a branch is accepting if it contains a final state.
—A ***Büchi*** condition is given by a subset $F \subseteq Q$ of final states by letting $Buchi(F) = (Q^* F)^\omega$; that is, a branch is accepting if it contains infinitely many final states.
—A ***co-Büchi*** condition is given by a subset $F \subseteq Q$ of forbidden states by letting $coBuchi(F) = Q^*(Q \setminus F)^\omega$; that is, a branch is accepting it contains finitely many forbidden states.
—A ***parity*** condition is given by a color mapping $\text{Col} : Q \to \mathbb{N}$ by letting $Parity = \{q_0 q_1 q_2 \cdots \mid \liminf(\text{Col}(q_i))_i \text{ is even}\}$; that is, a branch is accepting if the smallest color appearing infinitely often is even.

All these conditions are examples of ω-regular acceptance conditions; that is, Acc is regular set of ω-words [Perrin and Pin 2004].

Remark 1. The parity condition is expressive enough to capture the general case of an arbitrary ω-regular condition. Indeed, it is well known that Acc is accepted by a *deterministic* parity word automaton. By taking the synchronized product of this automaton with the tree automaton, we obtain a parity tree automaton accepting the same language (see, e.g., Perrin and Pin [2004]).

When it is clear from the context, we may replace, in the description of $\mathcal{A}$, *Acc* with F (for a reachability, Büchi, or co-Büchi condition) or Col (for a parity condition), and we shall refer to the automaton as a reachability (*resp.* Büchi, co-Büchi, parity) tree automaton. A set L of trees is a ***regular*** language if there exists a parity tree automaton $\mathcal{A}$ such that $L = L(\mathcal{A})$. The class of regular tree languages is robust, as illustrated by the following statement.

THEOREM 2. *The class of regular tree languages is a Boolean algebra [Rabin 1969; Emerson et al. 1993].*

A regular tree language is nonempty if and only if it contains a regular tree. Testing the emptiness of a regular tree language (defined by a given parity automaton) is in NP $\cap$ coNP.

2.3. Markov Decision Process

2.3.1. Perfect Information Setting. A ***probability distribution*** over a countable set X is a mapping $d : X \to [0, 1]$ such that $\sum_{x \in X} d(x) = 1$. In the sequel, we denote by $\mathcal{D}(X)$ the set of probability distributions over X. In this article, all probabilities will be rational numbers, which will be described in binary when dealing with encoding.

An ***arena*** is a tuple $\mathcal{G} = \langle S, s_{\text{ini}}, \Sigma, \zeta \rangle$ where S is a countable set of states, s_{ini} is an initial state, Σ is a finite set of ***actions,*** and $\zeta : S \times \Sigma \to \mathcal{D}(S)$ is the transition (total) function.

A play in such an arena proceeds as follows. It starts in state s_{ini} as Éloïse picks an action σ, and a successor state is chosen according to the probability distribution $\zeta(s_{\text{ini}}, \sigma)$. Then, Éloïse chooses a new action and the state is updated, and so on forever. Hence, a ***play*** is an infinite sequence $s_0 s_1 s_2 \cdots \in S^\omega$ such that $s_0 = s_{\text{ini}}$ and for every $i \geq 0$, there exists a $\sigma \in \Sigma$ with $\zeta(s_i, \sigma)(s_{i+1}) > 0$. In the sequel, we refer to a prefix of a play as a ***partial play,*** and we denote by Plays the set of all plays.

A ***(pure) strategy***[4] for Éloïse is a function $\varphi : S^* \to \Sigma$ assigning to every partial play an action. Of special interest are those strategies that do not require memory: a strategy φ is ***memoryless*** if $\varphi(\lambda \cdot s) = \varphi(\lambda' \cdot s)$ for all partial play λ, λ' and all states s (i.e., φ only depends on the current state). A play $\lambda = s_0 s_1 s_2 \cdots$ is ***consistent*** with a strategy φ if $\zeta(s_i, \varphi(v_0 \cdots v_i))(s_{i+1}) > 0$, for all $i \geq 0$.

Now, for any partial play λ, the ***cylinder*** for λ is the set $\text{Cyl}(\lambda) = \lambda S^\omega \cap \text{Plays}$. Let $\mathcal{F}_P$ be the σ-algebra generated by the set of cylinders. Then, $(\text{Plays}, \mathcal{F}_P)$ is a measurable space.

A strategy φ induces a probability space over $(\text{Plays}, \mathcal{F}_P)$ as follows: one defines a measure μ_φ on cylinders and then uniquely extends it to a probability measure on $\mathcal{F}_P$ using the Carathéodory's unique extension theorem. For this, we first define inductively μ_φ on cylinders:

- as all plays start from s_{ini}, we let $\mu_\varphi(\text{Cyl}(s_{\text{ini}})) = 1$;
- for any partial play λ ending in some state s, we let $\mu_\varphi(\text{Cyl}(\lambda \cdot s')) = \mu_\varphi(\text{Cyl}(\lambda)) \cdot \zeta(s, \varphi(\lambda))(s')$.

We also denote by μ_φ the unique extension of μ_φ to a probability measure on $\mathcal{F}$. Then, $(\text{Plays}, \mathcal{F}_P, \mu_\varphi)$ is a probability space.

An ***objective*** is a measurable set $\mathcal{O} \subseteq$ Plays: a play is winning if it belongs to $\mathcal{O}$. A ***Markov decision process (MDP, aka*** **one-and-half-player game*)*** is a pair $\mathbb{G} = (\mathcal{G}, \mathcal{O})$ where $\mathcal{G}$ is an arena and $\mathcal{O}$ is an objective. In the sequel, we should focus on ω-regular objectives (which are easily seen to be measurable) whose definitions are the same as for the acceptance condition on tree automata (the only differences is that we may have an infinite set of states and that we restrict ourselves to the set Plays).

A strategy φ is ***almost-surely winning*** (resp. ***positively winning***) if $\mu_\varphi(\mathcal{O}) = 1$ (*resp.* $\mu_\varphi(\mathcal{O}) > 0$). If such a strategy exists, we say that Éloïse ***almost-surely wins*** (*resp.* ***positively wins***) $\mathbb{G}$. The ***value*** of $\mathbb{G}$ is defined as $\text{Val}(\mathbb{G}) = \sup_\varphi \mu_\varphi(\mathcal{O})$, and a strategy φ is ***optimal*** if $\text{Val}(\mathbb{G}) = \mu_\varphi(\mathcal{O})$.

[4]We do not consider here randomized strategies because, in the setting of this article, they are useless. Note that for finite MDP, optimal strategies—when they exist—can always be chosen to be pure.

When the set of actions Σ is reduced to one element, the MDP $(\mathcal{G}, \mathcal{O})$ is called a ***Markov chain,*** and we omit the unique action from all the definitions. The set Plays is called the set of ***traces*** of the Markov chain and is denoted Traces. We write $\mu_{\mathcal{G}}$ as the probability measure associated with the unique strategy. We say that the Markov chain ***almost-surely*** fulfils its objective if $\mu_{\mathcal{G}}(\mathcal{O}) = 1$.

MDPs over finite arenas enjoy many good properties.

THEOREM 3. *Let $\mathbb{G}$ be an MDP over a finite arena with a parity objective. Then, one can decide in polynomial time whether Éloïse almost-surely (*resp. *positively) wins. Moreover, Éloïse always has an optimal memoryless strategy [Courcoubetis and Yannakakis 1990; Chatterjee et al. 2004].*

2.3.2. Imperfect Information Setting. Now we consider the case where Éloïse has imperfect information about the current state. For this, we consider an equivalence relation $\sim$ over S. We let $[s]_\sim$ be the equivalence class of s for $\sim$ and $S/_\sim$ be the set of equivalence classes of $\sim$ over S.

The intuitive meaning of $\sim$ is that two states $s_1 \sim s_2$ cannot be distinguished by Éloïse. We easily extend $\sim$ to partial plays: $s_0 s_1 \cdots s_n \sim s'_0 s'_1 \cdots s'_n$ if and only if $s_i \sim s'_i$ for all $i = 0, \ldots, n$. Because two equivalent plays $\lambda_1 \sim \lambda_2$ cannot be distinguished by Éloïse, she should therefore behave the same in both of them.

Hence, we should only consider so-called observation-based strategies. An ***observation-based (pure) strategy*** is a function $\varphi : (S/_\sim)^* \to \Sigma$, that is, to choose her next action, Éloïse considers the sequence of observations she has obtained so far[5]. In particular, an observation-based strategy φ is such that $\varphi(\lambda) = \varphi(\lambda')$ whenever $\lambda \sim \lambda'$. In this context, a ***memoryless strategy*** is a function from $S/_\sim \to \mathcal{D}(\Sigma)$, that is, it only depend on the current equivalence class.

A ***partial observation MDP (POMDP, aka*** **one-and-half-player imperfect information game*)*** is a triple $(\mathcal{G}, \sim, \mathcal{O})$ where $\mathcal{G}$ is an arena, $\sim$ is an equivalence relation over states, and $\mathcal{O}$ is an objective. We say that Éloïse ***almost-surely wins*** (resp. positively wins) $\mathbb{G}$ if she has an almost-surely (resp. positively) winning *observation-based* strategy. Finally, the ***value*** of $\mathbb{G}$ is defined as $\mathrm{Val}(\mathbb{G}) = \sup_\varphi \mu_\varphi(\mathcal{O})$ where φ ranges over observation-based strategies; optimality is defined as previously.

The following decidability results are known for POMDP:

THEOREM 4. *In a POMDP with a Büchi (*resp. *co-Büchi) objective, deciding whether Éloïse almost-surely (*resp. *positively) wins is ExpTime-complete. Moreover, if Éloïse has an almost-surely (*resp. *positively) winning strategy, she has an almost-surely (*resp. *positively) winning strategy with finite memory [Baier et al. 2008].*

*In a POMDP with a co-Büchi (*resp. *Büchi) objective, it is undecidable whether Éloïse almost-surely (*resp. *positively) wins.*

Remark 5. The results in Theorems 3 and 4 do not depend on the encoding of probability distributions because the only relevant information is to determine which probabilities are non-zero.

3. QUALITATIVE TREE LANGUAGES

3.1. Definitions

In the classical definition, a run of a tree automaton $\mathcal{A}$ is accepting if all its branches satisfy the acceptance condition. In this article, we introduce a more relaxed notion of acceptance: a run is qualitatively accepting if almost every (in the sense of the

[5]By abuse of notation, we shall write $\varphi(s_0 \cdots s_n)$ to mean $\varphi([s_0]_\sim \cdots [s_n]_\sim)$

measure μ) branch in it is accepting. More formally, consider a tree automaton $\mathcal{A}$ with an ω-regular acceptance condition Acc. A run ρ of $\mathcal{A}$ is ***qualitatively accepting*** if the set $\mathrm{AccBr}(\rho) = \{\pi \in \{0,1\}^\omega \mid \rho(\pi) \in \mathrm{Acc}\}$ has measure 1; that is, $\mu(\mathrm{AccBr}(\rho)) = 1$. Note that, thanks to Proposition 6, the set $\mathrm{AccBr}(\rho)$ is indeed measurable. A tree t is ***qualitatively accepted*** by $\mathcal{A}$ if there exists a qualitatively accepting run of $\mathcal{A}$ over t, and the set of all trees qualitatively accepted by $\mathcal{A}$ is denoted $L_{\mathrm{Qual}}(\mathcal{A})$. Finally, a *qualitative tree language* is a set L of trees such that there exists a parity automaton $\mathcal{A}$ such that $L_{\mathrm{Qual}}(\mathcal{A}) = L$.

PROPOSITION 6. *Let $\mathcal{A}$ be a tree automaton equipped with an ω-regular acceptance condition, and let ρ be a run of $\mathcal{A}$. The set* $\mathrm{AccBr}(\rho)$ *is measurable.*

PROOF. This is a simple ad hoc proof of Proposition 6. Nevertheless, one should notice that Corollary 37 (see Section 4.1.2) directly implies Proposition 6.

The reachability case (with final states F) is obvious because the set $\mathrm{AccBr}(\rho)$ is the (countable) union of all cones $\mathcal{C}(u)$ with $\rho(u) \in F$.

We focus now on the co-Büchi case (with forbidden states F). Recall that a branch is accepting if and only if there is a node after which there are only states in $Q\backslash F$.

We claim that

$$\mathrm{AccBr}(\rho) = \bigcup_{u \in \{0,1\}^*} \Big(\mathcal{C}(u) \setminus \bigcup_{u \sqsubseteq v,\, \rho(v) \in F} \mathcal{C}(v)\Big).$$

Let $b \in \mathrm{AccBr}(\rho)$ and let u be a node after which all node labels in b belong to $Q \setminus F$. Then, $b \in \mathcal{C}(u)$ and for all $v \sqsupseteq u$, $\rho(v) \notin F$: hence, b belongs to $\mathcal{C}(u) \setminus \bigcup_{u \sqsubseteq v, \rho(v)\in F} \mathcal{C}(v)$. Now, let $b \notin \mathrm{AccBr}(\rho)$: b contains infinitely many occurrences of forbidden states. Therefore, for all node u in b there is another node v below u on b that is labeled by a state in F, meaning that $b \in \bigcup_{u \sqsubseteq v, \rho(v)\in F} \mathcal{C}(v)$. Equivalently, $b \notin \mathcal{C}(u) \setminus \bigcup_{u \sqsubseteq v, \rho(v)\in F} \mathcal{C}(v)$. Therefore, the set $\mathrm{AccBr}(\rho)$ is measurable. The parity conditions follow from the previous case because one can express them as a Boolean combination of co-Büchi conditions. The general case of an arbitrary ω-regular condition is obtained as follows. First, one considers a *deterministic* parity word automaton recognizing Acc and then takes the synchronized product of this automaton with the tree automaton. This leads to a parity tree automaton that accepts the same language and whose runs are in bijection with the ones of the original automaton. Moreover, as this bijection preserves the set of accepting branches, one can conclude from the parity case. □

Example 7. Let $\mathcal{L}_{\mathtt{a}}$ be the language of $\{\mathtt{a},\mathtt{b}\}$-labeled trees whose set of branches containing at least one a has measure 1. This language is recognized by the following reachability deterministic automaton $\mathcal{A} = \langle \{\mathtt{a},\mathtt{b}\}, \{q_{\mathrm{ini}}, q_f\}, q_{\mathrm{ini}}, \Delta, \{q_f\}\rangle$ where: $\Delta = \{q_{\mathrm{ini}} \xrightarrow{\mathtt{b}} (q_{\mathrm{ini}}, q_{\mathrm{ini}}), q_{\mathrm{ini}} \xrightarrow{\mathtt{a}} (q_f, q_f), q_f \xrightarrow{\mathtt{a}} (q_f, q_f), q_f \xrightarrow{\mathtt{b}} (q_{\mathrm{ini}}, q_{\mathrm{ini}})\}$.

If one considers $\mathcal{A}$ as a Büchi automaton, the accepted language consists of those trees whose set of branches containing infinitely many a have a measure 1.

Example 8. Let $\mathcal{L}_1$ be the language of trees t such that, in almost every branch, there is a node u labeled by a such that the subtree $t[u]$ has only a on its leftmost branch. This language is recognized by the nondeterministic reachability automaton $\mathcal{A} = \langle A, Q, q_w, \Delta, \{q_{acc}\}\rangle$ with $A = \{\mathtt{a},\mathtt{b}\}$, $Q = \{q_w, q_l, q_{acc}, q_{rej}\}$, and Δ contains the following transitions: $q_w \xrightarrow{*} (q_w, q_w)$, $q_w \xrightarrow{\mathtt{a}} (q_l, q_{acc})$, $q_l \xrightarrow{\mathtt{a}} (q_l, q_{acc})$, $q_l \xrightarrow{\mathtt{b}} (q_{rej}, q_{rej})$, $q_{acc} \xrightarrow{*} (q_{acc}, q_{acc})$, $q_{rej} \xrightarrow{*} (q_{rej}, q_{rej})$ (here, $*$ is a shorthand for an arbitrary letter). Intuitively, the automaton can wait in state q_w as long as it wants. Using the second transition, the automaton can guess that the node u (labeled by a) has a leftmost branch containing only a. This assumption is checked by sending on the leftmost branch the state q_l and the accepting state q_{acc} on all other branches. As long as the nodes are labeled by a state, q_l is propagated to the left son. If all nodes on the leftmost branch starting at u are labeled by a, this branch will be rejecting—but this does not affect the measure because there are only countably many such branches). If a node v labeled by b is encountered in state $q_{l,}$ the nonaccepting state q_{rej} is propagated on all branches. This last scenario cannot occur in an accepting run because these cones of rejecting branches have a strictly positive measure. Hence, the automaton is penalized for wrong guesses.

For the same reasons as for regular tree languages (cf. Remark 1), the parity condition is expressive enough to capture any ω-regular conditions: for any automaton $\mathcal{A}$ with an ω-regular acceptance condition, there exists a parity automaton $\mathcal{B}$ such that $L_{\text{Qual}}(\mathcal{A}) = L_{\text{Qual}}(\mathcal{B})$.

Thanks to the following proposition, we can only focus on complete automata.

