Power of Randomization in Automata on Infinite Strings

Rohit Chadha, A. Prasad Sistla, and Mahesh Viswanathan

[1] Univ. of Illinois at Urbana-Champaign
[2] Univ. of Illinois at Chicago

Abstract. Probabilistic Büchi Automata (PBA) are randomized, finite state automata that process input strings of infinite length. Based on the threshold chosen for the acceptance probability, different classes of languages can be defined. In this paper, we present a number of results that clarify the power of such machines and properties of the languages they define. The broad themes we focus on are as follows. We precisely characterize the complexity of the emptiness, universality, and language containment problems for such machines, answering canonical questions central to the use of these models in formal verification. Next, we characterize the languages recognized by PBAs topologically, demonstrating that though general PBAs can recognize languages that are not regular, topologically the languages are as simple as ω-regular languages. Finally, we introduce Hierarchical PBAs, which are syntactically restricted forms of PBAs that are tractable and capture exactly the class of ω-regular languages.

1 Introduction

Automata on infinite (length) strings have played a central role in the specification, modeling and verification of non-terminating, reactive and concurrent systems [20, 10, 21, 8, 17]. However, there are classes of systems whose behavior is probabilistic in nature; the probabilistic behavior being either due to the employment of randomization in the algorithms executed by the system or due to other uncertainties in the system, such as failures, that are modeled probabilistically. While Markov Chains and Markov Decision Processes have been used to model such behavior in the formal verification community [15], both these models do not adequately capture *open*, *reactive* probabilistic systems that continuously accept inputs from an environment. The most appropriate model for such systems are probabilistic automata on infinite strings, which are the focus of study in this paper.

Probabilistic Büchi Automata (PBA) have been introduced in [3] to capture such computational devices. These automata generalize probabilistic finite automata (PFA) [14, 16, 12] from finite length inputs to infinite length inputs. Informally, PBA's are like finite-state automata except that they differ in two respects. First, from each state and on each input symbol, the PBA may roll a dice to determine the next state. Second, the notion of acceptance is different because PBAs are probabilistic in nature and have infinite length input strings. The behavior of a PBA on a given infinite input string can be captured by an infinite Markov chain that defines a probability measure on the space of runs/executions of the machine on the given input. Like Büchi automata, a run is considered to be accepting if some accepting state occurs infinitely often, and therefore, the probability of acceptance of the input is defined to be the measure of all accepting runs on the given input. There are two possible languages that one can associate with a PBA $\mathcal{B}$ [3, 2] — $\mathcal{L}_{>0}(\mathcal{B})$ (called *probable semantics*) consisting of all strings whose probability of acceptance is non-zero, and $\mathcal{L}_{=1}(\mathcal{B})$ (called *almost sure semantics*) consisting all strings whose probability of acceptance is 1. Based on these two languages, one can define two classes of languages — $\mathbb{L}(\mathrm{PBA}^{>0})$, and $\mathbb{L}(\mathrm{PBA}^{=1})$ which are the collection of all languages (of infinite length strings) that can be accepted by some PBA with respect to probable, and almost sure semantics, respectively. In this paper we study the expressive power of, and decision problems for these classes of languages.

We present a number of new results that highlight three broad themes. First, we establish the precise complexity of the canonical decision problems in verification, namely, emptiness, universality, and language containment, for the classes $\mathbb{L}(\mathrm{PBA}^{>0})$ and $\mathbb{L}(\mathrm{PBA}^{=1})$. For the decision problems, we focus our attention on RatPBAs which are PBAs in which all transition probabilities are rational. First we show the problem of checking emptiness of the language $\mathcal{L}_{=1}(\mathcal{B})$ for a RatPBA $\mathcal{B}$ is **PSPACE**-complete, which substantially improves the result of [2] where it was shown to be decidable in **EXPTIME** and conjectured to be **EXPTIME**-hard. This upper bound is established by observing that the complement of the language $\mathcal{L}_{=1}(\mathcal{B})$ is recognized by a special PBA $\mathcal{M}$ (with probable semantics) called a *finite state probabilistic monitor* (FPM) [4, 5] and then exploiting a result in [5] that shows that the language of an FPM is non-empty if and only if there is an *ultimately periodic word* in the language. This observation of the existence of ultimately periodic words does not carry over to the class $\mathbb{L}(\mathrm{PBA}^{>0})$. However, we show that $\mathcal{L}_{>0}(\mathcal{B})$, for a RatPBA $\mathcal{B}$, is non-empty iff it contains a *strongly asymptotic word*, which is a generalization of ultimately periodic word. This allows us to show that the emptiness problem for $\mathbb{L}(\mathrm{PBA}^{>0})$, though undecidable as originally shown in [2], is Σ^0_2-complete, where Σ^0_2 is a set in the second level of the arithmetic hierarchy. Next we show that the universality problems for $\mathbb{L}(\mathrm{PBA}^{>0})$ and $\mathbb{L}(\mathrm{PBA}^{=1})$ are also Σ^0_2-complete and **PSPACE**-complete, respectively. Finally, we show that for both $\mathbb{L}(\mathrm{PBA}^{>0})$ and $\mathbb{L}(\mathrm{PBA}^{=1})$, the language containment problems are Σ^0_2-complete. This is another surprising observation — given that emptiness and universality are both in **PSPACE** for $\mathbb{L}(\mathrm{PBA}^{=1})$, one would expect language containment to be at least decidable.

The second theme brings to sharper focus the correspondence between nondeterminism and probable semantics, and between determinism and almost sure semantics, in the context of automata on infinite words. This correspondence was hinted at in [2]. There it was observed that $\mathbb{L}(\mathrm{PBA}^{=1})$ is a strict subset of $\mathbb{L}(\mathrm{PBA}^{>0})$ and that while Büchi, Rabin and Streett acceptance conditions all yield the same class of languages under the probable semantics, they yield different classes of languages under the almost sure semantics. These observations mirror the situation in non-probabilistic automata — languages recognized by deterministic Büchi automata are a strict subset of the class of languages recognized by nondeterministic Büchi automata, and while Büchi, Rabin and Streett acceptances are equivalent for nondeterministic machines, Büchi acceptance is strictly weaker than Rabin and Streett for deterministic machines. In this paper we further strenghten this correspondence through a number of results on the closure properties as well as the topological structure of $\mathbb{L}(\mathrm{PBA}^{>0})$ and $\mathbb{L}(\mathrm{PBA}^{=1})$.