PROPOSITION 9. *For any tree automaton $\mathcal{A}$ with an ω-regular acceptance condition, there exists a complete tree automaton $\mathcal{B}$ with the same acceptance condition and such that $L_{\text{Qual}}(\mathcal{A}) = L_{\text{Qual}}(\mathcal{B})$.*

PROOF. Let $\mathcal{A} = \langle A, Q, q_{\text{ini}}, \Delta, \text{Acc}\rangle$ be a possibly incomplete automaton. Define a complete automaton $\mathcal{B} = \langle A, Q' = Q \uplus \{q_{\text{rej}}\}, q_{\text{ini}}, \Delta', \text{Acc}\rangle$, where the set Δ' is Δ augmented with $\{q \xrightarrow{a} (q_{\text{rej}}, q_{\text{rej}}) \mid \nexists\, q \xrightarrow{a} (q_0, q_1) \in \Delta\} \cup \{q_{\text{rej}} \xrightarrow{a} (q_{\text{rej}}, q_{\text{rej}}) \mid a \in A\}$; that is, we add a transition to the sink state q_{rej} whenever a transition is missing.

Note that because the acceptance condition is unchanged, a branch going through q_{rej} is rejecting. In particular, because q_{rej} is a sink state, a run that contains q_{rej} is a rejecting cone and thus is rejecting (because the measure of the rejecting cone is strictly positive). Therefore, the only accepting runs of $\mathcal{B}$ are actually runs that only uses transitions in $\mathcal{A}$, and hence are runs of $\mathcal{A}$. Therefore, $L_{\text{Qual}}(\mathcal{A}) = L_{\text{Qual}}(\mathcal{B})$. □

Unsurprisingly, determinism is a restriction.

PROPOSITION 10. *There is a qualitative tree language that cannot be qualitatively accepted by any deterministic automaton.*

PROOF. Let $\mathcal{L}_{\mathtt{a}}$ be the qualitative language (see Example 7) of $\{\mathtt{a}, \mathtt{b}\}$-labeled trees whose set of branches containing at least one a has a measure 1. Consider now the language $\mathcal{L}'_{\mathtt{a}} = \{t \mid t[0] \in \mathcal{L}_{\mathtt{a}} \text{ or } t[1] \in \mathcal{L}_{\mathtt{a}}\}$. Clearly, $\mathcal{L}'_{\mathtt{a}}$ is qualitative (it suffices to guess one subtree and check that it belongs to $\mathcal{L}_{\mathtt{a}}$ while accepting without any further consideration the other subtree). By contradiction, assume that there is a deterministic automaton $\mathcal{A}$ that qualitatively accepts $\mathcal{L}'_{\mathtt{a}}$. Indeed, consider the two trees $t_{\mathtt{a},\mathtt{b}}$ and $t_{\mathtt{b},\mathtt{a}}$ defined as follows: $t_{\mathtt{a},\mathtt{b}}(\varepsilon) = t_{\mathtt{b},\mathtt{a}}(\varepsilon) = \mathtt{b}$; the left subtree of the root of $t_{\mathtt{a},\mathtt{b}}$ (*resp.* $t_{\mathtt{b},\mathtt{a}}$) has only a's (*resp.* b's) and the right subtree of the root of $t_{\mathtt{a},\mathtt{b}}$ (*resp.* $t_{\mathtt{b},\mathtt{a}}$) has only b's (*resp.* a's). Both $t_{\mathtt{a},\mathtt{b}}$ and $t_{\mathtt{b},\mathtt{a}}$ belongs to $\mathcal{L}'_{\mathtt{a}}$. The run of $\mathcal{A}'$ on $t_{\mathtt{a},\mathtt{b}}$ starts with the same transition at the root, and therefore one can combine them and get a run for the tree $t_{\mathtt{b}}$ whose nodes are all labeled by b, which leads to a contradiction because $t_{\mathtt{b}} \notin \mathcal{L}'_{\mathtt{a}}$. □

3.2. On the Choice of Measure μ

The choice of the measure μ, although natural, is arbitrary. Considering the measure μ_p for some $0 < p < \frac{1}{2}$ would not affect the results obtained in this article (provided that definitions of the games are modified accordingly). However, note that changing the measure does change the accepted language for a given automaton.

PROPOSITION 11. *Let $0 < p < q < 1$ be two reals. Let $\mathcal{A}$ be the (deterministic and complete) automaton of Example 7. Then, there is a tree t such that $\mu_p(Acc(\mathcal{A}, t)) = 0$*

and $\mu_q(Acc(\mathcal{A},t)) = 1$, where $Acc(\mathcal{A},t)$ denotes the set of accepting branches for the unique run of $\mathcal{A}$ over t.

Proof. First recall that $\mathcal{A}$ accepts those trees whose set of branches containing infinitely many a has a measure 1.

We define the $\{\mathtt{a},\mathtt{b}\}$-labeled tree t by letting for all $u \in \{0,1\}^*$ $t(u) = \mathtt{a}$ if and only if $\mu_p(\mathrm{Cone}(u)) \leq \mu_q(\mathrm{Cone}(u))$.

Let ε be the maximum among $p, q, 1-p, 1-q$.

Following the proof of Kakutani [1948] as presented in Muchnik et al. [1998] and Bienvenu [2008], we consider the measure μ_r for $r = \frac{p+q}{2}$.

We first establish that for all $w \in \{0,1\}^*$,

$$\mu_r(\mathrm{Cone}(w))^2 \geq \mu_p(\mathrm{Cone}(w))\mu_q(\mathrm{Cone}(w))\alpha^{|w|}$$

where $\alpha = 1 + \frac{(p-q)^2}{4\varepsilon^2}$ (note that $\alpha > 1$). Indeed,

$$\begin{aligned}
\mu_r(\mathrm{Cone}(w))^2 &= \left(\frac{(p+q)^2}{4}\right)^{|w|_0} \left(\frac{((1-p)+(1-q))^2}{4}\right)^{|w|_1} \\
&= p^{|w|_0}q^{|w|_0}\left(1+\frac{(p-q)^2}{4pq}\right)^{|w|_0}(1-p)^{|w|_1} \\
&\quad (1-q)^{|w|_1}\left(1+\frac{((1-p)-(1-q))^2}{4(1-p)(1-q)}\right)^{|w|_1} \\
&= p^{|w|_0}q^{|w|_0}\left(1+\frac{(p-q)^2}{4pq}\right)^{|w|_0}(1-p)^{|w|_1} \\
&\quad (1-q)^{|w|_1}\left(1+\frac{(p-q)^2}{4(1-p)(1-q)}\right)^{|w|_1} \\
&\geq p^{|w|_0}q^{|w|_0}\alpha^{|w|_0}(1-p)^{|w|_1}(1-q)^{|w|_1}\alpha^{|w|_1} \\
&= \mu_p(\mathrm{Cone}(w))\mu_q(\mathrm{Cone}(w))\alpha^{|w|}
\end{aligned}$$

Let K_n be the set of words $w \in \{0,1\}^n$ such that $\mu_p(\mathrm{Cone}(w)) \leq \mu_q(\mathrm{Cone}(w))$, and let C_n be the disjoint union of all $\mathrm{Cone}(w)$ for all $w \in K_n$.

For all $w \in K_n$, we have $\mu_r(\mathrm{Cone}(w)) \geq \mu_p(\mathrm{Cone}(w))\alpha^{n/2}$, and for all $w \notin K_n$ we have $\mu_r(\mathrm{Cone}(w)) \geq \mu_q(\mathrm{Cone}(w))\alpha^{n/2}$.

Hence, $\begin{aligned}[t] \mu_p(C_n) &= \textstyle\sum_{w\in K_n}\mu_p(\mathrm{Cone}(w)) \\ &\leq \alpha^{-n/2}\textstyle\sum_{w\in K_n}\mu_r(\mathrm{Cone}(w)) \\ &\leq \alpha^{-n/2}\end{aligned}$

Similarly, $\begin{aligned}[t] \mu_q(C_n) &= 1-\mu_q(\overline{C_n}) \\ &= 1-\textstyle\sum_{w\notin K_n,\, w\in\{0,1\}^n}\mu_q(\mathrm{Cone}(w)) \\ &\geq 1-\alpha^{-n/2}\textstyle\sum_{w\notin K_n,\, w\in\{0,1\}^n}\mu_r(\mathrm{Cone}(w)) \\ &\geq 1-\alpha^{-n/2}\end{aligned}$

By definition of $\mathcal{A}$, we have

$$\mathrm{Acc}(\mathcal{A},t) = \bigcap_{k\geq 0}\bigcup_{n\geq k} C_n.$$

For $k \geq 1$, we let D_k denote the set $\bigcup_{n\geq k} C_n$.

As for all $k \geq 1$, $\mu_p(D_k) \leq \sum_{n\geq k}\alpha^{-\frac{n}{2}} \leq \frac{\alpha^{\frac{-k}{2}}}{1-\alpha^{\frac{-1}{2}}}$, we have $\lim_{k\to\infty}\mu_p(D_k) = 0$ (recall that $\alpha > 1$). Moreover, for all $k \geq 0$, $\mu_q(D_k) = 1$ (as $\alpha > 1$ and as for all $n \geq k$, $\mu_q(D_k) \geq \mu_q(C_n) \geq 1-\alpha^{-n/2}$).

As the sequence of sets $(D_k)_{k\geq 0}$ is decreasing for the inclusion, we can thank Bauer [2001, Theorem 3.2, p. 10] that:

$$\begin{cases}\mu_p(\mathrm{Acc}(\mathcal{A},t)) \;=\; \lim_{k\to\infty}\mu_p(D_k)=0\\ \mu_q(\mathrm{Acc}(\mathcal{A},t)) \;=\; \lim_{k\to\infty}\mu_q(D_k)=1\end{cases}\qquad\square$$

A more general definition is to associate with any letter a in the alphabet a pair $(p^0_{\mathrm{a}}, p^1_{\mathrm{a}}) \in [0,1]^2$ with $p^0_{\mathrm{a}} + p^1_{\mathrm{a}} = 1$ and then to define the measure of a cone in a tree t by letting $\mu(\mathrm{Cone}(u_1\cdots u_n)) = p^{u_1}_{t(\varepsilon)} p^{u_2}_{t(u_1)} \cdots p^{u_n}_{t(u_1\cdots u_{n-1})}$. Intuitively, the node label determines the respective weights of the left and right sons in the definition of the measure. In particular, the measure μ_p is the one obtained by letting $(p^0_{\mathrm{x}}, p^1_{\mathrm{x}}) = (p, 1-p)$ for all letters x in the alphabet.

Again, with such a measure, the results obtained in this article (provided that definitions of the games are modified accordingly) remain correct.

Remark 12. Following Proposition 11, a natural question is whether the class of qualitative tree languages is the same for each distribution μ_p with $0 < p < 1$ and the same for the above variant with probability values p^0_a, p^1_a for all letters in the alphabet. From the statement of Proposition 11, we conjecture that the answer to these questions is no, but we leave it open.

3.3. Pumping Lemma

Let t be a tree and $u \in \{0,1\}^*$ be a node. A pair $\Delta = (t, u)$ is called a ***pointed tree***. With a pointed tree $\Delta_1 = (t_1, u_1)$ and a tree t_2, we associate a new tree, $\Delta_1 \cdot t_2$, by plugging t_2 into t_1 instead of the subtree rooted at u_1. Formally, $\Delta_1 \cdot t_2(u) = t_1(u)$ if u_1 is not a prefix of u and $\Delta_1 \cdot t_2(u) = t_2(u')$ if $u = u_1u'$ for some $u' \in \{0,1\}^*$. We can also define the product of two pointed trees $\Delta_1 = (t_1, u_1)$ and $\Delta_2 = (t_2, u_2)$ by letting $\Delta_1 \cdot \Delta_2 = (\Delta_1 \cdot t_2, u_1 \cdot u_2)$. Finally, with a pointed tree $\Delta = (t, u)$, we associate a tree Δ^ω by taking an ω-iteration of the product: $\Delta^\omega(v) = t(v')$ where v' is the shortest word s.t. $v = u^k v'$ for some $k \geq 0$.

Qualitative tree languages enjoy a pumping lemma (see Figure 13 for an illustration), which contrasts with regular tree languages. Intuitively, pumping does not change acceptance for the qualitative semantics because it may only introduce a set of rejecting branches of measure zero (whereas for the classical semantics introducing a rejecting branch would make the run rejecting).

LEMMA 13. *Let $\mathcal{A}$ be an n-states parity automaton, t be a tree in $L_{\mathrm{Qual}}(\mathcal{A})$, and u be a node of depth greater than n. Then there exists three pointed trees Δ_1, Δ_2, and Δ_3 such that $t = \Delta_1 \cdot \Delta_2 \cdot \Delta_3 \cdot t[u]$ and $\Delta_1 \cdot \Delta_2^\omega \in L_{\mathrm{Qual}}(\mathcal{A})$.*

PROOF. Let ρ be an accepting run of $\mathcal{A}$ over t. Since u has depth greater than the number of states, there are two nodes u_1 and u_2 such that $u_1 \sqsubseteq u_2 \sqsubseteq u$ and $\rho(u_1) = \rho(u_2) = q$ for some state q. We now decompose t as $t = \Delta_1 \cdot \Delta_2 \cdot \Delta_3 \cdot t[u]$ with $\Delta_1 = (t, u_1)$, $\Delta_2 = (t[u_1], u_2)$ and $\Delta_3 = (t[u_2], u)$. We decompose ρ in a similar way, by letting $\rho = \Theta_1 \cdot \Theta_2 \cdot \Theta_3 \cdot \rho[u]$ with $\Theta_1 = (\rho, u_1)$, $\Theta_2 = (\rho[u_1], u_2)$ and $\Theta_3 = (\rho[u_2], u)$. Note that $\Theta_1 \cdot \Theta_2^\omega$ is a run of $\mathcal{A}$ over $\Delta_1 \cdot \Delta_2^\omega$. A branch in $\Theta_1 \cdot \Theta_2^\omega$ can be rejecting for three reasons: it is a rejecting branch in $\rho \setminus \rho[u_1]$; or it is a branch in some of the copies of $\rho[u_1] \setminus \rho[u_2]$; or it is the newly created branch $u_1 v^\omega$ where v is the word such that $u_2 = u_1 \cdot v$. Because ρ is accepting, the measure of each of these three sets of rejecting branches is 0 (for the second set, it is a finite union of sets of measure 0). Therefore $\Theta_1 \cdot \Theta_2^\omega$ is accepting, hence $\Delta_1 \cdot \Delta_2^\omega \in L_{\mathrm{Qual}}(\mathcal{A})$. $\square$

3.4. Closure Properties

We now investigate the closure properties of qualitative tree languages under Boolean operations.

PROPOSITION 14. *Qualitative tree languages are closed under union and intersection.*

PROOF. The union is free thanks to the nondeterminism of the automata model. For the intersection of two languages, $L_{\mathrm{Qual}}(\mathcal{A}_1)$ and $L_{\mathrm{Qual}}(\mathcal{A}_2)$, we consider the usual product automaton whose states set is the Cartesian product of the states set of $\mathcal{A}_1$ and of $\mathcal{A}_2$ and that simulates $\mathcal{A}_1$ (*resp.* $\mathcal{A}_2$) on its first (*resp.* second) component. Hence, runs of the product automaton are in bijection with pairs of runs of $\mathcal{A}_1$ and $\mathcal{A}_2$. The (ω-regular) acceptance condition requires that the acceptance condition of $\mathcal{A}_1$ (*resp.* $\mathcal{A}_2$) is fulfilled on the first (*resp.* second) component. Hence, the set of accepting branches in a run of the product automaton is the intersection of the set of accepting branches in the corresponding runs of $\mathcal{A}_1$ and $\mathcal{A}_2$. Hence, it has measure 1 if and only if both sets have measure 1; that is, a run is accepting if and only if both corresponding runs in $\mathcal{A}_1$ and $\mathcal{A}_2$ are accepting. □

Unsurprisingly, qualitative tree languages are not closed under complement. This is a simple consequence of the pumping lemma.

PROPOSITION 15. *Qualitative tree languages are not closed under complement.*

PROOF. Let $\mathcal{L}_{\mathtt{a}}$ be the set of $\{\mathtt{a},\mathtt{b}\}$-labeled trees whose set of branches containing an $\mathtt{a}$ has measure 1. Clearly, $\mathcal{L}_{\mathtt{a}}$ is a qualitative tree language. However, we will show that its complement $\overline{\mathcal{L}_{\mathtt{a}}}$ does not satisfy the pumping lemma, hence is not qualitative. By contradiction, assume that $\overline{\mathcal{L}_{\mathtt{a}}} = L_{\mathrm{Qual}}(\mathcal{A})$ for some n-state automaton $\mathcal{A}$. Now define a tree t by letting $u = 0^n$ and $t(v) = \mathtt{b}$ if $v \in t[u]$ and $t(v) = \mathtt{a}$ otherwise. Hence, in t a tree is labeled by a $\mathtt{b}$ if and only if it belongs to the subtree rooted at u, and therefore the branches in $\mathrm{Cone}(u)$ contain finitely many $\mathtt{a}$, implying that $t \in \overline{\mathcal{L}_{\mathtt{a}}}$. Now, if one uses Lemma 13 for t and u, the tree $\Delta_1 \cdot \Delta_2^{\omega}$ (as in the statement of Lemma 13) is the tree whose nodes are all labeled by $\mathtt{a}$; that is, $\Delta_1 \cdot \Delta_2^{\omega} \notin \overline{\mathcal{L}_{\mathtt{a}}}$, leading a contradiction. □

3.5. Emptiness Problem

It is well known that tree automata (as acceptors of regular languages) and two-player (perfect information) game are closely related [Gurevich and Harrington 1982; Grädel et al. 2002]. In particular, the emptiness problem for regular tree languages and the problem of deciding the winner in a parity game on a finite graph are polynomially equivalent. From the proof of this result also follows that a regular tree language is nonempty if and only if it contains a regular tree.