First we consider closure properties. It was shown in [2] that the class $\mathbb{L}(\mathrm{PBA}^{>0})$ is closed under all the Boolean operations (like the class of languages recognized by nondeterministic Büchi automata) and that $\mathbb{L}(\mathrm{PBA}^{=1})$ is not closed under complementation. We show that $\mathbb{L}(\mathrm{PBA}^{=1})$ is, however, closed under intersection and union, just like the class of languages recognized by deterministic Büchi automata. We also show that every language in $\mathbb{L}(\mathrm{PBA}^{>0})$ is a Boolean combination of languages in $\mathbb{L}(\mathrm{PBA}^{=1})$, exactly like every ω-regular language (or languages recognized by nondeterministic Büchi machines) is a Boolean combination of languages recognized by deterministic Büchi machines. Next, we characterize the classes topologically. There is natural topological space on infinite length strings called the *Cantor topology* [18]. We show that, like ω-regular languages, all the classes of languages defined by PBAs lie in very low levels of this Borel hierarchy. We show that $\mathbb{L}(\mathrm{PBA}^{=1})$ is strictly contained in $\mathcal{G}_\delta$, just like the class of languages recognized by deterministic Büchi is strictly contained in $\mathcal{G}_\delta$. From these results, it follows that $\mathbb{L}(\mathrm{PBA}^{>0})$ is in the Boolean closure of $\mathcal{G}_\delta$ much like the case for ω-regular languages.

The last theme identifies syntactic restrictions on PBAs that capture regularity. Much like PFAs for finite word languages, PBAs, though finite state, allow one to recognize non-regular languages. It has been shown [3, 2] that both $\mathbb{L}(\mathrm{PBA}^{>0})$ and $\mathbb{L}(\mathrm{PBA}^{=1})$ contain non-ω-regular languages. A question initiated in [3] was to identify restrictions on PBAs that ensure that PBAs have the same expressive power as finite-state (non-probabilistic) machines. One such restriction was identified in [3], where it was shown that *uniform* PBAs with respect to the probable semantics capture exactly the class of ω-regular languages. However, the uniformity condition identified by Baier et. al. was semantic in nature. In this paper, we identify one simple syntactic restriction that capture regularity both for probable semantics and almost sure semantics. The restriction we consider is that of a hierarchical structure. A *Hierarchical* PBA *(*HPBA*)* is a PBA whose states are partitioned into different levels such that, from any state q, on an input symbol a, at most one transition with non-zero probability goes to a state at the same level as q and all others go to states at higher level. We show that HPBA with respect to probable semantics define exactly the class of ω-regular languages, and with respect to almost sure semantics define exactly the class of ω-regular languages in $\mathbb{L}(\mathrm{PBA}^{=1})$, namely, those recognized by deterministic Büchi automata. Next, HPBAs not only capture the notion of regularity, they are also very tractable. We show that the emptiness and universality problems for HPBA with probable semantics are **NL**-complete and **PSPACE**-complete, respectively; for almost sure semantics, emptiness is **PSPACE**-complete and universality is **NL**-complete. This is interesting because this is the exact same complexity as that for (non-probabilistic) Büchi automata. In contrast, the emptiness problem for uniform PBA has been shown to be in **EXPTIME** and co-**NP**-hard [3]; thus, they seem to be less tractable than HPBA.

The rest of the paper is organized as follows. After discussing closely related work, we start with some preliminaries (in Section 2) before introducing PBAs. We present our results about the probable semantics in Section 3, and almost sure semantics in Section 4. Hierarchical PBAs are introduced in Section 5, and conclusions are presented in Section 6. In the interest of space, some proofs have been deferred to the Appendix.

Related Work. Probabilistic Büchi automata (PBA), introduced in [3], generalize the model of Probabilistic Finite Automata [14, 16, 12] to consider inputs of infinite length. In [3], Baier and Größer only considered the probable semantics for PBA. They also introduced the model of uniform PBAs to capture ω-regular languages and showed that the emptiness problem for such machines is in **EXPTIME** and co-**NP**-hard. The almost sure semantics for PBA was first considered in [2] where a number of results were established. It was shown that $\mathbb{L}(\mathrm{PBA}^{>0})$ are closed under all Boolean operations, $\mathbb{L}(\mathrm{PBA}^{=1})$ is strictly contained in $\mathbb{L}(\mathrm{PBA}^{>0})$, the emptiness problem for $\mathbb{L}(\mathrm{PBA}^{>0})$ is undecidable, and the emptiness problem of $\mathbb{L}(\mathrm{PBA}^{=1})$ is in **EXPTIME**. We extend and sharpen the results of this paper. In a series of previous papers [4, 5], we considered a special class of PBAs called FPMs (Finite state Probabilistic Monitors) whose accepting set of states consists of all states excepting a rejecting state which is also absorbing. There we proved a number of results on the expressiveness and decidability/complexity of problems for FPMs. We draw on many of these observations to establish new results for the more general model of PBAs.

2 Preliminaries

We assume that the reader is familiar with arithmetical hierarchy (for the sake of convenience of the reader, we have also introduced them in Appendix A). The set of natural numbers will be denoted by $\mathbb{N}$, the closed unit interval by $[0, 1]$ and the open unit interval by $(0, 1)$. The power-set of a set X will be denoted by 2^X.

Sequences. Given a finite set S, $|S|$ denotes the cardinality of S. Given a sequence (finite or infinite) $\kappa = s_0, s_1, \ldots$ over S, $|\kappa|$ will denote the length of the sequence (for infinite sequence $|\kappa|$ will be ω), and $\kappa[i]$ will denote the ith element s_i of the sequence. As usual S^* will denote the set of all finite sequences/strings/words over S, S^+ will denote the set of all finite sequences/strings/words over S and S^ω will denote the set of all infinite sequences/strings/words over S. Given $\eta \in S^*$ and $\kappa \in S^* \cup S^\omega$, $\eta\kappa$ is the sequence obtained by concatenating the two sequences in order. Given $\mathsf{L}_1 \subseteq \Sigma^*$ and $\mathsf{L}_2 \subseteq \Sigma^\omega$, the set $\mathsf{L}_1\mathsf{L}_2$ is defined to be $\{\eta\kappa \mid \eta \in \mathsf{L}_1 \text{ and } \kappa \in \mathsf{L}_2\}$. Given natural numbers $i, j \leq |\kappa|$, $\kappa[i : j]$ is the finite sequence $s_i, \ldots s_j$, where $s_k = \kappa[k]$. The set of *finite prefixes* of κ is the set $\mathit{Pref}(\kappa) = \{\kappa[0, j] \mid j \in \mathbb{N}, j \leq |\kappa|\}$.

Languages of infinite words. A language L of infinite words over a finite alphabet Σ is a subset of Σ^ω. (Please note we restrict only to finite alphabets). A set of languages of infinite words over Σ is said to be a class of languages of infinite words over Σ. Given a class $\mathcal{L}$, the Boolean closure of $\mathcal{L}$, denoted $\mathsf{BCl}(\mathcal{L})$, is the smallest class containing $\mathcal{L}$ that is closed under the Boolean operations of complementation, union and intersection.

Automata and ω-regular Languages. A *finite automaton on infinite words* $\mathcal{A}$ over a (finite) alphabet Σ is a tuple (Q, q_0, F, Δ), where Q is a finite set of states, $\Delta \subseteq Q \times \Sigma \times Q$ is the transition relation, $q_0 \in Q$ is the initial state, and F defines the accepting condition. The nature of F depends on the type of automaton we are considering; for a *Büchi automaton* $F \subseteq Q$, while for a *Rabin automaton* F is a finite subset of $2^Q \times 2^Q$. If for every $q \in Q$ and $a \in \Sigma$, there is exactly one q' such that $(q, a, q') \in \Delta$ then $\mathcal{A}$ is called a *deterministic* automaton. Let $\alpha = a_0, a_1, \ldots$ be an infinite string over Σ.

A *run* r *of* $\mathcal{A}$ on α is an infinite sequence $s_0, s_1, \ldots$ over Q such that $s_0 = q_0$ and for every $i \geq 0$, $(s_i, a_i, s_{i+1}) \in \Delta$. The notion of an *accepting run* depends on the type of automaton we consider. For a Büchi automaton r is accepting if some state in F appears infinitely often in r. On the other hand for a Rabin automaton, r is accepting if it satisfies the *Rabin acceptance condition*— there is some pair $(B_i, G_i) \in F$ such that all the states in B_i appear only finitely many times in r, while at least one state in G_i appears infinitely many times. The automaton $\mathcal{A}$ *accepts* the string α if it has an accepting run on α. The *language accepted (recognized) by* $\mathcal{A}$, denoted by $\mathcal{L}(\mathcal{A})$, is the set of strings that $\mathcal{A}$ accepts. A language $\mathsf{L} \subseteq \Sigma^\omega$ is called *ω-regular* iff there is some Büchi automata $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}) = \mathsf{L}$. In this paper, given a fixed alphabet Σ, we will denote the class of ω-regular languages by $\mathsf{Regular}$. It is well-known that unlike the case of finite automata on finite strings, deterministic Büchi automata are less powerful than nondeterministic Büchi automata. On the other hand, nondeterministic Rabin automata and deterministic Rabin automata have the expressive power and they recognize exactly the class $\mathsf{Regular}$. Finally, we will sometimes find it convenient to consider automata $\mathcal{A}$ that do not have finitely many states. We will say that a language L is *deterministic* iff it can be accepted by a deterministic Büchi automaton that does not necessarily have finitely many states. We denote by $\mathrm{Deterministic}$ the collection of all deterministic languages. Please note that the class $\mathrm{Deterministic}$ strictly contains the class of languages recognized by finite state deterministic Büchi automata. The following are well-known results [13, 18].

Proposition 1. $\mathsf{L} \in \mathsf{Regular} \cap \mathrm{Deterministic}$ iff there is a finite state deterministic Büchi automaton $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}) = \mathsf{L}$. Furthermore, $\mathsf{Regular} \cap \mathrm{Deterministic} \subsetneq \mathsf{Regular}$ and $\mathsf{Regular} = \mathsf{BCl}(\mathsf{Regular} \cap \mathrm{Deterministic})$.

Topology on infinite strings. The set Σ^ω comes equipped with a natural topology called the *Cantor topology*. The collection of open sets is the collection $\mathcal{G} = \{\mathsf{L}\Sigma^\omega \mid \mathsf{L} \subseteq \Sigma^+\}$.[3] The collection of closed sets, $\mathcal{F}$, is the collection of *prefix-closed sets* — L is prefix-closed if for every infinite string α, if every prefix of α is a prefix of some string in L, then α itself is in L. In the context of verification of reactive systems, closed sets are also called *safety languages* [11, 1]. One remarkable result in automata theory is that the class of languages $\mathcal{G}_\delta$ coincides exactly with the class of languages recognized by infinite-state deterministic Büchi automata [13, 18]. This combined with the fact that the class of ω-regular languages is the Boolean closure of ω-regular deterministic Büchi automata yields that the class of ω-regular languages is strictly contained in $\mathsf{BCl}(\mathcal{G}_\delta)$ which itself is strictly contained in $\mathcal{G}_{\delta\sigma} \cap \mathcal{F}_{\sigma\delta}$ [13, 18]. (For a precise definition of the sets $\mathcal{G}_\delta$ and $\mathcal{G}_{\delta\sigma} \cap \mathcal{F}_{\sigma\delta}$, please see Appendix A).

Proposition 2. $\mathcal{G}_\delta = \mathrm{Deterministic}$, and $\mathsf{Regular} \subsetneq \mathsf{BCl}(\mathcal{G}_\delta) \subsetneq \mathcal{G}_{\delta\sigma} \cap \mathcal{F}_{\sigma\delta}$.