We show that a similar connection exists between tree automata as acceptors of qualitative tree languages and MDP. For this, fix a parity tree automaton $\mathcal{A} = \langle A, Q, q_{\mathrm{ini}}, \Delta, \mathrm{Col}\rangle$ and a tree t. Consider the arena, depicted in Figure 2, $\mathcal{G}_{\mathcal{A},t} = \langle S, s_{\mathrm{ini}}, \Sigma, \zeta\rangle$ where $S = Q \times \{0,1\}^* \cup \{\perp\}$, $s_{\mathrm{ini}} = (q_{\mathrm{ini}}, \varepsilon)$, $\Sigma = \Delta$ and ζ is defined as follows. First, we let $d_\perp$ be the distribution defined by $d_\perp(s) = 1$ if $s = \perp$ and $d_\perp(s) = 0$ otherwise, and, for all $q_0, q_1 \in Q$, and $u \in \{0,1\}^*$, we let $d_{q_0,q_1,u}$ be the distribution such that $d_{q_0,q_1,u}(q_0, u0) = d_{q_0,q_1,u}(q_1, u1) = 1/2$ and $d_{q_0,q_1,u}(s) = 0$ for all other $s \in S$. Then we let $\zeta((q,u),(q',a,q_0,q_1)) = d_\perp$ if $q \neq q'$ or $a \neq t(u)$, $\zeta((q,u),(q,t(u),q_0,q_1)) = d_{q_0,q_1,u}$ and $\zeta(\perp,\sigma) = d_\perp$ for all $\sigma \in \Delta$. Finally, we define a coloring function ρ by letting $\rho((q,u)) = \mathrm{Col}(q)$ and $\rho(\perp) = 1$, and we call $\mathbb{G}_{\mathcal{A},t} = (\mathcal{G}_{\mathcal{A},t}, \mathcal{O}_\rho)$ the MDP equipped with the parity objective $\mathcal{O}_\rho$ defined by ρ.

Then, the following holds:

THEOREM 16. *The tree t belongs to $L_{\mathrm{Qual}}(\mathcal{A})$ if and only if Éloïse almost-surely wins in $\mathbb{G}_{\mathcal{A},t}$.*

PROOF. In $\mathbb{G}_{\mathcal{A},t}$, a partial play that does not visit $\perp$ is a sequence of the form $(q_0,u_0)(q_1,u_1)\cdots(q_k,u_k)$ where for all i, $u_{i+1} = u_i x_i$ and for some $x_i = 0, 1$. Now, with

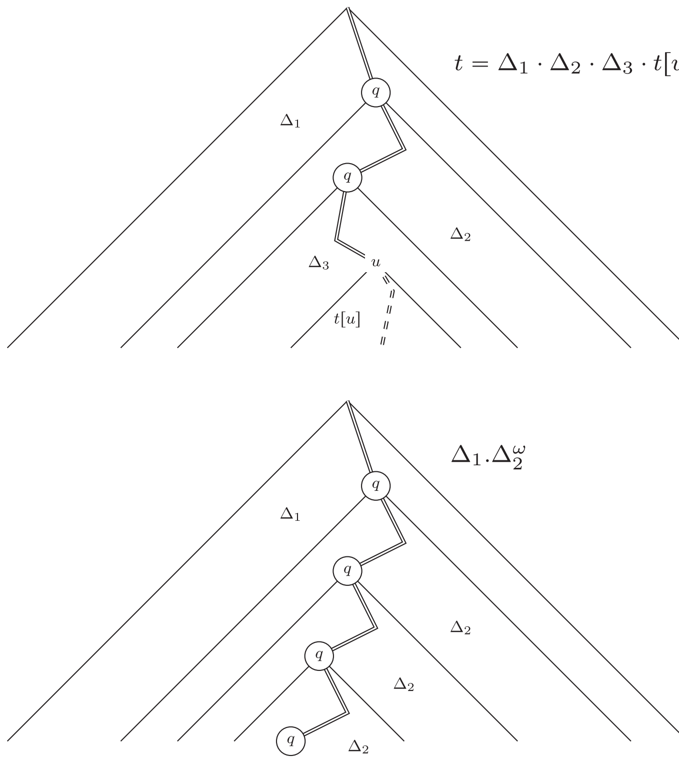

Fig. 1. Pumping Lemma.

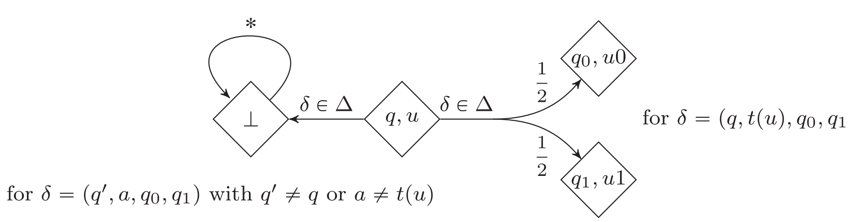

Fig. 2. The arena $\mathcal{G}_{\mathcal{A},t}$ of the acceptance game.

any strategy φ of Éloïse we can associate a strategy φ' of Éloïse that is defined only on those partial plays that are compatible with it (meaning that if in some partial play λ Éloïse does not respect φ' then $\varphi'(\lambda)$ is undefined) and coincide with φ when defined; it is easily seen that φ and φ' lead the same plays and define the same probability measure. This means that one can only focus on such strategies for Éloïse. We should also only focus on strategies that never reach $\perp$; that is, it always chooses an action of the form $(q, t(u), q_0, q_1)$ if the current state is (q, u). Now, one can remark that for any node u there is a unique partial play λ where Éloïse respects φ' and that ends in a state of the form (q, u) for some q such that φ' is defined on λ. Therefore, an equivalent way to see φ' is as a $\boldsymbol{Q}$-labeled tree that should additionally verify the local properties imposed by Δ; namely, the tree should be a run of $\mathcal{A}$ over t. It should also be clear that the set of plays where Éloïse respects φ' is the set of branches in the run associated with φ' and that this map preserves the measure (in particular the set of winning plays

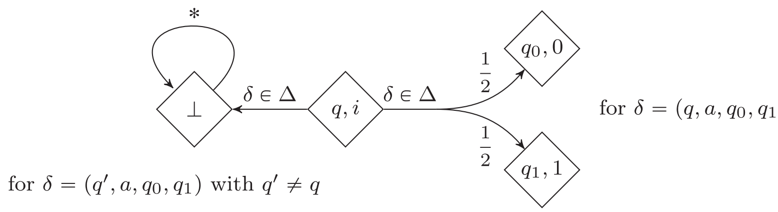

Fig. 3. The arena $\mathcal{G}_{\mathcal{A}}$ of the emptiness game.

has the same measure as the set of accepting branches in the run), meaning that a strategy with value ξ is mapped to a run whose set of accepting branches has measure ξ and vice-versa. Because this map from strategies to runs is bijective, this concludes the proof: a tree t belongs to $L_{\mathrm{Qual}}(\mathcal{A})$ if and only if there is an accepting run of $\mathcal{A}$ over t if and only if there is an almost-surely winning strategy for Éloïse in $\mathbb{G}_{\mathcal{A},t}$. □

Consider the (finite) arena $\mathcal{G}_{\mathcal{A}} = \langle S, s_{\mathrm{ini}}, \Sigma, \zeta \rangle$, depicted in Figure 3, where $S = Q \times \{0, 1\} \cup \{q_{\mathrm{ini}}, \perp\}$, $s_{\mathrm{ini}} = q_{\mathrm{ini}}$, $\Sigma = \Delta$, and ζ is defined as follows. First, we let $d_\perp$ be the distribution defined by $d_\perp(s) = 1$ if $s = \perp$ and $d_\perp(s) = 0$ otherwise, and, for all $q_0, q_1 \in Q$, we let d_{q_0,q_1} be the distribution such that $d_{q_0,q_1}((q_0, 0)) = d_{q_0,q_1}((q_1, 1)) = 1/2$ and $d_{q_0,q_1}(s) = 0$ for all other $s \in S$. Then we let $\zeta((q, i), (q, a, q_0, q_1)) = d_{q_0,q_1}$, $\zeta((q, i), (q', a, q_0, q_1)) = d_\perp$ if $q \neq q'$, $\zeta(q_{\mathrm{ini}}, (q_{\mathrm{ini}}, a, q_0, q_1)) = d_{q_0,q_1}$, $\zeta(q_{\mathrm{ini}}, (q, a, q_0, q_1)) = d_\perp$ if $q \neq q_{\mathrm{ini}}$, and $\zeta(\perp, \sigma) = d_\perp$ for all $\sigma \in \Delta$. Finally, we define a coloring function ρ by letting $\rho((q, i)) = \mathrm{Col}(q)$ and $\rho(\perp) = 1$, and we call $\mathbb{G}_{\mathcal{A}} = (\mathcal{G}_{\mathcal{A}}, \mathcal{O}_\rho)$ the MDP equipped with the parity objective $\mathcal{O}_\rho$ defined by ρ. Then, the following hold:

THEOREM 17. *The language $L_{\mathrm{Qual}}(\mathcal{A})$ is nonempty if and only if Éloïse almost-surely wins in $\mathbb{G}_{\mathcal{A}}$ from q_{ini}.*

PROOF. Following the same lines as for the proof of Theorem 16, it is easily seen that strategies that avoid state $\perp$ are in bijection with pairs of (t, ρ) where t is an A-labeled tree and ρ is a run of $\mathcal{A}$ over t. Moreover, this map preserves the measure, meaning that a strategy with value ξ is mapped to a pair (t, ρ) such that the set of accepting branches in ρ has measure ξ. In particular, Éloïse almost surely wins in $\mathbb{G}_{\mathcal{A}}$ if and only if there is a tree t that is qualitatively accepted by $\mathcal{A}$. □

COROLLARY 18. *Let $\mathcal{A}$ be a parity tree automaton. Then one can decide whether $L_{\mathrm{Qual}}(\mathcal{A}) = \emptyset$ in polynomial time. Moreover, if $L_{\mathrm{Qual}}(\mathcal{A}) \neq \emptyset$, it contains a regular tree, and such a tree can be constructed in polynomial time.*

PROOF. Emptiness in polynomial time follows from Theorems 17 and 3. Now, if Éloïse has an almost-surely strategy, then she has a memoryless one (Theorem 3). Then the tree and the run associated with this strategy are regular, and the run is qualitatively accepting. As one can compute (when exists) an almost-surely winning positional strategy, one can also compute (when exists) in polynomial time a regular tree in $L_{\mathrm{Qual}}(\mathcal{A})$. □

Remark 19. Motivated by a decision problem for qualitative tree language, we designed a polynomial reduction of the emptiness problem to the problem of deciding almost-surely winning in a finite MDP.

As already mentioned, a similar connection exists between tree automata (as acceptors of regular tree languages) and two-player (perfect information) games (see, e.g., Gurevich and Harrington [1982] and Grädel et al. [2002]). Indeed, the emptiness problem for regular tree languages and the problem of deciding the winner in a parity game on a finite graph are polynomially equivalent.

Hence, one may ask whether, conversely, the problem of deciding almost-surely winning in a finite MDP can be polynomially reduced to the emptiness problem for qualitative tree languages. It is indeed possible, and the proof is very similar to the one from a two-player game to a regular tree language. We briefly sketch the proof here.

First note that one can, up to coding, restrict its attention to finite MDP with only two actions, such that from any state and any action there are always two possible successors that can both be reached with same probability 1/2. Then, one designs a deterministic tree automaton whose states are identified with the ones of the MDP, whose input alphabet is identified with the set of actions of the MDP, and whose transition function mimics the one of the MDP. Then one concludes by noting that there is a bijection between trees and strategies and that this bijection is such that the measure of the accepting branches in the (unique) run of the automaton on a tree equals the value of the corresponding strategy in the MDP.

3.6. Regular Tree Languages and Qualitative Tree Languages Are Incomparable

In this section, we prove that regular tree languages and qualitative tree languages are incomparable.

PROPOSITION 20. *There is a regular tree language that is not qualitative.*

PROOF. Consider the regular tree language L of those $\{\mathtt{a}, \mathtt{b}\}$-labeled trees that contain at least one node labeled by $\mathtt{b}$. By contradiction, assume that $L = L_{\text{Qual}}(\mathcal{A})$ for some n-state automaton $\mathcal{A}$ and let $t \in L$ be the tree defined by $t(u) = \mathtt{b}$ if $u = 0^n$ and $t(u) = \mathtt{a}$ otherwise. Apply Lemma 3.3 to t and u: the tree $\Delta_1 \cdot \Delta_2^\omega$ (using the notations of Lemma 13) is the tree whose nodes are all labeled by $\mathtt{a}$; thus $\Delta_1 \cdot \Delta_2^\omega \notin L$, leading to a contradiction. □

THEOREM 21. *There is a qualitative tree language that is not regular.*

PROOF. Let $\mathcal{L}_\mathtt{a}$ be the language of trees whose sets of branches containing at least one $\mathtt{a}$ have a measure of 1. This language is qualitative, as noticed in Example 7. In the sequel, we prove that $\mathcal{L}_\mathtt{a}$ is not regular.

We first prove that, for any regular tree t, if there is no cone in t whose branches only contain the letter $\mathtt{b}$, then $t \in \mathcal{L}_\mathtt{a}$. Let t be a regular tree, we can assume w.l.o.g. that if there is a node labeled by $\mathtt{a}$, then all its descendants are labeled by $\mathtt{a}$. Then the property "there is no cone in t whose branches only contain $\mathtt{b}$" is the same as "every subtree contains a subtree made only of $\mathtt{a}$." Let $X_1, \dots, X_n$ be the n different subtrees of t, and for all i, let μ_i be the measure of the set of branches containing $\mathtt{a}$ in X_i (we call it the value of X_i). We can assume w.l.o.g. that $\forall i\ \mu_1 \leq \mu_i$. If X_{i_1} and X_{i_2} are the two sons of X_1, we know that $\mu_1 = \frac{\mu_{i_1}+\mu_{i_2}}{2}$. Since $\mu_1 \leq \mu_i$ for $i = i_1, i_2$, $\mu_{i_1} = \mu_{i_2} = \mu_1$. Hence, we can prove by induction that for all X_i of minimal value, all the subtrees of X_i have minimal value, too. Since there is a subtree of $\mathtt{a}$ (of value 1) in X_1, $\mu_1 = 1$ hence, for all i, $\mu_i = 1$, hence the value of t is 1, hence $t \in \mathcal{L}_\mathtt{a}$.

We assume by contradiction that $\mathcal{L}_\mathtt{a}$ is regular. The closure properties of regular tree languages implies that the following language $\mathcal{L}$ is also regular:

$$\mathcal{L} = \left\{t \,\middle|\, t \notin \mathcal{L}_\mathtt{a} \wedge \text{“}\textit{there is no cone in } t \textit{ whose branches only contain } \mathtt{b}\text{”}\right\}$$

Using our previous characterization of regular trees in $\mathcal{L}_\mathtt{a}$, it follows that $\mathcal{L}$ does not contain any regular tree, hence $\mathcal{L}$ is empty (Theorem 2). Then, to raise a contradiction, we build a (non+regular) tree $t_0 \in \mathcal{L}$.

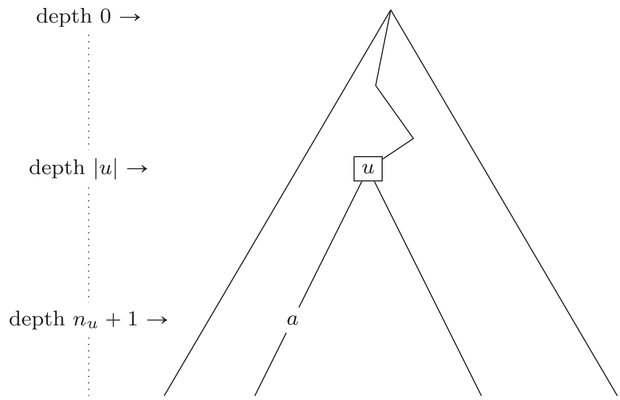

Fig. 4. The tree t_0.