2.1 Probabilistic Büchi automata

We shall now recall the definition of probabilistic Büchi automata given in [3]. Informally, PBA's are like finite-state deterministic Büchi automata except that the transition

[3] This topology is also generated by the metric $d : \Sigma^\omega \times \Sigma^\omega \to [0, 1]$ where $d(\alpha, \beta)$ is 0 iff $\alpha = \beta$; otherwise it is $\frac{1}{2^i}$ where i is the smallest integer such that $\alpha[i] \neq \beta[i]$.

function from a state on a given input is described as a probability distribution that determines the probability of the next state. PBAs generalize the probabilistic finite automata (PFA) [14, 16, 12] on finite input strings to infinite input strings. Formally,

Definition: A *finite state probabilistic Büchi automata (*PBA*)* over a finite alphabet Σ is a tuple $\mathcal{B} = (Q, q_s, Q_f, \delta)$ where Q is a finite set of *states*, $q_s \in Q$ is the *initial state*, $Q_f \subseteq Q$ is the set of *accepting/final states*, and $\delta : Q \times \Sigma \times Q \to [0, 1]$ is the *transition relation* such that for all $q \in Q$ and $a \in \Sigma$, $\sum_{q' \in Q} \delta(q, a, q') = 1$. In addition, if $\delta(q, a, q')$ is a rational number for all $q, q' \in Q, a \in \Sigma$, then we say that $\mathcal{M}$ is a rational probabilistic Büchi automata (RatPBA).

Notation: The transition function δ of PBA $\mathcal{B}$ on input a can be seen as a square matrix δ_a of order $|Q|$ with the rows labeled by "current" state, columns labeled by "next state" and the entry $\delta_a(q, q')$ equal to $\delta(q, a, q')$. Given a word $u = a_0 a_1 \dots a_n \in \Sigma^+$, δ_u is the matrix product $\delta_{a_0}\delta_{a_1} \dots \delta_{a_n}$. For an empty word $\epsilon \in \Sigma^*$ we take δ_ϵ to be the identity matrix. Finally for any $Q_0 \subseteq Q$, we say that $\delta_u(q, Q_0) = \sum_{q' \in Q_0} \delta_u(q, q')$. Given a state $q \in Q$ and a word $u \in \Sigma^+$, $\mathsf{post}(q, u) = \{q' \mid \delta_u(q, q') > 0\}$.

Intuitively, the PBA starts in the initial state q_s and if after reading $a_0, a_1 \dots, a_n$ results in state q, then it moves to state q' with probability $\delta_{a_{i+1}}(q, q')$ on symbol a_{i+1}. Given a word $\alpha \in \Sigma^\omega$, the PBA $\mathcal{B}$ can be thought of as a infinite state Markov chain which gives rise to the standard σ-algebra defined using cylinders and the standard probability measure on Markov chains [19, 9]. (See Appendix A for the formal defintion). We denote this measure by $\mu_{\mathcal{B},\alpha}$. A *run* of the PBA $\mathcal{B}$ is an infinite sequence $\rho \in Q^\omega$. A run ρ is *accepting* if $\rho[i] \in Q_f$ for infinitely many i. A run ρ is said to be *rejecting* if it is not accepting. The set of accepting runs and the set of rejecting runs are measurable [19]. Given a word α, the measure of the set of accepting runs is said to be the *probability of accepting* α and is henceforth denoted by $\mu^{acc}_{\mathcal{B},\alpha}$; and the measure of the set of rejecting runs is said to be the *probability of rejecting* α and is henceforth denoted by $\mu^{rej}_{\mathcal{B},\alpha}$. Clearly $\mu^{acc}_{\mathcal{B},\alpha} + \mu^{rej}_{\mathcal{B},\alpha} = 1$. Following, [3, 2], a PBA $\mathcal{B}$ on alphabet Σ defines two *semantics*:

- $\mathcal{L}_{>0}(\mathcal{B}) = \{\alpha \in \Sigma^\omega \mid \mu^{acc}_{\mathcal{B},\alpha} > 0\}$, henceforth referred to as the *probable semantics* of $\mathcal{B}$, and
- $\mathcal{L}_{=1}(\mathcal{B}) = \{\alpha \in \Sigma^\omega \mid \mu^{acc}_{\mathcal{B},\alpha} = 1\}$, henceforth referred to as the *almost-sure semantics* of $\mathcal{B}$.

This gives rise to the following classes of languages of infinite words.

Definition: Given a finite alphabet Σ, $\mathbb{L}(\text{PBA}^{>0}) = \{\mathsf{L} \subseteq \Sigma^\omega \mid \exists \text{PBA } \mathcal{B}.\ \mathsf{L} = \mathcal{L}_{>0}(\mathcal{B})\}$ and $\mathbb{L}(\text{PBA}^{=1}) = \{\mathsf{L} \subseteq \Sigma^\omega \mid \exists \text{PBA } \mathcal{B}.\ \mathsf{L} = \mathcal{L}_{=1}(\mathcal{B})\}$.

Remark: Given $x \in [0, 1]$, one can of course, also define the languages $\mathcal{L}_{>x}(\mathcal{B}) = \{\alpha \in \Sigma^\omega \mid \mu^{acc}_{\mathcal{B},\alpha} > x\}$ and $\mathcal{L}_{\geq x}(\mathcal{B}) = \{\alpha \in \Sigma^\omega \mid \mu^{acc}_{\mathcal{B},\alpha} \geq x\}$. The exact value of x is not important and thus one can also define classes $\mathbb{L}(\text{PBA}^{>\frac{1}{2}})$ and $\mathbb{L}(\text{PBA}^{\geq\frac{1}{2}})$.

Probabilistic Rabin automaton. Analogous to the definition of a PBA and RatPBA, one can define a Probabilistic Rabin automaton PRA and RatPRA [2, 7]; where instead of using a set of final states, a set of pairs of subsets of states is used. A run in that case is said to be accepting if it satisfies the Rabin acceptance condition. It is shown in [2, 7] that PRAs have the same expressive power under both probable and almost-sure semantics. Furthermore, it is shown in [2, 7] that for any PBA $\mathcal{B}$, there is PRA $\mathcal{R}$ such that a word α is accepted by $\mathcal{R}$ with probability 1 iff α is accepted by $\mathcal{B}$ with probability > 0. All other words are accepted with probability 0 by $\mathcal{R}$.