For every node $u \in \{0, 1\}^*$, we let n_u be the integer whose binary representation is $1 \cdot u$. We define a tree t_0 as follows: let $v \in \{0, 1\}^*$; if there exists some u such that $v = u.0^{n_u+1-|u|}$ then $t_0(v) = \mathtt{a}$, otherwise $t_0(v) = \mathtt{b}$ (see Figure 4). We now establish that $t_0 \notin \mathcal{L}_\mathtt{a}$. First note that the set of branches in t_0 that contains at least one a is obtained by taking the union of those cones $\mathcal{C}(u)$ such that $t_0(u) = \mathtt{a}$. Then remark that, for every level ℓ, there is one and only one node u of depth ℓ labeled by a (except for $\ell = 0, 1$ where there are no such u). Thus, we can bound the measure μ of the set of branches in t_0 that contains at least one a:

$$\mu \leq \sum_{\ell=2}^{+\infty} 2^{-\ell} = \frac{1}{2}.$$

This proves that $t_0 \notin \mathcal{L}_\mathtt{a}$. Moreover, it follows from the definition, that for every node u, there is a branch (the leftmost one) in the cone $\mathcal{C}(u)$ that contains an a, hence $t_0 \in \mathcal{L}$, which contradicts the fact that $\mathcal{L}$ is empty. □

3.7. The Value of a Tree May Not Be Reached

So far, we have defined qualitative acceptance of a tree by the existence of a run whose set of accepting branches has measure 1. We can refine this notion by defining the ***value*** of a tree as follows. For a tree automaton $\mathcal{A}$, and a tree t, we let

$$\mathrm{Val}_\mathcal{A}(t) = \sup_{\rho_t \text{ run of } \mathcal{A} \text{ over } t} \mu(\mathrm{AccBr}(\rho_t)).$$

In particular, $L_{\mathrm{Qual}}(\mathcal{A})$ is the set of trees t whose value is 1 and is reached for some run (i.e., the sup is a max). The following result proves that the value may not be reached by some run.

THEOREM 22. *There is a reachability automaton $\mathcal{A}$ and a tree t such that* $\mathrm{Val}_\mathcal{A}(t) = 1$ *but $t \notin L_{\mathrm{Qual}}(\mathcal{A})$.*

PROOF. Let $\mathcal{A} = \langle A, Q, q_w, \Delta, \{q_{acc}\}\rangle$ with $A = \{\mathtt{a}, \mathtt{b}\}$, $Q = \{q_w, q_l, q_{acc}, q_{rej}\}$, and Δ contains the following transitions: $q_w \xrightarrow{*} (q_w, q_w)$, $q_w \xrightarrow{*} (q_l, q_{acc})$, $q_l \xrightarrow{\mathtt{a}} (q_{acc}, q_l)$, $q_l \xrightarrow{\mathtt{b}} (q_{rej}, q_{rej})$ (here $*$ is a shorthand for an arbitrary letter). Intuitively, the automaton can wait in state q_w as long as it wants. It can at some point use the second transition: this leads it to accept (all branches in) the subtree rooted at the left son, as well as those subtrees rooted in the branch going left and then right forever, as long as this branch does not contain a node labeled $\mathtt{b}$, in which case the subtree rooted at this node is rejected.

Define $h : \{0,1\}^* \to \{0,1\}^*$ by letting $h(u) = u.0.1^{|u|}$, for all $u \in \{0,1\}^*$. Now, consider the tree t defined by $t(u) = \mathtt{b}$ if u belongs to $h(\{0,1\}^*)$—the image of the set $\{0,1\}^*$ by h—and $t(u) = \mathtt{a}$ otherwise.

We claim that any run ρ_t of $\mathcal{A}$ on t is such that $\mu(\mathrm{AccBr}(\rho_t)) < 1$. Indeed, either the only transition used in ρ_t is $q_w \xrightarrow{*} (q_w, q_w)$, and therefore all branches are rejecting. Otherwise, the transition $q_w \xrightarrow{*} (q_l, q_{acc})$ is used at least once. Then pick u a node of minimal depth where this transition is used: the cone $\mathrm{Cone}(h(u))$ is rejecting, and therefore $\mu(\mathrm{AccBr}(\rho_t)) \leq 1 - 2^{|h(u)|} < 1$.

Now, for all integer $i \geq 1$, consider the run ρ_t^i where the transition $q_w \xrightarrow{*} (q_w, q_w)$ is used for all nodes at depth $<i$, and the transition $q_w \xrightarrow{*} (q_l, q_{acc})$ is used for all nodes at depth i (then there is no more freedom in defining for the rest of the run). Then, for all node u at depth i, the branches in $\mathrm{Cone}(h(u))$ are rejecting and those in $\mathrm{Cone}(u) \setminus \mathrm{Cone}(h(u))$ are accepting. Therefore, $\mu(\mathrm{AccBr}(\rho_t)) = 1 - 2^i \times 2^{-(i+1+i)} = 2^{-(i+1)}$. Thus, $\sup_{\rho_t^i} \mu(\mathrm{AccBr}(\rho_t)) = 1$, implying that $\mathrm{Val}_{\mathcal{A}}(t) = 1$.

Actually, the proof of Theorem 16 directly leads to the following.

COROLLARY 23. *Let $\mathcal{A}$ be a parity tree automaton and let t be a tree. Then,* $\mathrm{Val}_{\mathcal{A}}(t) = \mathrm{Val}(\mathbb{G}_{\mathcal{A},t})$.

3.8. Positive Tree Languages

So far, we favored the almost-sure acceptance condition (i.e., requiring the measure to be equal to 1) over the positive one (i.e., requiring the measure to be strictly positive). However, the decidability results on MDP stated in Theorem 3 still hold if we replace the almost-sure acceptance by the positive acceptance [Courcoubetis and Yannakakis 1990; Chatterjee et al. 2004]. We now discuss the impact when considering positive acceptance instead of almost-sure acceptance, and this motivates our choice to focus on almost-sure acceptance.

We say that a run ρ of a tree automaton $\mathcal{A}$ is ***positively accepting*** if the measure of its set of accepting branches is (strictly) positive; that is, $\mu(\mathrm{AccBr}(\rho)) > 0$. A tree t is ***positively accepted*** if there exists a positively accepting run of $\mathcal{A}$ over t, and we denote by $L_{\mathrm{Pos}}(\mathcal{A})$ the set of all trees positively accepted by $\mathcal{A}$. Finally, a ***positive tree language*** is a language L of trees such that there exists a parity automaton $\mathcal{A}$ with $L_{\mathrm{Pos}}(\mathcal{A}) = L$.

Note that, contrarily to the qualitative semantics, we can no longer assume that our automata are complete. In particular, the naive idea of adding a sink state does not work because it would have new runs that may be accepting (going to the sink state yields a rejecting cone, but this may not affect the positivity of the measure of the set of accepting branches).

Example 24. Consider the language $\mathcal{L}_{\mathtt{a}}^{>0}$ of $\{\mathtt{a}, \mathtt{b}\}$-labeled trees that have a non-negligible set of branches containing infinitely many $\mathtt{a}$'s. This language is positively accepted by a deterministic Büchi tree automaton (that goes into a final state whenever an $\mathtt{a}$ is read and into a nonfinal state otherwise) and hence is a positive tree language.

Remark 25. The complement of a language qualitatively accepted by a *deterministic* tree automaton $\mathcal{A}$ is positively accepted by a deterministic tree automaton $\mathcal{B}$. Indeed, it suffices to define $\mathcal{B}$ starting from $\mathcal{A}$ and dualize its acceptance condition (namely, increment the value of the coloring function by 1).

Let us now give two examples of languages that are not positive tree languages. For this, we introduce the language $\mathcal{L}_{\mathtt{a}\wedge\mathtt{a}}$ of $\{\mathtt{a},\mathtt{b}\}$-labeled trees containing an a in both their left and right subtrees, which is formally defined as $\mathcal{L}_{\mathtt{a}\wedge\mathtt{a}} = \{t \mid \exists u, v \in \{0,1\}^* \text{ s.t. } t(0u) = t(1v) = \mathtt{a}\}$.

Proposition 26. *The language $\mathcal{L}_{\mathtt{a}}$ of $\{\mathtt{a},\mathtt{b}\}$-labeled trees whose set of branches containing an* a *has a measure of* 1 *and the language $\mathcal{L}_{\mathtt{a}\wedge\mathtt{a}}$ of $\{\mathtt{a},\mathtt{b}\}$-labeled trees containing an* a *in both their left and right subtrees are not positive tree languages.*

Proof. The proofs for both languages rely on the same family of counterexamples. For every $n \geq 0$, let t_n be the tree defined by $t_n(u) = \mathtt{b}$ for $|u| \leq n$ and $t_n(u) = \mathtt{a}$ otherwise, and let t_b be the tree such that $t_{\mathtt{b}}(u) = \mathtt{b}$ for all $u \in \{0,1\}^*$.

We are going to show that if t_{n+2} is positively accepted by a parity automaton $\mathcal{A}$ with n states, then $\mathcal{A}$ also accepts either the tree t^0_{n+2} obtained by replacing the left subtree of t_{n+2} by $t_{\mathtt{b}}$ or the tree t^1_{n+2} obtained by replacing the right subtree of t_{n+2} by t_b.

This property will allow us to conclude, as for all $n \geq 0$, t_n belongs to $\mathcal{L}_{\mathtt{a}}$ but t^0_n and t^1_n does not, then $\mathcal{L}_{\mathtt{a}}$ is not a positive language. The same arguments work for $\mathcal{L}_{\mathtt{a}\wedge\mathtt{a}}$.

Let ρ be a positively accepting run of $\mathcal{A}$ on t_{n+2}. Without loss of generality, we can assume that the set of accepting branches in the right subtree has a strictly positive measure. We are going to show that $\mathcal{A}$ admits a run ρ' (not necessarily a positively accepting one) on $t_{\mathtt{b}}$ starting from $\rho(0)$. By replacing the left subtree of ρ by ρ', we then obtain a positively accepting run for $\mathcal{A}$ on t^1_{n+2}.

For all $i \geq 1$, consider the set $Q_i = \{\rho(0u) \mid u \in \{0,1\}^* \text{ and } 0 \leq |u| \leq i-1\}$ of those states appearing in the left subtree of ρ at a depth of at most i. Because the sequence of the Q_i is increasing, we have $Q_n = Q_{n+1}$. For all $q \in Q_n$, there exists a least one transition δ_q of $\mathcal{A}$ of the form $(q, \mathtt{b}, q_1, q_2)$ with q_1 and $q_2 \in Q_n$ (recall that in t_{n+2} the first $n+2$ levels only contain b's). By using the transition δ_q when in state q and starting from state q_0, we easily construct a run of $\mathcal{A}$ on $t_{\mathtt{b}}$ starting in state $\rho(0)$. □

Proposition 27. *The classes of positive and qualitative tree languages are incomparable.*

Proof. Let $\mathcal{L}_{\mathtt{a}}$ be the qualitative language of those $\{\mathtt{a},\mathtt{b}\}$-labeled trees whose set of branches containing an a have a measure of 1. We proved in Proposition 15 that its complement $\overline{\mathcal{L}_{\mathtt{a}}}$ is not qualitative. But $\overline{\mathcal{L}_{\mathtt{a}}}$ is a positive tree language because it is the complement of a language qualitatively accepted by a deterministic tree automaton (see Remark 25).

Conversely, we have seen in Proposition 26 that the qualitative language $\mathcal{L}_{\mathtt{a}}$ is not positive. □

Remark 28. A natural open question is whether a tree language that is both positively and qualitatively accepted is always regular.

As it is the case for qualitative languages, the positive languages are incomparable with the regular languages.

Proposition 29. *The class of positive tree languages is incomparable with the class of regular languages.*

Proof. The language $\mathcal{L}_{\mathtt{a}\wedge\mathtt{a}}$ of $\{\mathtt{a},\mathtt{b}\}$-labeled trees containing an a in both their left and right subtrees is an example of a regular language that is not a positive tree language (cf. Proposition 26).

We are now going to show that the positive tree language $\mathcal{L}^{>0}_{\mathtt{a}}$ of $\{\mathtt{a},\mathtt{b}\}$-labeled trees that have a non-negligible set of branches containing infinitely many a's (cf. Example 24) is not a regular language.

Toward a contradiction, we assume that $\mathcal{L}_{\mathrm{a}}^{>0}$ is accepted by a parity tree automaton $\mathcal{A} = \langle A, Q, q_{ini}, \Delta, \mathrm{Col}\rangle$ (with the standard acceptance condition for runs).

For every $i \geq 1$, we let r_i be the (finite) tree whose domain consists of all words of length at most i and in which only 0^i is labeled by a and all other nodes are labeled by b. We denote by $|r_i| = i$ the height of r_i. The probability p_{r_i} to visit an a-labeled node in r_i when starting from the root is $\frac{1}{2^i}$. We let $\mathcal{T}$ denote the set $\{r_i \mid i \geq 1\}$ of all such trees.

To every sequence $s = (t_i)_{i\geq 0}$ of trees in $\mathcal{T}$, we associate an infinite $\{\mathrm{a}, \mathrm{b}\}$-labeled tree $t_{s,}$ which is intuitively obtained as follows: start with t_1 and glue to every leaf a copy of t_2, to every leaf glue a copy of $t_{3,}$ and so on. Formally, $t_s(u) = \mathrm{a}$ if u is of the form $v0^{|t_i|}$ with $|v| = \sum_{j<i} |t_j|$ and $t_s(u) = \mathrm{b}$ otherwise.

By Borel-Cantelli's lemma [Bauer 1996, Lemma 11.1, p. 70], the tree t_s belongs to $L_{\mathrm{a}}^{>0}$ if and only if $\sum_{i\geq 0} p_{r_i} = +\infty$.

We extend the notion of runs for $\mathcal{A}$ to finite trees. A run of $\mathcal{A}$ on finite tree $t : \{0, 1\}^{\leq n} \mapsto \{\mathrm{a}, \mathrm{b}\}$ is a finite tree $\rho : \{0, 1\}^{\leq n} \mapsto Q$ such that for all $u \in \{0, 1\}^{<n}$, $(\rho(u), t(a), \rho(u0), \rho(u1))$ belongs to Δ. We can summarize the run ρ by a finite information called the profile of the run and denoted $\mathrm{Prof}(\rho)$. The profile $\mathrm{Prof}(\rho)$ of ρ is defined as:

$$\mathrm{Prof}(\rho) = (\rho(\varepsilon), R) \quad \text{where} \quad R = \{(m, \rho(u)) \mid u \in \{0, 1\}^n \text{ and } m = \inf_{v \sqsubseteq u} \mathrm{Col}(v)\}.$$

By extension, the profile of a finite tree t, denoted Prof(t), is the set $\{\mathrm{Prof}(\rho) \mid \rho$ run of $\mathcal{A}$ on $t\}$ of all possible profiles of runs of $\mathcal{A}$ on t.

This notion induces an equivalence relation on finite trees denoted $\equiv_{\mathcal{A},}$ which intuitively equates trees that cannot be distinguished by $\mathcal{A}$. Formally, for all finite trees t and t', $t \equiv_{\mathcal{A}} t'$ if and only if $\mathrm{Prof}(t) = \mathrm{Prof}\ (t')$.

The following claim is straightforward.

Claim 30. *Let $s = (t_i)_{i\geq 1}$ and $s' = (t'_i)_{i\geq 1}$ be two sequences of finite trees in $\mathcal{T}$. If for all $i \geq 1$, $t_i \equiv_{\mathcal{A}} t'_i$ then $\mathcal{A}$ accepts t_s if and only if $\mathcal{A}$ accepts $t_{s'}$.*

Now note that there are only finitely many different possible profiles; hence, there exists an infinite sequence $s = (r_{i_j})_{j\geq 0}$ of trees with the same profile and such that the sequence $(i_j)_{j\geq 0}$ is strictly increasing. Because the $(i_j)_{j\geq 0}$ sequence is strictly increasing, it implies that $\sum_{j\geq 0} p_{r_{i_j}} < +\infty$; hence, the tree t_s does not belong to $L_{\mathrm{a}}^{>0}$. Now consider the constant sequence $s' = r_{i_0}, r_{i_0}, \ldots$: one has $\sum_{j\geq 0} p_{r_{i_0}} = +\infty$; hence, $t_{s'}$ belongs to $L_{\mathrm{a}}^{>0}$. This leads to a contradiction with $\mathcal{A}$ accepting $L_{\mathrm{a}}^{>0}$ because $\mathcal{A}$ behaves the same on both t_s and $t_{s'}$; that is, $\mathcal{A}$ accepts t_s if and only if $\mathcal{A}$ accepts $t_{s'}$. □

The most notable difference between positive and qualitative tree languages is that positive tree languages are not closed under intersection.

Proposition 31. *The class of positive tree languages is closed under union but neither under intersection nor complementation.*

Proof. The closure under union is immediate thanks to nondeterminism. The nonclosure under complementation will be a consequence of the closure under union and the nonclosure under intersection. However, one can also derive it directly by considering the language $\mathcal{L}_{\mathrm{a}}$: we have shown in the proof of Proposition 26 that this language is not positive whereas its complement is positive.

We now prove the nonclosure under intersection. Consider the language L_0 (*resp.* L_1) of those trees t containing a non-negligible set of branches containing an a in the left (*resp.* right) subtree of the root. Both languages are positively accepted by a deterministic reachability automaton. But we have seen in Proposition 26 that their intersection $\mathcal{L}_{\mathrm{a}\wedge\mathrm{a}}$ is not a positive tree language. □

However, positive tree languages enjoy all decidability properties of qualitative languages presented in Section 3.5.

THEOREM 32. *The tree t belongs to $L_{\text{Pos}}(\mathcal{A})$ if and only if Éloïse positively wins in $\mathbb{G}_{\mathcal{A},t}$.*

PROOF. The proof is identical to that of Theorem 16. □

THEOREM 33. *The language $L_{\text{Pos}}(\mathcal{A})$ is nonempty if and only if Éloïse positively wins in $\mathbb{G}_{\mathcal{A}}$ from q_{ini}.*

PROOF. The proof is identical to that of Theorem 17. □

COROLLARY 34. *Let $\mathcal{A}$ be a parity tree automaton. Then one can decide whether $L_{\text{Pos}}(\mathcal{A}) = \emptyset$ in polynomial time. Moreover, if $L_{\text{Pos}}(\mathcal{A}) \neq \emptyset$, it contains a regular tree, and such a tree can be constructed in polynomial time.*

PROOF. Because the properties expressed in Theorem 3 of MDP with almost-sure winning condition also hold for the positive winning condition, we can use the same proof as for Corollary 18. □

4. BEYOND NONDETERMINISTIC AUTOMATA: THE PROBABILISTIC SETTING

Following Rabin [1963] for finite words and Baier and Größer [2005] and Baier et al. [2008, 2012] for infinite words, we investigate probabilistic automata on infinite trees. That is, the set of transitions of an automaton is replaced by a probability distribution over the set of all transitions, which induces a probability measure on the set of runs of the automaton. Now, a tree is accepted if almost every run over the input tree is accepting. For the run, we may use either the classical or the qualitative acceptance criterion.

4.1. Definitions

4.1.1. Probabilistic Tree Automata. A ***probabilistic tree automaton*** $\mathcal{A}$ is a tuple $\langle A, Q, q_{ini}, \delta, \text{Acc}\rangle$ where A is the ***input alphabet***, Q is a finite ***set of states***, $q_{\text{ini}} \in Q$ is the ***initial state***, $\text{Acc} \subseteq Q^\omega$ is the ***acceptance condition,*** and δ is a mapping from $Q \times A \times Q \times Q$ to $[0, 1]$ such that for all $q \in Q$ and $\mathtt{a} \in A$, $\sum_{q_0,q_1 \in Q} \delta(q, \mathtt{a}, q_0, q_1) = 1$. Intuitively, the value $\delta(q, \mathtt{a}, q_0, q_1)$ is the probability for a transition $q \xrightarrow{\mathtt{a}} (q_0, q_1)$ to be used by the automaton when it is in state q and reads the symbol $\mathtt{a}$.

This probability distribution on the transitions induces a probability measure on the set of runs of $\mathcal{A}$. In this setting, a ***run*** of $\mathcal{A}$ is simply a Q-labeled tree whose root is labeled by the initial state q_{ini}. We denote by $\text{Runs}\,(\mathcal{A})$ (or simply Runs if $\mathcal{A}$ is clear from the context) the set of all runs of $\mathcal{A}$. We denote by $\text{AccRuns}(\mathcal{A})$ the set of accepting runs of $\mathcal{A}$ and by $\text{QualAccRuns}(\mathcal{A})$ the set of qualitatively accepting runs of $\mathcal{A}$.

In the sequel, we will show that, for a given tree t, the sets $\text{AccRuns}(\mathcal{A})$ and $\text{QualAccRuns}(\mathcal{A})$ are measurable (Proposition 38), and this will allows us to define almost-sure acceptance of a tree by a probabilistic automaton.

4.1.2. Measurability of $\text{AccRuns}(\mathcal{A})$ *and* $\text{QualAccRuns}(\mathcal{A})$. We first define a σ-algebra for $\text{Runs}(\mathcal{A})$. A ***partial run*** is a partial function $\lambda : \{0, 1\}^* \to Q$ with $\lambda(\varepsilon) = q_{\text{ini}}$ and such that $\text{Dom}(\lambda)$ is finite, nonempty, prefix-closed, and proper; that is, such that for all $w \in \{0, 1\}^*$, $w0 \in \text{Dom}(\lambda)$ iff $w1 \in \text{Dom}(\lambda)$. If λ is a partial run, we denote by $\text{Inner}(\lambda)$ the set $\{w \in \text{Dom}(\lambda) \mid w0 \in \text{Dom}(\lambda) \text{ and } w1 \in \text{Dom}(\lambda)\}$ of nonleaf nodes. The cylinder associated with λ, denoted $\text{Cyl}(\lambda)$, is the set of runs consistent with λ; that is,

$$\text{Cyl}(\lambda) = \{\rho \in \text{Runs} \mid \forall w \in \text{Dom}(\lambda), \rho(w) = \lambda(w)\}$$

Let $\mathcal{F}_R$ be the σ-algebra generated by the cylinders. By Carathéodory's extension theorem, there exists a unique probability measure μ_t on the measurable space (Runs, $\mathcal{F}_R$) such that, for all partial run $\lambda : \{0,1\}^* \to Q$,

$$\mu_t(\mathrm{Cyl}_{\mathcal{A}}(\lambda)) = \prod_{w \in \mathrm{Inner}(\lambda)} \delta(\lambda(w), t(w), \lambda(w0), \lambda(w1)).$$

Note that both μ_t and (Runs, $\mathcal{F}_R$) depend on t. Also note that the definition of μ_t does not make sense without the properness condition in the definition of a partial run.

Remark 35. A ***balanced partial run*** λ is a mapping from $\{0,1\}^n$ to Q for some $n \geq 0$ with $\lambda(\varepsilon) = q_{\mathrm{ini}}$. Hence, it is a special kind of partial play in the previous sense. It is straightforward to note that the σ-algebra generated by balanced partial play is $\mathcal{F}_R$. Moreover, one has

$$\mu_t(\mathrm{Cyl}_{\mathcal{A}}(\lambda)) = \prod_{w \in \{0,1\}^{\leq n-1}} \delta(\lambda(w), t(w), \lambda(w0), \lambda(w1)).$$

Let $\mathcal{A}$ be an automaton, and define the mapping $f_{\mathcal{A}} : \mathrm{Runs} \times \mathrm{Br} \to [0,1]$ associating to a pair $(\rho, \pi) \in \mathrm{Runs} \times \mathrm{Br}$ the value 1 if $\rho(\pi)$ belongs to Acc and 0 otherwise. The following lemma proves that $f_{\mathcal{A}}$ is integrable.

LEMMA 36. *Let $\mathcal{A}$ be a probabilistic tree automaton with an ω-regular acceptance condition, and let t be a tree. The mapping $f_{\mathcal{A}}$ is integrable in the product space* $(\mathrm{Runs}(\mathcal{A}), \mathcal{F}_R, \mu_t) \otimes (\mathrm{Br}, \mathcal{F}_{\mathrm{Br}}, \mu)$.

PROOF. The product space $(\mathrm{Runs}, \mathcal{F}_R) \otimes (\mathrm{Br}, \mathcal{F}_{\mathrm{Br}})$ is generated by the sets $\mathrm{Cone}(w) \times \mathrm{Cyl}(\lambda)$ for all $w \in \{0,1\}^*$ and all partial run λ (cf. Bauer [2001, Theorem 22.1, p. 132]). Thanks to Remark 35, we can restrict our attention to balanced partial runs.

According to Bauer [2001, Theorem 9.1, p. 50], it is enough to show that $\mathrm{AccPairs}(\mathcal{A}) = \{(\rho, \pi) \mid \rho(\pi) \in \mathrm{Acc}\}$ is measurable in the product space.

We first establish the result for a Büchi acceptance condition. For that, we let $\mathcal{A} = \langle A, Q, q_{\mathrm{ini}}, \delta, F \rangle$ be a probabilistic tree automaton with a Büchi acceptance condition.

For $u \in \{0,1\}^*$, we introduce the set R_u of all pairs $(\rho, \pi) \in \mathrm{Runs} \times \mathrm{Br}$ such that π admits u as a prefix and $\rho(v) \notin F$ for all $u \not\sqsupseteq v \sqsubseteq \pi$. The set $\mathrm{AccPairs}(\mathcal{A})$ is the complement of the countable union $\bigcup_{u \in \{0,1\}^*} R_u$. Hence, it is enough to show R_u is measurable for all $u \in \{0,1\}^*$.

Let us fix a node $u \in \{0,1\}^*$. For all $n > |u|$, let C_n be the union of all sets $\mathrm{Cyl}(\lambda) \times \mathrm{Cone}(w)$ where $\lambda : \{0,1\}^n \to Q$ is a partial run and $w \in \{0,1\}^n \not\sqsupseteq u$ is a word such that $\lambda(v) \notin F$ for all $u \not\sqsupseteq v \sqsubseteq w$. We claim that the set R_u is equal to $\bigcap_{n > |u|} C_n$ and hence is measurable. The direct inclusion, $R_u \subseteq \bigcap_{n>|u|} C_n$, is obvious. For the converse one, let (ρ, π) be in $\bigcap_{n>|u|} C_n$. For all $n > |u|$, the node $w_n \in \{0,1\}^n$, witnessing that (ρ, π) belongs to C_n is indeed the unique node of π of depth n. Because n can be arbitrarily large, it follows that every node v in π of depth $> |u|$ is such $\rho(v) \notin F$, which exactly means that $(\rho, \pi) \in R_u$.

The case of parity conditions follows from the previous case because one can express them as a Boolean combination of Büchi conditions.

The general case of an arbitrary ω-regular condition is obtained as follows. First, one considers a *deterministic* parity word automaton recognizing Acc, and then one takes the synchronized product of this automaton with the tree automaton. This leads to a parity tree automaton accepting the same language and whose runs are in bijection with the ones of the original automaton. Moreover, because this bijection preserves the set of accepting branches, one can conclude from the parity case. □

Note that the previous lemma implies that the set of accepting branches in a run of a tree automaton with an ω-regular acceptance condition is measurable in $(\mathrm{Br}, \mathcal{F}_{\mathrm{Br}})$. This leads to an alternative proof of Proposition 6.

COROLLARY 37. *Let $\mathcal{A}$ be a tree automaton equipped with an ω-regular acceptance condition, and let ρ be a run of $\mathcal{A}$ over some tree t. Then, the set* $\mathrm{AccBr}(\rho)$ *of accepting branches in t is measurable.*

PROOF. Let $\mathcal{A}$ be a tree automaton with an ω-regular acceptance condition, t be a tree, and ρ be a run of $\mathcal{A}$ on t. The set $\mathrm{AccBr}(\rho)$ is equal to $\{\lambda \in \mathrm{Br} \mid \rho(\lambda) \in \mathrm{Acc}\}$. As by the proof of Lemma 36, $\mathrm{AccPairs}(\mathcal{A})$ is measurable in the product space, $\mathrm{AccBr}(\rho)$ is also measurable [Bauer 2001, Lemma 23.1, p. 135]). □

We are now ready to establish the measurability of the sets $\mathrm{AccRuns}(\mathcal{A})$ and $\mathrm{QualAccRuns}(\mathcal{A})$.

PROPOSITION 38. *For all probabilistic tree automata $\mathcal{A}$ with an ω-regular acceptance condition, the sets* $\mathrm{AccRuns}(\mathcal{A})$ *and* $\mathrm{QualAccRuns}(\mathcal{A})$ *are measurable.*

PROOF. We first consider the case of $\mathrm{AccRuns}(\mathcal{A})$. As seen in Proposition 6, it is enough to establish the property for the Büchi acceptance condition.

Let $\langle A, Q, q_{ini}, \delta, F\rangle$ be a probabilistic tree automaton with a Büchi acceptance condition.

For $u \in \{0,1\}^*$, we introduce the set R_u of all runs ρ of $\mathcal{A}$ for which there exists an infinite branch π, such that $u \sqsubseteq \pi$ and for all $u \sqsubseteq v \not\sqsubseteq \pi$, $\rho(v) \notin F$. The set of accepting runs $\mathrm{AccRuns}(\mathcal{A})$ is the complement of the countable union $\bigcup_{u\in\{0,1\}^*} R_u$. Hence, it is enough to show R_u is measurable for all $u \in \{0,1\}^*$.

Let us fix a node $u \in \{0,1\}^*$. For all $n > |u|$, let C_n be the union of all cylinders $\mathrm{Cyl}(\lambda)$ where $\lambda : \{0,1\}^n \to Q$ is a (balanced) partial run for which there exists a word $w \in \{0,1\}^n \sqsupseteq u$ such that $\lambda(v) \notin F$ for all $u \not\sqsubseteq v \sqsubseteq w$.

We claim that R_u is equal to $\bigcap_{n>|u|} C_n$ and is hence measurable. The direct inclusion, $R_u \subseteq \bigcap_{n>|u|} C_n$, is obvious. For the converse one, let ρ be a run in $\bigcap_{n>|u|} C_n$. For all $n > |u|$, let w_n be a word in $\{0,1\}^n$ witnessing that ρ belongs to C_n. Let T be the prefix-closure of the set $\{w_n \mid n > |u|\}$. By Koenig's lemma, T contain an infinite branch π. Clearly, u is a prefix of π and for all $u \sqsubseteq v \not\sqsubseteq \pi$, $\rho(v) \notin F$ as v is a prefix of one of the w_n. Hence, ρ belongs to R_u.

Let us now consider the case of $\mathrm{QualAccRuns}(\mathcal{A})$. From Lemma 36, we have that $\mathrm{AccPairs}(\mathcal{A}) = \{(\rho, \pi) \mid \rho(\pi) \in \mathrm{Acc}\} = f_{\mathcal{A}}^{-1}(\{1\})$ is measurable in the product space. Hence, the numerical function $g : \mathrm{Runs} \to [0,1]$ associating to a run ρ the measure of its set of accepting branches (i.e., $\mu(\mathrm{AccPairs}(\mathcal{A})_\rho)$ where $\mathrm{AccPairs}(\mathcal{A})_\rho = \{\lambda \in \mathrm{Br} \mid \rho(\lambda) \in \mathrm{Acc}\}$) is measurable [Bauer 2001, Lemma 23.1, p. 135]. Because $\mathrm{QualAccRuns}(\mathcal{A})$ is equal to $g^{-1}(\{1\})$, it is measurable. □

4.1.3. Almost-Surely Accepted Trees. Proposition 38 shows that the following definition is well-founded.

A tree t is ***(almost-surely) accepted*** by $\mathcal{A}$ with the classical (*resp.* qualitative) semantics if almost all runs of $\mathcal{A}$ on t are accepting (*resp.* qualitatively accepting); that is, $\mu_t(\mathrm{AccRuns}(\mathcal{A})) = 1$ (*resp.* $\mu_t(\mathrm{QualAccRuns}(\mathcal{A})) = 1$). We denote by $L^{=1}(\mathcal{A})$ (*resp.* $L^{=1}_{\mathrm{Qual}}(\mathcal{A})$) the set of trees accepted by $\mathcal{A}$ with the classical (*resp.* qualitative) semantics for runs. More formally, we define $L^{=1}(\mathcal{A}) = \{t \mid \mu_t(\mathrm{AccRuns}(\mathcal{A})) = 1\}$ and $L^{=1}_{\mathrm{Qual}}(\mathcal{A}) = \{t \mid \mu_t(\mathrm{QualAccRuns}(\mathcal{A})) = 1\}$.

Remark 39. Our motivation for considering almost-sure acceptance and not positive acceptance is discussed in Section 4.5.

The following easy lemma is used later to prove Proposition 42 and Proposition 43.

LEMMA 40. *Let* $(\Omega, \mathcal{F}, \mu)$ *be a probability space and* f *be a measurable function from* Ω *to* $[0, 1]$, *then* $\int_\Omega f d\mu = 1$ *if and only if* $\mu(f^{-1}(\{1\})) = 1$.

PROOF. Because f is bounded, measurability implies integrability. Consider the integrable mapping $g = 1 - f$. Clearly, $\int_\Omega f d\mu = 1$ if and only if $\int_\Omega g d\mu = 0$. By Bauer [2001, Theorem 13.2, p. 71], $\int_\Omega g d\mu = 0$ if and only if $g^{-1}(\{0\})$ has measure 1. The announced equivalence follows. □

Remark 41. The definition of Baier and Größer [2005] (for ω-words) allows for incomplete automata (i.e., for all $q \in Q$, the sum $\sum_{q' \in Q} \delta(q, a, q') = 1$ or 0). However it is easy to verify that every automaton can be rendered complete without changing the acceptance condition and the measure of acceptance. Notice that removing all states q such that $\sum_{q_1, q_2 \in Q} \delta(q, a, q_1, q_2) = 0$ does not change the measure of the set of accepting runs (either in the classical or qualitative sense). By iteratively applying this process, we obtain an equivalent complete automata.

Combining Lemma 36 and Lemma 40, we can show that the almost-sure acceptance of a tree t by an automaton $\mathcal{A}$ for the qualitative semantics can be defined by integrating the mapping $f_{\mathcal{A}}$.