Proposition 3 ([2]). For any PBA $\mathcal{B}$ there is a PRA $\mathcal{R}$ such that $\mathcal{L}_{>0}(\mathcal{B}) = \mathcal{L}_{>0}(\mathcal{R}) = \mathcal{L}_{=1}(\mathcal{R})$ and $\mathcal{L}_{=0}(\mathcal{B}) = \mathcal{L}_{=0}(\mathcal{R})$. Furthermore, if $\mathcal{B}$ is a RatPBA then $\mathcal{R}$ can be taken to be RatPRA and the construction of $\mathcal{R}$ is recursive in this case.

Finite probabilistic monitors (FPM)s. We identify one useful syntactic restriction of PBAs, called *finite probabilistic monitors* (FPM)s. In an FPM, all the states are accepting except a special absorbing *reject* state. We studied them extensively in [4, 5].

Definition: A PBA $\mathcal{M} = (Q, q_s, Q_f, \delta)$ on Σ is said to be an FPM if there is a state $q_r \in Q$ such that $q_r \neq q_s$, $Q_f = Q \setminus \{q_r\}$ and $\delta(q_r, a, q_r) = 1$ for each $a \in \Sigma$. The state q_r said to be the *reject* state of $\mathcal{M}$. If in addition $\mathcal{M}$ is a RatPBA, we say that $\mathcal{M}$ is a rational finite probabilistic monitor (RatFPM).

3 Probable semantics

In this section, we shall study the expressiveness of the languages contained in $\mathbb{L}(\mathrm{PBA}^{>0})$ as well as the complexity of deciding emptiness and universality of $\mathcal{L}_{>0}(\mathcal{B})$ for a given RatPBA $\mathcal{B}$. We assume that the alphabet Σ is fixed and contains at least two letters.

3.1 Expressiveness

It was already shown in [3] that the class of ω-regular languages is strictly contained in the class $\mathbb{L}(\mathrm{PBA}^{>0})$ and that $\mathbb{L}(\mathrm{PBA}^{>0})$ is closed under the Boolean operations of complementation, finite intersection and finite union. We will now show that even though the class $\mathbb{L}(\mathrm{PBA}^{>0})$ strictly contains ω-regular languages, it is not topologically harder. More precisely, we will show that for any PBA $\mathcal{B}$, $\mathcal{L}_{>0}(\mathcal{B})$ is a $\mathsf{BCl}(\mathcal{G}_\delta)$-set. The proof of this fact relies on two facts. The first is that just as the class of ω-regular languages is the Boolean closure of the class of ω-regular recognized by deterministic Büchi automata, the class $\mathbb{L}(\mathrm{PBA}^{>0})$ coincides with the Boolean closure of the class $\mathbb{L}(\mathrm{PBA}^{=1})$. This is the content of the following Theorem whose proof is of independent interest and shall be used later in establishing that the containment of languages of two PBAs under almost-sure semantics is undecidable (see Theorem 4).

Theorem 1. $\mathbb{L}(\mathrm{PBA}^{>0}) = \mathsf{BCl}(\mathbb{L}(\mathrm{PBA}^{=1}))$.

Proof. First observe that it was already shown in [2] that $\mathbb{L}(\mathrm{PBA}^{=1}) \subseteq \mathbb{L}(\mathrm{PBA}^{>0})$. Since $\mathbb{L}(\mathrm{PBA}^{>0})$ is closed under Boolean operations, we get that $\mathsf{BCl}(\mathbb{L}(\mathrm{PBA}^{=1})) \subseteq \mathbb{L}(\mathrm{PBA}^{>0})$. We have to show the reverse inclusion.

It suffices to show that given a PBA $\mathcal{B}$, the language $\mathcal{L}_{>0}(\mathcal{B}) \in \mathsf{BCl}(\mathbb{L}(\mathrm{PBA}^{=1}))$. Fix $\mathcal{B}$. Recall that results of [2, 7] (see Proposition 3) imply that there is a probabilistic Rabin automaton (PRA) $\mathcal{R}$ such that 1) $\mathcal{L}_{>0}(\mathcal{B}) = \mathcal{L}_{=1}(\mathcal{R}) = \mathcal{L}_{>0}(\mathcal{R})$ and 2)

$\mathcal{L}_{=0}(\mathcal{B}) = \mathcal{L}_{=0}(\mathcal{R})$. Let $\mathcal{R} = (Q, q_s, F, \delta)$ where $F \subseteq 2^Q \times 2^Q$ is the set of the Rabin pairs. Assuming that F consists of n-pairs, let $F = ((B_1, G_1), \ldots, (B_n, G_n))$.

Given an index set $\mathcal{I} \subseteq \{1, \ldots, n\}$, let $\mathsf{Good}_{\mathcal{I}} = \cup_{r \in \mathcal{I}} G_r$. Let $\mathcal{R}_{\mathcal{I}}$ be the PBA obtained from $\mathcal{R}$ by taking the set of final states to be $\mathsf{Good}_{\mathcal{I}}$. In other words, $\mathcal{R}_{\mathcal{I}} = (Q, q_s, \mathsf{Good}_{\mathcal{I}}, \delta)$. Given $\mathcal{I} \subseteq \{1, \ldots, n\}$ and an index $j \in \mathcal{I}$, let $\mathsf{Bad}_{\mathcal{I},j} = B_j \cup \cup_{r \in \mathcal{I}, r \neq j} G_r$. Let $\mathcal{R}^j_{\mathcal{I}}$ be the PBA obtained from $\mathcal{R}$ by taking the set of final states to be $\mathsf{Bad}_{\mathcal{I},j}$, *i.e.*, $\mathcal{R}^j_{\mathcal{I}} = (Q, q_s, \mathsf{Bad}_{\mathcal{I},j}, \delta)$. The result follows from the following claim.