PROPOSITION 42. *Let* $\mathcal{A}$ *be a probabilistic tree automaton with an* ω*-regular acceptance condition and let* t *be a tree. Then we have:*

$$t \in L^{=1}_{\mathrm{Qual}}(\mathcal{A}) \Leftrightarrow \int f_{\mathcal{A}} d\mu_t \otimes \mu = 1.$$

PROOF. Because $f_{\mathcal{A}}$ is measurable (Lemma 36), by Tonelli's theorem [Bauer 2001, Theorem 23.6, p. 138], the mapping $g : \mathrm{Runs} \to [0, 1]$ associating to a run $\rho \in \mathrm{Runs}$ the value $\int_{\mathrm{Br}} f_{\mathcal{A}}(\rho, \cdot) d\mu$ is measurable.

	t belongs to $L^{=1}_{\mathrm{Qual}}(\mathcal{A})$	
iff	$\mu_t(g^{-1}(\{1\})) = 1$	
iff	$\int_{\mathrm{Runs}} g\, d\mu_t = 1$	by Lemma 40
iff	$\int_{\mathrm{Runs}} \int_{\mathrm{Br}} f_{\mathcal{A}} d\mu d\mu_t = 1$	by definition of g
iff	$\int_{\mathrm{Runs} \times \mathrm{Br}} f_{\mathcal{A}} d\mu_t \otimes \mu = 1$	by Tonelli's theorem [Bauer 2001, Theorem 23.6, p. 138] □

4.1.4. Examples. We conclude this section with examples of languages accepted by probabilistic tree automata.

For an ω-word language $L \subseteq \{\mathtt{a}, \mathtt{b}\}^\omega$, we denote by $\mathrm{Path}^{=1}(L)$ the set of trees labeled by $\{\mathtt{a}, \mathtt{b}\}$ with almost all their branch labels in L (i.e., $\mu(\{\pi \in \mathrm{Br} \mid t(\pi) \in L\}) = 1$). It is easy to see that, for any ω-regular language L, the tree language $\mathrm{Path}^{=1}(L)$ is a qualitative tree language.

More interestingly, if L is almost-surely accepted by a probabilistic ω-word automaton[6] with an ω-regular acceptance condition, we can show that $\mathrm{Path}^{=1}(L)$ is accepted by a probabilistic tree automaton (with the qualitative semantics).

PROPOSITION 43. *Given a probabilistic* ω*-word automaton* $\mathcal{B}$ *with an* ω*-regular acceptance condition, there exists a probabilistic tree automaton* $\mathcal{A}$ *with the same acceptance condition such that* $L^{=1}_{\mathrm{Qual}}(\mathcal{A})$ *is equal to* $\mathrm{Path}^{=1}(L^{=1}(\mathcal{B}))$.

[6] In the context of this article, probabilistic ω-word automata are simply probabilistic tree automata running over unary trees. For such an automaton $\mathcal{B}$, we denote by $L^{=1}(\mathcal{B})$ the language almost-surely accepted by $\mathcal{B}$.

PROOF. Let $\mathcal{B} = \langle A, Q, q_{ini}, \delta, \mathrm{Acc}\rangle$ be a (complete) probabilistic ω-word automaton with an ω-regular condition. Consider the probabilistic tree automaton $\mathcal{A}$ simulating $\mathcal{B}$ on all branches. Formally, $\mathcal{A}$ is equal to $\langle A, Q, q_{ini}, \delta', F\rangle$ where, for all $p, q \in Q$ and $x \in A$, $\delta'(p, x, q, q) = \delta(p, x, q)$.

Let t be a tree and let $g : \mathrm{Br} \mapsto [0, 1]$ be the mapping associating to a branch π the value $\int f_{\mathcal{A}}(\cdot, \pi) d\mu_t$. We have the following claim.

CLAIM 44. *For any $\pi \in \mathrm{Br}$, $g(\pi) = 1$ if and only if $t(\pi)$ belongs to $L^{=1}(\mathcal{B})$.*

PROOF OF CLAIM 44. Let $\pi = a_1 a_2 \cdots \in \mathrm{Br}$ and let $w = t(\pi)$. It is enough to show that: $g(\pi) = \int f_{\mathcal{A}}(\cdot, \pi) d\mu_t$ is equal to $\mu_w(\mathrm{AccRuns}(\mathcal{B}))$. By definition $\int f_{\mathcal{A}}(\cdot, \pi) d\mu_t$ is equal to $\mu_t(\mathrm{AccRuns}_\pi(\mathcal{A}))$ where $\mu_t(\mathrm{AccRuns}_\pi(\mathcal{A}))$ is the set of runs ρ of $\mathcal{A}$ such that $\rho(\pi)$ is accepting.

Consider the mapping $\psi : \mathrm{Runs}(\mathcal{A}) \to \mathrm{Runs}(\mathcal{B})$ associating to a run ρ of $\mathcal{A}$ the corresponding run $\rho(\varepsilon)\rho(a_1) \cdots$ of $\mathcal{B}$. We claim that ψ is measurable and that μ_w is the image of μ_t under ψ. This claim in particular implies that:

$$\begin{aligned}\mu_w(\mathrm{AccRuns}(\mathcal{B})) &= \mu_t(\psi^{-1}(\mathrm{AccRuns}(\mathcal{B}))\\ &= \mu_t(\mathrm{AccRuns}_\pi(\mathcal{A}))\\ &= \textstyle\int f_{\mathcal{A}}(\cdot, \pi)\, d\mu_t.\end{aligned}$$

To substantiate our claim, it is enough to show that $\psi^{-1}(\mathrm{Cyl}(\lambda))$ is a measurable subset of $\mathrm{Runs}(\mathcal{A})$ such that $\mu_t(\psi^{-1}(\mathrm{Cyl}(\lambda))) = \mu_w(\mathrm{Cyl}(\lambda))$. The first part implies that ψ is measurable and the second that the image measure of μ_t under ψ, denoted $\psi(\mu_t)$, coincides with μ_w on the cylinders. By Carathéodory's unique extension theorem, this in turn implies that μ_w and $\psi(\mu_t)$ coincide.

Let $\lambda = q_0 \ldots q_n$ be a partial run of $\mathcal{B}$. Consider the partial balanced run η of $\mathcal{A}$ defined for $i \in [0, n-1]$ by $\eta(a_1 \cdots a_{i-1} 0) = \eta(a_1 \cdots a_{i-1} 1) = q_i$. It is clear that $\psi^{-1}(\mathrm{Cyl}(\lambda)) = \mathrm{Cyl}(\eta)$. Moreover, we have:

$$\begin{aligned}\mu_t(\mathrm{Cyl}(\eta)) &= \prod_{j\in[0,n-1]} \delta'(q_j, t(a_1 \cdots a_j), q_{j+1}, q_{j+1})\\ &= \prod_{j\in[0,n-1]} \delta(q_j, t(a_1 \cdots a_j), q_{j+1})\\ &= \mu_w(\mathrm{Cyl}(\lambda)). \qquad \square\end{aligned}$$

We are now ready to conclude:

$$\begin{array}{ll} t \in L^{=1}_{\mathrm{Qual}}(\mathcal{A}) & \\ \Leftrightarrow \displaystyle\int f_{\mathcal{A}} d\mu_t \otimes \mu = 1 & \text{Proposition 42}\\ \Leftrightarrow \displaystyle\int g d\mu = 1 & \text{Tonelli's Theorem}\\ \Leftrightarrow g(\pi) = 1 \text{ almost everywhere} & \text{Lemma 40}\\ \Leftrightarrow \mu(\{\pi \mid t(\pi) \in L^{=1}(\mathcal{B})\}) = 1 & \text{Claim 2}\\ \Leftrightarrow t \in \mathrm{Path}^{=1}(L^{=1}(\mathcal{B})) & \end{array}$$

4.2. Acceptance Game for Qualitative Probabilistic Tree Automata

Fix a probabilistic tree automaton $\mathcal{A} = \langle A, Q, q_{\mathrm{ini}}, \delta, \mathrm{Acc}\rangle$ and a tree t. We define an *infinite* Markov chain $\mathcal{M}_{\mathcal{A},t} = (G_{\mathcal{A},t}, \mathcal{O}_{\mathrm{Acc}})$ such that $\mathcal{M}_{\mathcal{A},t}$ almost-surely fulfils its objective iff t belongs to $L^{=1}_{\mathrm{Qual}}(\mathcal{A})$. Compared with the acceptance game for qualitative tree automata, the transition is no longer chosen by Éloïse: it is now randomly chosen with the probability distribution given by $\mathcal{A}$. Hence, this is why we simply obtain a Markov chain instead of an MDP.

The arena $G_{\mathcal{A},t}$ is equal to $\langle S, s_{\text{ini}}, \zeta\rangle$, where $S = Q \times \{0,1\}^* \cup \Delta \times \{0,1\}^*$ with $\Delta = Q \times Q \times Q$, $s_{\text{ini}} = (q_{\text{ini}}, \varepsilon)$, and $\zeta : S \mapsto D(S)$ is defined as follows. For all $w \in \{0,1\}^*$ and all $q \in Q$, $\zeta((q,w))((q,q_0,q_1)) = \delta(q, t(w), q_0, q_1)$ for all $q_0, q_1 \in Q$ and is equal to 0 otherwise. For all $w \in \{0,1\}^*$ and $q_0, q_1 \in Q$, $\zeta(((q,q_0,q_1),w),(q_0,w0) = \zeta(((q,q_0,q_1),w),(q_1,w1)) = \frac{1}{2}$ and 0 otherwise. Recall that $\mu_{\mathcal{M}_{\mathcal{A},t}}$ denotes the probability measure associated to $\mathcal{M}_{\mathcal{A},t}$.

Note that a trace in $\mathcal{M}_{\mathcal{A},t}$ can be uniquely represented by an infinite sequence $((p_0, q_0^0, q_0^1), a_0)((p_1, q_1^0, q_1^0), a_1)\ldots$ labeled by $\Delta \times \{0,1\}$ such that $p_0 = q_{\text{ini}}$ and for all $i \geq 0$, $p_{i+1} = q_i^{a_i}$. The objective $\mathcal{O}_{\text{Acc}}$ is defined as the set of those traces $((p_0, q_0^0, q_0^1), a_0)((p_1, q_1^0, q_1^0), a_1)\ldots$ such that $p_0 p_1 \ldots$ belongs to Acc.

Proposition 45. *Let $\mathcal{A}$ be a probabilistic tree automaton with an ω-regular acceptance condition and let t be a tree. We have $t \in L^{=1}_{\text{Qual}}(\mathcal{A})$ iff $\mathcal{M}_{\mathcal{A},t}$ almost-surely fulfils its objective.*

Proof. Let $\text{AccPairs}(\mathcal{A}) = \{(\rho,\pi) \mid \rho(\pi) \in \text{Acc}\} \subseteq \text{Runs} \times \text{Br}$. By Proposition 42, to establish the desired equivalence, it is sufficient to show that $\mu_t \otimes \mu(\text{AccPairs}(\mathcal{A})) = \mu_{\mathcal{M}_{\mathcal{A},t}}(\mathcal{O}_{\text{Acc}})$.

Consider the mapping $\psi : \text{Runs} \times \text{Br} \mapsto \text{Traces}$ associating with a pair $(\rho, a_0a_1\ldots)$, the trace $(\rho(\varepsilon), \rho(0), \rho(1))a_0(\rho(a_0), \rho(a_00), \rho(a_01))a_1\ldots$ of the Markov chain. It is clear that $\text{AccPairs}(\mathcal{A}) = \psi^{-1}(\mathcal{O}_{\text{Acc}})$.

We claim that ψ is measurable and that $\mu_{\mathcal{M}_{\mathcal{A},t}}$ is the image of $\mu_t \otimes \mu$ under ψ. In particular, this implies that $\mu_{\mathcal{M}_{\mathcal{A},t}}(\mathcal{O}_{\text{Acc}}) = \mu_t \otimes \mu(\text{AccPairs}(\mathcal{A}))$.

To substantiate our claim, it is enough to show, for any partial trace θ, that $\psi^{-1}(\text{Cyl}(\theta))$ is a measurable subset of $\text{Runs} \times \text{Br}$ such that $\mu_t \otimes \mu(\psi^{-1}(\text{Cyl}(\theta))) = \mu_{\mathcal{M}_{\mathcal{A},t}}(\text{Cyl}(\theta))$. The first part implies that ψ is measurable and the second that the image measure of $\mu_t \otimes \mu$ under ψ, denoted $\psi(\mu_t \otimes \mu)$, coincides with $\mu_{\mathcal{M}_{\mathcal{A},t}}$ on the cylinders. By Carathéodory's unique extension theorem, this in turn implies that $\mu_{\mathcal{M}_{\mathcal{A},t}}$ and $\psi(\mu_t \otimes \mu)$ coincide.

Let $\theta = (p_0, q_0^0, q_0^1)a_0 \ldots a_{n-1}(p_n, q_n^0, q_n^1)$ be a partial trace. Take $w = a_0 \ldots a_{n-1}$ and λ the (nonbalanced) partial run of $\mathcal{A}$ defined for all $i \in [0,n]$ by $\lambda(a_0 \cdots a_{i-1}) = p_i$, $\lambda(a_0 \cdots a_{i-1}0) = q_i^0$, and $\lambda(a_0 \ldots a_{i-1}1) = q_i^1$. We have $\psi^{-1}(\text{Cyl}(\theta)) = \text{Cyl}(\lambda) \times \text{Cyl}(w)$. By definition, $\mu_t(\text{Cyl}(\lambda))$ is equal to $\prod_{0 \leq i \leq n-1} \delta(p_i, t(a_0 \cdots a_{i-1}), q_i^0, q_i^1)$ and hence

$$
\begin{aligned}
\mu_t \otimes \mu(\psi^{-1}(\text{Cyl}(\theta))) &= \frac{1}{2^n} \prod_{0 \leq i \leq n-1} \delta\big(p_i, t(a_0 \cdots a_{i-1}), q_i^0, q_i^1\big) \\
&= \prod_{0 \leq i \leq n-1} \frac{1}{2} \delta\big(p_i, t(a_0 \cdots a_{i-1}), q_i^0, q_i^1\big) \\
&= \mu_{\mathcal{M}_{\mathcal{A},t}}(\text{Cyl}(\theta)). \qquad \square
\end{aligned}
$$

4.3. Decidability and Undecidability Results

In this section, we show that the emptiness problem for probabilistic Büchi tree automata is decidable for the qualitative semantics for runs. We reduce this problem to deciding almost-surely winning in a POMDP, and the reduction works for any ω-regular acceptance condition. However, the corresponding decision problem on POMDPs is only decidable for the Büchi condition. Hence, we only obtain decidability in the Büchi case.

Let $\mathcal{A} = \langle A, Q, q_{\text{ini}}, \delta, \text{Acc}\rangle$ be a probabilistic automaton with an ω-regular acceptance condition and let $\Delta = Q \times Q \times Q$.

Define a POMDP $\mathbb{G}_{\mathcal{A}} = (\mathcal{G}, \sim, \mathcal{O})$ as follows. The arena $\mathcal{G}$ is equal to $\langle S, s_{\text{ini}}, A, \zeta\rangle$ where $S = Q \times \{0, 1, \perp\} \times (\Delta \cup \{\perp\})$, $s_{\text{ini}} = (q_0, \perp, \perp)$, and ζ is defined as follows. For all $a \in A$ and $(p, x, t) \in S$, $\zeta((p, x, t), a)$ is the distribution that assigns $\frac{1}{2}\delta(p, a, q_0, q_1)$ to $(q_y, y, (p, q_0, q_1))$ where $y = 0, 1$ and 0 to all other state. The objective $\mathcal{O}$ is the set of plays for which the sequence of states obtained when projecting on the first component belongs to Acc. The equivalence $\sim$ is defined by $(q, x, t) \sim (q', x', t')$ iff $x = x'$. Intuitively in $\mathbb{G}_{\mathcal{A}}$, Éloïse describes a branch along a tree and Random builds a piece of a run along this branch. Because Éloïse does not observe the state in the run constructed by Random, it does not influence her choice for the branch.

Theorem 46. *Let $\mathcal{A}$ be a probabilistic tree automaton with an ω-regular acceptance condition. The language $L^{=1}_{\text{Qual}}(\mathcal{A})$ is nonempty if and only if Éloïse almost-surely wins in $\mathbb{G}_{\mathcal{A}}$.*

Proof. From the definitions, we easily remark that $\{[\lambda]_\sim \mid \lambda \in \text{Plays}\}$ is in bijection with $= \{0, 1\}^*$. Hence, strategies are in bijection with A-labeled trees. Once such a strategy φ_t (seen as a tree t) is fixed, the resulting Markov chain is, up to renaming, $\mathbb{G}_{\mathcal{A},t}$; meaning that the value of φ_t is the value of $\mathbb{G}_{\mathcal{A},t}$. In particular, Éloïse almost-surely wins in $\mathbb{G}_{\mathcal{A}}$ iff there is some t such that $\text{Val}(\mathbb{G}_{\mathcal{A},t}) = 1$ iff $t \in L^{=1}_{\text{Qual}}(\mathcal{A})$ (thanks to Proposition 45). $\square$

For the Büchi acceptance condition, this leads to a decidability result for the emptiness problem.

Corollary 47. *Let $\mathcal{A}$ be a probabilistic Büchi tree automaton. Deciding emptiness of $L^{=1}_{\text{Qual}}(\mathcal{A})$ is an* ExpTime*-complete problem. Moreover, if $L^{=1}_{\text{Qual}}(\mathcal{A}) \neq \emptyset$, it contains a regular tree.*

Proof. The ExpTime upper bound follows from the polynomial time reduction to deciding almost-surely winning in a Büchi POMDP. The lower bound follows from Proposition 43: emptiness of probabilistic Büchi ω-word automata with the almost-sure acceptance (which is ExpTime-complete [Baier et al. 2008; Baier et al. 2012]) can be reduced to our problem. Indeed, if $\mathcal{B}$ is a probabilistic Büchi ω-word automaton, we can construct a probabilistic Büchi tree automaton $\mathcal{A}$ of linear size (cf. Proposition 43) such that $L^{=1}_{\text{Qual}}(\mathcal{A})$ is equal to $\text{Path}^{=1}(L^{=1}(\mathcal{B}))$ and, in particular, $L^{=1}_{\text{Qual}}(\mathcal{A})$ is empty if and only if $L^{=1}(\mathcal{B})$ is empty.

The proof of Theorem 46 together with the fact that a *finite memory* optimal strategy always exists in POMDP imply the existence of a regular tree when $L^{=1}_{\text{Qual}}(\mathcal{A}) \neq \emptyset$. $\square$

On the negative side, we show that the emptiness problem for probabilistic co-Büchi tree automata is undecidable for both the classical and qualitative semantics for runs. These results are obtained by reduction to the undecidability of the emptiness problem for co-Büchi ω-word automata with the almost-sure acceptance [Baier et al. 2008].

Theorem 48. *The following problems are undecidable:*

(1) *given a probabilistic co-Büchi tree automaton $\mathcal{A}$, decide if $L^{=1}(\mathcal{A}) = \emptyset$,*
(2) *given a probabilistic co-Büchi tree automaton $\mathcal{A}$, decide if $L^{=1}_{\text{Qual}}(\mathcal{A}) = \emptyset$.*

Proof. Both undecidability results are shown by reduction to the undecidability of the emptiness problem for co-Büchi ω-word automata (with the almost-sure acceptance).

Let $\mathcal{B} = \langle A = \{\mathtt{a}, \mathtt{b}\}, Q, q_{ini}, \delta, F\rangle$ be a probabilistic co-Büchi ω-word automaton.

(1) We construct a probabilistic co-Büchi tree automaton $\mathcal{A}$ that simulates $\mathcal{B}$ on the left-most branch of the tree and checks that all other branches contain only a. Formally, $\mathcal{A}$ is equal to $\langle A, Q \cup \{q_a, q_\perp\}, q_{\text{ini}}, \delta', F \cup \{q_\perp\}\rangle$. The probability distribution δ' is given by:

$$\begin{cases} \delta'(q_a, \mathtt{a}, q_a, q_a) &= 1 \\ \delta'(q_a, \mathtt{b}, q_\perp, q_\perp) &= 1 \\ \delta'(q_\perp, x, q_\perp, q_\perp) &= 1 & \text{for any } x \in A \\ \delta'(p, x, q, q_a) &= \delta(p, x, q) & \text{for any } x \in A \text{ and any } p, q \in Q. \end{cases}$$

In all other cases, δ' is equal to 0.

We claim that a tree t belongs to $L^{=1}(\mathcal{A})$ if and only if for all $u \notin 0^*$, $t(u) = a$ and $t(0^\omega) \in L^{=1}(\mathcal{B})$. In particular, $L^{=1}(\mathcal{A})$ is empty if and only if $L^{=1}(\mathcal{B})$ is empty.

Let t be a tree accepted by $L^{=1}(\mathcal{A})$. Toward a contradiction, assume that for some $u \notin 0^*$, $t(u) = b$. All accepting runs $\rho \in \text{AccRuns}(\mathcal{A})$ are such that $\rho(u) = q_a$. In other terms, AccRuns($\mathcal{A}$) is included in the set $\{\rho \in \text{Runs}(\mathcal{A}) \mid \rho(u) = q_a\}$, which has a null measure for μ_t. Hence, $\mu_t(\text{AccRuns}(\mathcal{A})) = 0$, which brings the contradiction.

We denote by w the ω-word over $\mathcal{A}$ corresponding to the left-most branch of t (i.e., $w = t(0^\omega)$). Consider the mapping p from AccRuns($\mathcal{A}$) to AccRuns($\mathcal{B}$) associating with a run $\rho \in \text{Runs}(\mathcal{A})$ the run $\rho(0^\omega)$.

For all partial run $\lambda = q_0q_1\ldots$ of $\mathcal{B}$, $p^{-1}(\text{Cyl}(\lambda))$ is a finite union of cylinders corresponding to the (balanced) partial runs of $\mathcal{A}$ of depth $|\lambda|$, which coincide with λ on their left-most branches. All these cylinders have measure 0 except for the one corresponding to the partial run η defined by $\eta(u) = q_{|u|}$ for $u \in 0^{\leq|\lambda|}$ and $\eta(u) = q_a$ otherwise. By construction of $\mathcal{A}$, $\mu_w(\text{Cyl}(\lambda)) = \mu_t(\text{Cyl}(\eta)) = \mu_t(p^{-1}(\text{Cyl}(\lambda))$. This implies that p is measurable and that the image of μ_t under p is equal to μ_w using Carathéodory's unique extension theorem.

The other direction admits a similar proof.

(2) The case of qualitative semantics directly follows from Proposition 43. Indeed, let $\mathcal{A}$ be a probabilistic co-Büchi tree automaton such that $L^{=1}_{\text{Qual}}(\mathcal{A})$ is equal to $\text{Path}^{=1}(L^{=1}(\mathcal{B}))$. Then $L^{=1}_{\text{Qual}}(\mathcal{A})$ is empty if and only if $L^{=1}(\mathcal{B})$ is empty. $\square$

4.4. Comparison with Regular Tree Languages and with Qualitative Tree Languages

We now discuss expressiveness. First, we exhibit a family of tree languages that are accepted by a co-Büchi probabilistic automaton but that are neither regular tree languages nor qualitative tree languages. On the other hand, we give an example of a qualitative tree language that no probabilistic automaton (regardless of its semantics) can accept and another example of a regular tree language that no probabilistic automaton (regardless of its semantics) can accept. Hence, this proves incomparability of probabilistic tree languages with both qualitative and regular tree languages.

For the first result, we consider, for all $0 < \lambda < 1$, the ω-word language L_λ over $\{\mathtt{a}, \mathtt{b}\}$ defined by

$$L_\lambda = \left\{ a^{k_1} b a^{k_2} b \ldots \mid k_1, k_2, \ldots > 0 \text{ s.t. } \prod_{i=1}^{\infty} (1 - \lambda^{k_i}) > 0 \right\}.$$

In Baier et al. [2008], L_λ is shown to be almost-surely accepted by a co-Büchi probabilistic automaton (actually they show that L_λ is positively accepted by a Büchi automaton; hence, as previously remarked in the introduction, it is accepted by a co-Büchi probabilistic automaton). Therefore, by Proposition 43, $\text{Path}^{=1}(L_\lambda)$ is a co-Büchi probabilistic qualitative tree language.

PROPOSITION 49. *For all* $0 < \lambda < 1$, $\mathrm{Path}^{=1}(L_\lambda)$ *is neither a regular tree language nor a qualitative tree language.*

PROOF. Let $0 < \lambda < 1$. We will show that $\mathrm{Path}^{=1}(L_\lambda)$ does not contain any regular tree. Hence, thanks to Theorem 2, it is not a regular tree language, and, thanks to Corollary 18, it is not a qualitative tree language.

Toward a contradiction, assume that L_λ contains a regular tree t_0. Consider the Markov chain $\mathcal{M}$ whose arena $\mathcal{G}$ is given by $(S, s_{\mathrm{ini}}, \zeta)$. The set of states S is the finite set of subtrees of t. The initial state s_{ini} is t_0. The transition function ζ is defined for all $t \in S$ by $\zeta(t)(t[0]) = \zeta(t)(t[1]) = \frac{1}{2}$ and 0 otherwise. Clearly, there exists a measure preserving bijection between the traces of $\mathcal{M}$ and the branchs of t_0.

To define the objective of the $\mathcal{M}$, consider the mapping ψ for S to $\{\mathtt{a}, \mathtt{b}\}$ associating to a state t the label of its root $t(\varepsilon)$. This mapping can canonically be extended to associate an ω-word over $\{\mathtt{a}, \mathtt{b}\}$ with each trace of $\mathcal{M}$. The objective $\mathcal{O}$ of $\mathcal{M}$ is the set of traces whose image under ψ belongs to L_λ.

As t_0 belongs to $\mathrm{Path}^{=1}(L_\lambda)$, $\mathcal{M}$ almost-surely fulfils its objective.

We claim that there exists some $k > 0$ such that the following event has a strictly positive probability: "The Markov chain from its current state reaches a state t with $\psi(t) = \mathtt{b}$ and in less than k steps another state t' with $\psi(t') = \mathtt{b}$."

Indeed, as t_0 belongs to $\mathrm{Path}^{=1}(L_\lambda)$, for all $t \in S$ there exists $u \sqsubseteq v$ such that $t(u) = t(v) = \mathtt{b}$. Hence, it suffices to take k to be the maximum over all t of the minimum of $|v| - |u|$ over all $u \sqsubseteq v$, such that $t(u) = t(v) = \mathtt{b}$.

By Borel-Cantelli lemma, this event occurs infinitely often along a trace with probability 1. In particular, this implies that with probability 1 for a trace θ with $\psi(\theta) = a^{k_1} b a^{k_2} b \ldots \mid k_1, k_2, \ldots > 0$, there are infinitely many k_i's that are equal to $k' \leq k$. Hence, $\psi(\theta)$ does not belongs to L_λ (as for all $n > 0$, $\prod_{i=1}^{\infty}(1 - \lambda^{k_i}) \leq (1 - \lambda^{k'})^n$). This contradicts the fact that $\mathcal{M}$ fulfils its objective with probability 1. □

Remark 50. Using the correspondence with POMDP introduced in Theorem 46, any co-Büchi automaton accepting $\mathrm{Path}^{=1}(L_\lambda)$ gives rise to an example of a co-Büchi POMDP $\mathbb{G}_{\mathcal{A}}$ in which Éloïse needs infinite memory to almost-surely win.

We now give an example of a qualitative tree language that no probabilistic tree automaton can accept. For this, we consider the language $\mathcal{L}_{\mathtt{a}}^{0\vee 1}$ of $\{\mathtt{a}, \mathtt{b}\}$-labeled trees such that either the left subtree or the right subtree of the root belongs to the language $\mathcal{L}_{\mathtt{a}}$ of example 7 (recall that $\mathcal{L}_{\mathtt{a}}$ is the language of $\{\mathtt{a}, \mathtt{b}\}$-labeled trees whose set of branches containing at least one $\mathtt{a}$ has measure 1). Formally, $\mathcal{L}_{\mathtt{a}}^{0\vee 1} = \{t \mid t[0] \in \mathcal{L}_{\mathtt{a}} \text{ or } t[1] \in \mathcal{L}_{\mathtt{a}}\}$. One easily verifies that $\mathcal{L}_{\mathtt{a}}^{0\vee 1}$ is a qualitative tree language. However, $\mathcal{L}_{\mathtt{a}}^{0\vee 1}$ cannot be recognized by a probabilistic tree automaton.

PROPOSITION 51. *The language* $\mathcal{L}_{\mathtt{a}}^{0\vee 1}$ *cannot be recognized by a probabilistic tree automaton (regardless of its semantics).*

PROOF. By contradiction, assume that there is some probabilistic parity tree automaton $\mathcal{A} = \langle\{\mathtt{a}, \mathtt{b}\}, Q, q_{ini}, \delta, \mathrm{Col}\rangle$ such that $\mathcal{L}_{\mathtt{a}}^{0\vee 1} = L_{\mathrm{Qual}}^{=1}(\mathcal{A})$ (the proof going exactly the same for the case where $\mathcal{L}_{\mathtt{a}}^{0\vee 1} = L^{=1}(\mathcal{A})$, we omit it). Let $t_{\mathtt{a}}$ (*resp.* $t_{\mathtt{b}}$) be the tree whose nodes are all labeled by $\mathtt{a}$ (*resp.* $\mathtt{b}$); that is, $t_{\mathtt{a}}(u) = \mathtt{a}$ (*resp.* $t_{\mathtt{b}}(u) = \mathtt{b}$) for all node $u \in \{0, 1\}^*$. Now let t_0 be the tree defined by $t_0(\varepsilon) = \mathtt{b}$, $t_0[0] = t_{\mathtt{a}}$ and $t_0[1] = t_{\mathtt{b}}$; and t_1 be the tree defined by $t_1(\varepsilon) = \mathtt{b}$, $t_1[0] = t_{\mathtt{b}}$ and $t_1[1] = t_{\mathtt{a}}$. Obviously, both t_0 and t_1 belong to $\mathcal{L}_{\mathtt{a}}^{0\vee 1} = L_{\mathrm{Qual}}^{=1}(\mathcal{A})$.

For any state $q \in Q$, let $\mathcal{A}_q = \langle\{\mathtt{a}, \mathtt{b}\}, Q, q, \delta, \mathrm{Col}\rangle$ be the automaton obtained from $\mathcal{A}$ by changing its initial state to be q. Then one easily checks that a tree t belongs to $L^{=1}_{\text{Qual}}(\mathcal{A})$ if and only if for all $(q_0, q_1) \in Q^2$ such that $\delta(q_{ini}, t(\varepsilon), q_0, q_1) > 0$ one has $t[0] \in L^{=1}_{\text{Qual}}(\mathcal{A}_{q_0})$ and $t[1] \in L^{=1}_{\text{Qual}}(\mathcal{A}_{q_1})$.

Now, for any $(q_0, q_1) \in Q^2$ such that $\delta(q_{ini}, \mathtt{b}, q_0, q_1) > 0$ one has that $t_{\mathtt{b}} \in L^{=1}_{\text{Qual}}(\mathcal{A}_{q_0})$ (because of $t_1 \in L^{=1}_{\text{Qual}}(\mathcal{A})$) and $t_{\mathtt{b}} \in L^{=1}_{\text{Qual}}(\mathcal{A}_{q_1})$ (because of $t_0 \in L^{=1}_{\text{Qual}}(\mathcal{A})$). Therefore, the tree t defined by $t(\varepsilon) = \mathtt{b}$ and $t[0] = t[1] = t_{\mathtt{b}}$ belongs to $L^{=1}_{\text{Qual}}(\mathcal{A})$, which leads to a contradiction as $t = t_{\mathtt{b}}$ and $t_{\mathtt{b}} \notin \mathcal{L}^{0\vee 1}_{\mathtt{a}} = L^{=1}_{\text{Qual}}(\mathcal{A})$. □

We now give an example of a regular tree language that no probabilistic tree automaton can accept. For this, let $t_{\mathtt{a}}$ (*resp.* $t_{\mathtt{b}}$) be the tree whose nodes are all labeled by $\mathtt{a}$ (*resp.* $\mathtt{b}$); that is, $t_{\mathtt{a}}(u) = \mathtt{a}$ (*resp.* $t_{\mathtt{b}}(u) = \mathtt{b}$) for all node $u \in \{0, 1\}^*$. Let t_0 be the tree defined by $t_0(\varepsilon) = \mathtt{b}$, $t_0[0] = t_{\mathtt{a}}$ and $t_0[1] = t_{\mathtt{b}}$; and t_1 be the tree defined by $t_1(\varepsilon) = \mathtt{b}$, $t_1[0] = t_{\mathtt{b}}$ and $t_1[1] = t_{\mathtt{a}}$. Finally, let $L_{0\vee 1} = \{t_0, t_1\}$: it is a regular tree language because it consists of two regular trees. However, $\mathcal{L}_{0\vee 1}$ cannot be recognized by a probabilistic tree automaton.

PROPOSITION 52. *The language $\mathcal{L}_{0\vee 1}$ cannot be recognized by a probabilistic tree automaton (regardless of its semantics).*

PROOF. One reasons by contradiction exactly as for Proposition 51 and concludes similarly that if there is a probabilistic automaton accepting $\mathcal{L}_{0\vee 1}$ then it also accepts tree $t_{\mathtt{b}} \notin \mathcal{L}_{0\vee 1}$, which leads to a contradiction. □

4.5. Variants

A natural variant is to replace the almost-sure acceptance condition on the set of runs by the *probable* one. That is, a tree t is ***probably accepted*** by $\mathcal{A}$ if $\mu_t(\text{AccRuns}(\mathcal{A})) > 0$.

Combining the conditions on the set of runs—almost-sure ($=1$) and probable (>0)—with the one on the set of accepting branches—qualitative (Qual) and positive (Pos)—we obtain four semantics for probabilistic tree automata denoted by $(>0, \text{Qual})$, $(>0, \text{Pos})$, $(=1, \text{Qual})$, and $(=1, \text{Pos})$ where the first component corresponds to the requirement on the set of accepting runs and the second to the requirement on the set of accepting branches of a run.

In Section 4, we mainly dealt with $(=1, \text{Qual})$-probabilistic automata that have a tight link with POMDP for the almost-sure winning condition (cf. Theorem 46). It can be shown that $(>0, \text{Pos})$-probabilistic automata share the same connection with POMDP with the *positive* winning condition. It implies that the emptiness problem for the $(>0, \text{Pos})$-probabilistic automata with a *co-Büchi* acceptance condition is ExpTime-complete.

When the two conditions were not of the same nature (as for the $(>0, \text{Qual})$ and $(=1, \text{Pos})$ semantics), we were unable to define a proper acceptance game.