Claim:

$$\mathcal{L}_{>0}(\mathcal{B}) = \bigcup_{\mathcal{I} \subseteq \{1,\ldots,n\}, j \in \mathcal{I}} \mathcal{L}_{=1}(\mathcal{R}_{\mathcal{I}}) \cap (\Sigma^\omega \setminus \mathcal{L}_{=1}(\mathcal{R}^j_{\mathcal{I}})).$$

The proof of the claim is detailed in Appendix B. □

The second component needed for showing that $\mathbb{L}(\mathrm{PBA}^{>0}) \subseteq \mathsf{BCl}(\mathcal{G}_\delta)$ is the fact that for any PBA $\mathcal{B}$ and $x \in [0, 1]$, the language $\mathcal{L}_{\geq x}(\mathcal{B})$ is a $\mathcal{G}_\delta$-set; once again the proof has been deferred to Appendix B.

Lemma 1. For any PBA $\mathcal{B}$ and $x \in [0, 1]$, $\mathcal{L}_{\geq x}(\mathcal{B})$ is a $\mathcal{G}_\delta$ set.

Using Lemma 1, one immediately gets that $\mathbb{L}(\mathrm{PBA}^{>0}) \subseteq \mathsf{BCl}(\mathcal{G}_\delta)$. Even though PBAs accept non-ω-regular languages, they cannot accept all the languages in $\mathsf{BCl}(\mathcal{G}_\delta)$.

Lemma 2. $\mathsf{Regular} \subsetneq \mathbb{L}(\mathrm{PBA}^{>0}) \subsetneq \mathsf{BCl}(\mathcal{G}_\delta)$.

Remark: Please note that Lemma 1 can be used to show that the classes $\mathbb{L}(\mathrm{PBA}^{>\frac{1}{2}})$ and $\mathbb{L}(\mathrm{PBA}^{\geq\frac{1}{2}})$ are also contained within the first few levels of Borel hierarchy. However, we can show that no version of Theorem 1 holds for those classes. More precisely, $\mathbb{L}(\mathrm{PBA}^{>\frac{1}{2}}) \not\subseteq \mathbb{L}(\mathrm{PBA}^{\geq\frac{1}{2}})$ and $\mathbb{L}(\mathrm{PBA}^{\geq\frac{1}{2}}) \not\subseteq \mathbb{L}(\mathrm{PBA}^{>\frac{1}{2}})$. These results are out of the scope of the paper.

3.2 Decision problems.

Given a RatPBA $\mathcal{B}$, the problems of emptiness and universality of $\mathcal{L}_{>0}(\mathcal{B})$ are known to be undecidable [2]. We sharpen this result by showing that the problem is Σ^0_2-complete. This is interesting in the light of the fact that problems on infinite string automata that are undecidable tend to typically lie in the analytical hierarchy, and not in the arithmetic hierarchy.

Before we proceed with the proof of the upper bound, let us recall an important property of finite-state Büchi automata [18, 13]. The language recognized by a finite-state Büchi automata $\mathcal{A}$ is non-empty iff there is a final state q_f of $\mathcal{A}$, and finite words u and v such that q_f is reachable from the initial state on input u, and q_f is reachable from the state q_f on input v. This is equivalent to saying that any non-empty ω-regular language contains an ultimately periodic word. We had extended this observation to FPMs in [4, 5]. In particular, we had shown that the language $\mathcal{L}_{>x}(\mathcal{M}) \neq \emptyset$ for a given $\mathcal{M}$ iff there exists a set of final states C of $\mathcal{M}$ and words u and v such that the probability of reaching C from the initial state on input u is $> x$ and for each state $q \in \mathsf{C}$ the probability of reaching C from q on input v is 1. This immediately implies that if

$\mathcal{L}_{>x}(\mathcal{M})$ is non-empty then $\mathcal{L}_{>x}(\mathcal{M})$ must contain an ultimately periodic word. In contrast, this fact does not hold for non-empty languages in $\mathbb{L}(\text{PBA}^{>0})$. In fact, Baier and Größer [3], construct a PBA $\mathcal{B}$ such that $\mathcal{L}_{>0}(\mathcal{B})$ does not contain any ultimately periodic word.

However, we will show that even though the probable semantics may not contain an ultimately periodic, they nevertheless are restrained in the sense that they must contain *strongly asymptotic* words. Given a PBA $\mathcal{B} = (Q, q_s, Q_f, \delta)$ and a set C of states of $\mathcal{B}$, a word $\alpha \in \Sigma^\omega$ is said to be *strongly asymptotic with respect to* $\mathcal{B}$ *and* C if there is an infinite sequence $i_1 < i_2 < ....$ such that 1): $\delta_{\alpha[0:i_1]}(q_s, C) > 0$ and 2) all $j > 0$, for all $q \in C$, the probability of being in C from q after passing through a final state on the finite input string $\alpha[i_j, i_{j+1}]$ is strictly greater than $1 - \frac{1}{2^j}$. A word α is said to be *strongly asymptotic with respect to* $\mathcal{B}$ if there is some C such that α is strongly asymptotic with respect to $\mathcal{B}$ and C . The following notations shall be useful.

Notation: Let $\mathcal{B} = (Q, q_s, Q_f, \delta)$. Given $C \subseteq Q$, $q \in C$ and a finite word $u \in \Sigma^+$, let $\delta_u^{Q_f}(q, C)$ be the probability that the PBA $\mathcal{B}$, when started in state q, on the input string u, is in some state in C at the end of u after passing through a final state. predicate that for some finite non-empty input string u, the probability of being in C having started from the initial state q_s is $> x$, i.e., $Reach(\mathcal{B}, C, x) = \exists u \in \Sigma^+ . \delta_u(q_s, C) > x$.

The asymptotic sequence property is an immediate consequence of the following Lemma.

Lemma 3. Let $\mathcal{B} = (Q, q_s, Q_f, \delta)$. For any $x \in [0, 1)$, $\mathcal{L}_{>x}(\mathcal{B}) \neq \emptyset$ iff $\exists C \subseteq Q$ such that $Reach(\mathcal{B}, C, x)$ is true and $\forall j > 0$ there is a finite non-empty word u_j such that $\forall q \in C.\ \delta_{u_j}^{Q_f}(q, C) > (1 - \frac{1}{2^j})$.