We now briefly discuss properties of $(>0, \text{Pos})$-probabilistic automata. If $\mathcal{A}$ is a probabilistic tree automaton with an ω-regular acceptance condition, we denote by $\text{QualAccRuns}^{>0}(\mathcal{A})$ the set of positively accepting runs of $\mathcal{A}$ and by $L^{>0}_{\text{Pos}}(\mathcal{A})$ the set of trees accepted by $\mathcal{A}$ with $(>0, \text{Pos})$-semantics. The Proposition 53 (similar to Proposition 38) justifies of the definition of $L^{>0}_{\text{Pos}}(\mathcal{A})$.

PROPOSITION 53. *For all probabilistic tree automata $\mathcal{A}$ with an ω-regular acceptance condition, the set* $\text{QualAccRuns}^{>0}(\mathcal{A})$ *is measurable.*

PROOF. From Lemma 36, we have that $\text{AccPairs}(\mathcal{A}) = \{(\rho, \pi) \mid \rho(\pi) \in \text{Acc}\} = f_{\mathcal{A}}^{-1}(\{1\})$ is measurable in the product space. Hence, the numerical function $g : \text{Runs} \to [0, 1]$ associating to a run ρ the measure of its set of accepting branches (i.e., $\mu(\text{AccPairs}(\mathcal{A})_\rho)$

where $\mathrm{AccPairs}(\mathcal{A})_\rho = \{\lambda \in \mathrm{Br} \mid \rho(\lambda) \in \mathrm{Acc}\}$) is measurable (cf. Bauer [2001, Lemma 23.1, p. 135]). Because $\mathrm{QualAccRuns}^{>0}(\mathcal{A})$ is equal to $g^{-1}(]0, 1])$, it is measurable. □

The following proposition is an adaptation of Proposition 42 to the setting of $(>0, \mathrm{Pos})$-probabilistic automata.

PROPOSITION 54. *Let $\mathcal{A}$ be a probabilistic tree automaton with an ω-regular acceptance condition and let t be a tree. We have*

$$t \in L^{>0}_{\mathrm{Pos}}(\mathcal{A}) \Leftrightarrow \int f_{\mathcal{A}} d\mu_t \otimes \mu > 0.$$

PROOF. Because $f_{\mathcal{A}}$ is measurable (Lemma 36) by Tonelli's theorem [Bauer 2001, Theorem 23.6, p. 138], the mapping $g : \mathrm{Runs} \to [0, 1]$ associating to a run $\rho \in \mathrm{Runs}$, the value $\int_{\mathrm{Br}} f_{\mathcal{A}}(\rho, \cdot) d\mu$ is measurable.

t belongs to $L^{>0}_{\mathrm{Pos}}(\mathcal{A})$
iff $\mu_t(g^{-1}(\{]0, 1]\})) > 0$
iff $\int_{\mathrm{Runs}} g\, d\mu_t > 0$ — By [Bauer 2001, Theorem 13.2, p. 71]
iff $\int_{\mathrm{Runs}} \int_{\mathrm{Br}} f_{\mathcal{A}} d\mu d\mu_t > 0$ — by definition of g
iff $\int_{\mathrm{Runs}\times\mathrm{Br}} f_{\mathcal{A}} d\mu_t \otimes \mu > 0$ — by Tonelli's theorem [Bauer 2001, Theorem 23.6, p. 138]. □

Similarly to Proposition 43, we obtain a connection with ω-word languages.

For an ω-word language $L \subseteq \{\mathtt{a}, \mathtt{b}\}^\omega$, we denote by $\mathrm{Path}^{>0}(L)$ the set of trees labeled by $\{\mathtt{a}, \mathtt{b}\}$ with a positive set of branch labels in L (i.e., $\mu(\{\pi \in \mathrm{Br} \mid t(\pi) \in L\}) > 0$). In the following statement, for a probabilistic word automaton $\mathcal{B}$, we denote by $L^{>0}(\mathcal{B})$ the language positively accepted by $\mathcal{B}$.

PROPOSITION 55. *Given a probabilistic ω-word automaton $\mathcal{B}$ with an ω-regular acceptance condition, there exists a probabilistic tree automaton $\mathcal{A}$ with the same acceptance condition such that $L^{>0}_{\mathrm{Pos}}(\mathcal{A})$ is equal to* $\mathrm{Path}^{>0}(L^{>0}(\mathcal{B}))$.

PROOF. Let $\mathcal{B} = \langle A, Q, q_{ini}, \delta, \mathrm{Acc}\rangle$ be a (complete) probabilistic ω-word automaton with an ω-regular condition. Consider the probabilistic tree automaton $\mathcal{A}$ simulating $\mathcal{B}$ on all branches. Formally, $\mathcal{A}$ is equal to $\langle A, Q, q_{ini}, \delta', F\rangle$ where, for all $p, q \in Q$ and $x \in A$, $\delta'(p, x, q, q) = \delta(p, x, q)$.

Let t be a tree and let $g : \mathrm{Br} \mapsto [0, 1]$ be the mapping associating to a branch π the value $\int f_{\mathcal{A}}(\cdot, \pi) d\mu_t$. We have the following claim whose proof is identical to that of Claim 44 in Proposition 43.

CLAIM 56. *For any $\pi \in \mathrm{Br}$, $g(\pi) > 0$ if and only if $t(\pi)$ belongs to $L^{>0}(\mathcal{B})$.*

We are now ready to conclude:

$t \in L^{>0}_{\mathrm{Pos}}(\mathcal{A})$
$\Leftrightarrow \int f_{\mathcal{A}} d\mu_t \otimes \mu > 0$ — Proposition 54
$\Leftrightarrow \int g d\mu = 1$ — Tonelli's Theorem
$\Leftrightarrow g(\pi) > 0$ on a non-null set — [Bauer 2001, Theorem 13.2 , p. 71]
$\Leftrightarrow \mu(\{\pi \mid t(\pi) \in L^{>0}(\mathcal{B})\}) > 0$ — Claim 56
$\Leftrightarrow t \in \mathrm{Path}^{>0}(L^{>0}(\mathcal{B}))$. □

We can also transfer the results on the acceptance game (Proposition 45 and Theorem 46).

Table I. Decidability Status of the Emptiness Problem for the Different Types of Probabilistic Semantics

	$L^{=1}(\mathcal{A})$	$L^{=1}_{\text{Qual}}(\mathcal{A})$	$L^{>0}_{\text{Pos}}(\mathcal{A})$	$L^{=1}_{\text{Pos}}(\mathcal{A})$ / $L^{>0}_{\text{Qual}}(\mathcal{A})$
Büchi	Open	ExpTime-complete (Corollary 47)	Undecidable (Theorem 60)	Open
co-Büchi	Undecidable (Theorem 48)	Undecidable (Theorem 48)	ExpTime-complete (Corollary 59)	Open

Proposition 57. *Let $\mathcal{A}$ be a probabilistic tree automaton with an ω-regular acceptance condition and let t be a tree. We have $t \in L^{>0}_{\text{Pos}}(\mathcal{A})$ iff $\mathcal{M}_{\mathcal{A},t}$ positively fulfils its objective.*

Proof. Let $\text{AccPairs}(\mathcal{A}) = \{(\rho, \pi) \mid \rho(\pi) \in \text{Acc}\}$. By Proposition 54, to establish the desired equivalence, it is sufficient to show that $\mu_t \otimes \mu(\text{AccPairs}(\mathcal{A})) = \mu_{\mathcal{M}_{\mathcal{A},t}}(\mathcal{O}_{\text{Acc}})$. The latter was done already in the proof of Proposition 45. □

Theorem 58. *Let $\mathcal{A}$ be a probabilistic tree automaton with an ω-regular acceptance condition. The language $L^{>0}_{\text{Pos}}(\mathcal{A})$ is nonempty if and only if Éloïse positively wins in $\mathbb{G}_{\mathcal{A}}$.*

Proof. The proof only differs from the one of Theorem 46 in its conclusion. Namely, Éloïse positively wins in $\mathbb{G}_{\mathcal{A}}$ iff there is some t such that $\text{Val}(\mathbb{G}_{\mathcal{A},t}) > 0$ iff $t \in L^{>0}_{\text{Pos}}(\mathcal{A})$ (thanks to Proposition 57). □

If we consider a co-Büchi acceptance condition, this leads to a decidability result for the emptiness problem (which is a dual version of Corollary 47).

Corollary 59. *Let $\mathcal{A}$ be a probabilistic co-Büchi tree automaton. Deciding emptiness of $L^{>0}_{\text{Pos}}(\mathcal{A})$ is an* ExpTime*-complete problem. Moreover, if $L^{>0}_{\text{Pos}}(\mathcal{A}) \neq \emptyset$, it contains a regular tree.*

Proof. The ExpTime upper bound follows from the polynomial time reduction to deciding positive winning in a co-Büchi POMDP. The proof of Theorem 58 together with the fact that a *finite memory* positively winning strategy always exists in POMDP imply the existence of a regular tree when $L^{>0}_{\text{Pos}}(\mathcal{A}) \neq \emptyset$.

The lower bound can be establish as for Corollary 47 by reduction of the emptiness problem of probabilistic co-Büchi ω-word automata with the probable semantics (which is ExpTime-complete [Baier et al. 2008; Baier et al. 2012]). For this, one first establishes a dual version of Proposition 43 and then concludes as in Corollary 47. □

Finally, as a dual version of Theorem 48, we show that the emptiness problem for a Büchi acceptance condition is undecidable.

Theorem 60. *The following problem is undecidable: given a probabilistic Büchi tree automaton $\mathcal{A}$, decide if $L^{>0}_{\text{Pos}}(\mathcal{A}) = \emptyset$.*

Proof. The proof proceeds by reduction to the undecidability of the emptiness problem for Büchi ω-word automata (with the positive acceptance).

Let $\mathcal{B} = \langle A = \{\mathtt{a}, \mathtt{b}\}, Q, q_{ini}, \delta, F\rangle$ be a probabilistic Büchi ω-word automaton. By Proposition 55, we can construct a probabilistic Büchi tree automaton $\mathcal{A}$ such that:

$$L^{>0}_{\text{Pos}}(\mathcal{A}) = \text{Path}^{>0}(L^{>0}(\mathcal{B})).$$

In particular, $L^{>0}_{\text{Pos}}(\mathcal{A})$ is empty if and only if $L^{>0}(\mathcal{B})$ is empty as well, which allows us to conclude. □

We terminate this Section with Table I, which summarizes the (un)decidability results and open questions on the emptiness problem for the different types of probabilistic semantics that we considered.

5. CONCLUSION

The first main contribution of this article was to introduce new acceptance criteria for nondeterministic automata on infinite trees: qualitative and positive acceptance. Both criteria define (incomparable) classes of languages with desirable closure and decidability properties (in particular, a polynomial time emptiness test). In addition, it leads to a tight connection with (finite) Markov decision processes in a similar flavor as regular tree automata do with two-player games.

The second main contribution was to define suitable notions of probabilistic automata. In particular, we showed that for Büchi (=1, Qual)-probabilistic automata, as well as for co-Büchi (>0, Pos)-probabilistic automata, emptiness is an EXPTIME-complete problem.

Unsurprisingly, there remain several open questions. Some of them are purely theoretical (mainly regarding expressiveness and decidability status of the emptiness problem for some variants), but the most pressing one concerns potential applications of this work. A quick answer to this latter challenge is to rely on the tight connection between qualitative tree languages and Markov decision processes as exposed in Section 3.5. Because these two objects are essentially the same one seen from two different perspectives (the qualitative tree languages being a sort of unfolding of a finite MDP), one can, for instance, rely on the modeling work made using MDP (see e.g., Baier and Katoen [2008, Chapter 10] for many valuable examples) to argue that qualitative tree languages are equally useful for such a purpose. However, the setting of MDP seems simpler for modeling purpose because it is closer to real systems. Concerning the probabilistic setting, the potential applications are not yet clear. Due to their incomparability with both regular and qualitative tree languages, we cannot simply extend existing applications. But this is also encouraging because it suggests potentially new applications. The hard part is that it mixes (for the qualitative semantics) two orthogonal notions of measure: the one on the run and the other the branches; if the one on branches has a simple interpretation (one looks at all possible executions of a system), the one on runs is trickier to interpret (in a sense, it speaks of the set of all outputs of a machine that processes the unfolding of a system).

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their thorough reviews and very useful comments that helped improve the clarity and the relevance of this article.

This work was supported by the following projects: AMIS (ANR 2010 JCJC 0203 01 AMIS) and FREC (ANR 2010 BLAN 0202 02 FREC).

REFERENCES

Christel Baier, Nathalie Bertrand, and Marcus Marcus Größer. 2008. On decision problems for probabilistic Büchi automata. In *Proceedings of the 11th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS'08).* Springer, 287–301.

Christel Baier, Marcus Größer, and Nathalie Bertrand. 2012. Probabilistic ω-automata. *Journal of the Association of Computing Machinery* 59, 1 (2012), 1.

Christel Baier and Marcus Marcus Größer. 2005. Recognizing omega-regular languages with probabilistic automata. In *Proceedings of the 20th IEEE Symposium on Logic in Computer Science (LiCS'05)*. IEEE Computer Society, 137–146.

Christel Baier and Joost-Pieter Katoen. 2008. *Principles of Model Checking (Representation and Mind Series)*. MIT Press.

Heinz Bauer. 1996. *Probability Theory*. Walter de Gruyter.

Heinz Bauer. 2001. *Measure and Integration Theory*. Walter de Gruyter.

Danièle Beauquier, Maurice Nivat, and Damian Niwinski. 1991. About the effect of the number of successful paths in an infinite tree on the recognizability by a finite automaton with Büchi conditions. In *Proceedings of Fundamentals of Computation Theory, 8th International Symposium (FCT'91).* Springer, 136–145.

Danièle Beauquier and Damian Niwiński. 1995. Automata on infinite trees with path counting constraints. *Information and Computation* 120, 1 (1995), 117–125.

Laurent Bienvenu. 2008. *Caractérisations de l'aléatoire par les jeux: impredictibilité et stochasticité*. Ph.D. Dissertation. Université de Provence.

Rohit Chadha, A. Prasad Sistla, and Mahesh Viswanathan. 2009. Power of randomization in automata on infinite strings. In *Proceedings of the 20th International Conference on Concurrency Theory (CONCUR'09)*. Springer, 229–243.

Krishnendu Chatterjee, Laurent Doyen, and Thomas A. Henzinger. 2009. Probabilistic weighted automata. In *Proceedings of the 20th International Conference on Concurrency Theory (CONCUR'09)*. Springer, 244–258.

Krishnendu Chatterjee, Marcin Jurdziński, and Thomas A. Henzinger. 2004. Quantitative stochastic parity games. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2004)*. SIAM, 121–130.

Alonzo Church. 1962. Logic, arithmetic and automata. In *Proceedings of the International Congress of Mathematicians*. 23–35.

Costas Courcoubetis and Mihalis Yannakakis. 1990. Markov decision processes and regular events (Extended Abstract). In *Proceedings of the 17th International Colloquium on Automata, Languages, and Programming (ICALP'90)*. Springer, 336–349.

E. Allen Emerson, Charanjit S. Jutla, and A. Prasad Sistla. 1993. On model-checking for fragments of μ-calculus. In *Proceedings of the 5th International Conference on Computer Aided Verification (CAV'93)*. Springer, 385–396.

Erich Grädel, Wolfgang Thomas, and Thomas Wilke (Eds.). 2002. *Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001]*. Lecture Notes in Computer Science, Vol. 2500. Springer.

Yuri Gurevich and Leo Harrington. 1982. Trees, automata, and games. In *Proceedings of the 14th Annual ACM Symposium on Theory of Computing (STOC'82)*. ACM, 60–65.

Shizuo Kakutani. 1948. On equivalence of infinite product measures. *Annals of Mathematics* 49 (1948), 214–224.

Christof Löding. 2011. Infinite games and automata theory. In *Lectures in Game Theory for Computer Scientists*, K. Apt and E. Grädel (Eds.). Cambridge University Press, 38–73.

Andrei A. Muchnik, Alexei L. Semenov, and Vladimir A. Uspensky. 1998. Mathematical metaphysics of randomness. *Theoretical Computer Science* 207, 2 (1998), 263–317.

A. Paz. 1971. *Introduction to Probabilistic Automata*. Academic Press.

Dominique Perrin and Jean-Éric Pin. 2004. *Infinite Words*. Academic Press.

Michael O. Rabin. 1963. Probabilistic automata. *Information and Control* 6, 3 (1963), 230–245.

Michael O. Rabin. 1969. Decidability of second-order theories and automata on infinite trees. *Transaction of the American Mathematical Society* 141 (1969), 1–35.

Michael O. Rabin. 1972. *Automata on Infinite Objects and Church's problem*. American Mathematical Society, Providence, R.I. iii+22 pages. Conference Board of the Mathematical Sciences Regional Conference Series in Mathematics, No. 13.

Robert D. Reisz. 1999. Decomposition theorems for probabilistic automata over infinite objects. *Informatica, Lithiuanian Academy of Science* 10, 4 (1999), 427–440.

Wolfgang Thomas. 1997. Languages, automata, and logic. In *Handbook of Formal Language Theory*, G. Rozenberg and A. Salomaa (Eds.). Vol. III. Springer, 389–455.

Moshe Y. Vardi and Thomas Wilke. 2007. Automata: From logics to algorithms. In *Logic and Automata: History and Perspectives*. Amsterdam University Press, 629–736.

Received April 2013; revised November 2013; accepted February 2014