Proof. The ($\Leftarrow$)-direction is proved in Appendix B. We outline here the proof of ($\Rightarrow$)-direction. The missing parts of the proof will be cast in terms of claims which will again be proved in Appendix B.

Assume that $\mathcal{L}_{>x}(\mathcal{B}) \neq \emptyset$. Fix an infinite input string $\gamma \in \mathcal{L}_{>x}(\mathcal{B})$. Recall that the probability measure generated by γ and $\mathcal{B}$ is denoted by $\mu_{\mathcal{B},\gamma}$. For the rest of this proof we will just write μ for $\mu_{\mathcal{B},\gamma}$.

We will call a non-empty set of states C *good* if there is an $\epsilon > 0$, a measurable set $\mathsf{Paths} \subseteq Q^\omega$ of runs, and an infinite sequence of natural numbers $i_1 < i_2 < i_3 < \ldots$ such that following conditions hold.

- $\mu(\mathsf{Paths}) \geq x + \epsilon$;
- For each $j > 0$ and each run ρ in Paths, we have that
  1. $\rho[0] = q_s$, $\rho[i_j] \in C$ and
  2. at least one state in the finite sequence $\rho[i_j, i_{j+1}]$ is a final state.

We say that a good set C is *minimal* if C is good but for each $q \in C$, the set $C \setminus \{q\}$ is not good. Clearly if there is a good set of states then there is also a minimal good set of states. We have the following claim which we prove in Appendix B.
Claim:

- There is a good set of states C.

– Let C be a minimal good set of states. Fix ϵ, Paths and the sequence $i_1 < i_2 < \dots$ which witness the fact that C is good set of states. For each $q \in C$ and each $j > 0$, let $\mathsf{Paths}_{j,q}$ be the subset of Paths such that each run in Paths passes through q at point i_j, *i.e.*, $\mathsf{Paths}_{j,q} = \{\rho \in \mathsf{Paths} \mid \rho[i_j] = q\}$. Then there exists a $p > 0$ such that $\mu(\mathsf{Paths}_{j,q}) \geq p$ for each $q \in C$ and each $j > 0$.

Now, fix a minimal set of good states C. Fix ϵ, Paths and the sequence $i_1 < i_2 < \dots$ which witness the fact that C is a good set of states. We claim that C is the required set of states. As $\mu(\mathsf{Paths}) \geq x + \epsilon$ and for each $\rho \in \mathsf{Paths}$, $\rho[i_1] \in C$, it follows immediately that $Reach(\mathcal{B}, C, x)$. Assume now, by way of contradiction, that there exists a $j_0 > 0$ such that for each finite word u, there exists a $q \in C$ such that $\delta_u^{Q_f}(q, C) \leq 1 - \frac{1}{2^{j_0}}$. Fix j_0. Also fix $p > 0$ be such that $\mu(\mathsf{Paths}_{j,q}) \geq p$ for each j and $q \in C$, where $\mathsf{Paths}_{j,q}$ is the subset of Paths such that each run in $\mathsf{Paths}_{j,q}$ passes through q at point i_j; the existence of p is guaranteed by the above claim.

We first construct a sequence of sets $L_i \subseteq Q^+$ as follows. Let $L_1 \subseteq Q^+$ be the set of finite words on states of Q of length $i_1 + 1$ such that each word in L_1 starts with the state q_s and ends in a state in C. Formally $L_1 = \{\eta \subseteq Q^+ \,|\, |\eta| = i_1 + 1, \eta[0] = q_s \text{ and } \eta[i_1] \in C\}$. Assume that L_r has been constructed. Let $L_{r+1} \subseteq Q^+$ be the set of finite words on states of Q of length $i_{r+1} + 1$ such that each word in L_{r+1} has a prefix in L_r, passes through a final state in between i_r and i_{r+1}, and ends in a state in C. Formally, $L_{r+1} = \{\eta \subseteq Q^+ \mid |\eta| = i_{r+1} + 1, \eta[0 : i_r] \in L_r, \exists i.(i_r < i < i_{r+1} \ \wedge \ \rho[i] \in Q_f)\}$.

Note that $L_r \Sigma^\omega$ is a decreasing sequence of measurable subsets and $\mathsf{Paths} \subseteq \cap_{r>1} L_r \Sigma^\omega$. Now, it is easy to see from the choice of j_0 and p that $\mu(L_{r+1}\Sigma^\omega) \leq \mu(L_r \Sigma^\omega) - \frac{p}{2^{j_0}}$. This, however, implies that there is a r_0 such that $\mu(L_{r_0}\Sigma^\omega) < 0$. A contradiction. □

Lemma 3 implies that checking the emptiness of $\mathcal{L}_{>0}(\mathcal{B})$ for a given a RatPBA $\mathcal{B}$ is in Π_2^0. We can exhibit that non-emptiness checking is Π_2^0-hard also. Since the class $\mathbb{L}(\mathrm{PBA}^{>0})$ is closed under complementation and the complementation procedure is recursive [2] for RatPBAs, we can conclude that checking universality of $\mathcal{L}_{>0}(\mathcal{B})$ is also Σ_2^0-complete. The same bounds also apply to checking language containment under probable semantics. Note that these problems were already shown to undecidable in [2], but the exact complexity was not computed therein.

Theorem 2. Given a RatPBA, $\mathcal{B}$, the problems 1) deciding whether $\mathcal{L}_{>0}(\mathcal{B}) = \emptyset$ and 2) deciding whether $\mathcal{L}_{>0}(\mathcal{B}) = \Sigma^\omega$, are Σ_2^0-complete. Given another RatPBA, $\mathcal{B}'$, the problem of deciding whether $\mathcal{L}_{>0}(\mathcal{B}) \subseteq \mathcal{L}_{>0}(\mathcal{B}')$ is also Σ_2^0-complete.

Remark: Lemma 3 can be used to show that emptiness-checking of $\mathcal{L}_{>\frac{1}{2}}(\mathcal{B})$ for a given RatPBA $\mathcal{B}$ is in Σ_2^0. In contrast, we had shown in [5] that the problem of deciding whether $\mathcal{L}_{>\frac{1}{2}}(\mathcal{M}) = \Sigma^\omega$ for a given FPM $\mathcal{M}$ lies beyond the arithmetical hierarchy.

4 Almost-sure semantics

The class $\mathbb{L}(\mathrm{PBA}^{=1})$ was first studied in [2], although they were not characterized topologically. In this section, we study the expressiveness and complexity of the class

$\mathbb{L}(\mathrm{PBA}^{=1})$. We will also demonstrate that the class $\mathbb{L}(\mathrm{PBA}^{=1})$ is closed under finite unions and intersections. As in the case of probable semantics, we assume that the alphabet Σ is fixed and contains at least two letters.

4.1 Expressiveness

Lemma 1 already implies that topologically, the class $\mathbb{L}(\mathrm{PBA}^{=1}) \subseteq \mathcal{G}_\delta$. Recall that $\mathcal{G}_\delta$ coincides exactly with the class of languages recognizable with infinite-state deterministic Büchi automata (see Section 2). Thanks to Theorem 1 and Lemma 2, it also follows immediately that the inclusion $\mathbb{L}(\mathrm{PBA}^{=1}) \subseteq \mathcal{G}_\delta$ is strict (otherwise we will have $\mathbb{L}(\mathrm{PBA}^{>0}) = \mathsf{BCl}(\mathbb{L}(\mathrm{PBA}^{=1})) = \mathsf{BCl}(\mathcal{G}_\delta)$). The fact that every language $\mathbb{L}(\mathrm{PBA}^{=1})$ is contained in $\mathcal{G}_\delta$ implies immediately that there are ω-regular languages not in $\mathbb{L}(\mathrm{PBA}^{=1})$. That there are ω-regular languages not in $\mathbb{L}(\mathrm{PBA}^{=1})$ was also proved in [2], although the proof therein is by explicit construction of an ω-regular language which is then shown to be not in $\mathbb{L}(\mathrm{PBA}^{=1})$. Our topological characterization of the class $\mathbb{L}(\mathrm{PBA}^{=1})$ has the advantage that we can characterize the intersection $\mathsf{Regular} \cap \mathbb{L}(\mathrm{PBA}^{=1})$ exactly: $\mathsf{Regular} \cap \mathbb{L}(\mathrm{PBA}^{=1})$ is the class of ω-regular languages that can be recognized by a finite-state deterministic Büchi automaton. The following is proved in Appendix C.

Proposition 4. For any PBA $\mathcal{B}$, $\mathcal{L}_{=1}(\mathcal{B})$ is a $\mathcal{G}_\delta$ set. Furthermore, $\mathsf{Regular} \cap \mathbb{L}(\mathrm{PBA}^{=1}) = \mathsf{Regular} \cap \mathrm{Deterministic}$ and $\mathsf{Regular} \cap \mathrm{Deterministic} \subsetneq \mathbb{L}(\mathrm{PBA}^{=1}) \subsetneq \mathcal{G}_\delta = \mathrm{Deterministic}$.

An immediate consequence of the characterization of the intersection $\mathsf{Regular} \cap \mathrm{Deterministic}$ is that the class $\mathbb{L}(\mathrm{PBA}^{=1})$ is not closed under complementation as the class of ω-regular languages recognized by deterministic Büchi automata is not closed under complementation. That the class $\mathbb{L}(\mathrm{PBA}^{=1})$ is not closed under complementation is also observed in [2], and is proved by constructing an explicit example. However, even though the class $\mathbb{L}(\mathrm{PBA}^{=1})$ is not closed under complementation, we have a "partial" complementation operation— for any PBA $\mathcal{B}$ there is another PBA $\mathcal{B}'$ such that $\mathcal{L}_{>0}(\mathcal{B}')$ is the complement of $\mathcal{L}_{=1}(\mathcal{B})$. This also follows from the results of [2] as they showed that $\mathbb{L}(\mathrm{PBA}^{=1}) \subseteq \mathbb{L}(\mathrm{PBA}^{>0})$ and $\mathbb{L}(\mathrm{PBA}^{>0})$ is closed under complementation. However our construction has two advantages: 1) it is much simpler than the one obtained by the constructions in [2], and 2) the PBA $\mathcal{B}'$ belongs to the restricted class of finite probabilistic monitors FPMs (see Section 2 for definition of FPMs). This construction plays a critical role in our complexity analysis of decision problems.

Lemma 4. For any PBA $\mathcal{B}$, there is an FPM $\mathcal{M}$ such that $\mathcal{L}_{=1}(\mathcal{B}) = \Sigma^\omega \setminus \mathcal{L}_{>0}(\mathcal{M})$.

Proof. Let $\mathcal{B} = (Q, q_s, Q_f, \delta)$. We construct $\mathcal{M}$ as follows. First we pick a new state q_r, which will be the reject state of the FPM $\mathcal{M}$. The set of states of $\mathcal{M}$ would be $Q \cup \{q_r\}$. The initial state of $\mathcal{M}$ will be q_s, the initial state of $\mathcal{B}$. The set of final states of $\mathcal{M}$ will be Q, the set of states of $\mathcal{B}$. The transition relation of $\mathcal{M}$ would be defined as follows. If q is not a final state of $\mathcal{B}$ then the transition function would be the same as for $\mathcal{B}$. If q is an final state of $\mathcal{B}$ then $\mathcal{M}$ will transit to the reject state with probability $\frac{1}{2}$ and with probability $\frac{1}{2}$ continue as in $\mathcal{B}$. Formally, $\mathcal{M} = (Q \cup \{q_r\}, q_s, Q, \delta_\mathcal{M})$ where $\delta_\mathcal{M}$ is defined as follows. For each $a \in \Sigma$, $q, q' \in Q$,

- $\delta_{\mathcal{M}}(q, a, q_r) = \frac{1}{2}$ and $\delta_{\mathcal{M}}(q, a, q') = \frac{1}{2}\delta(q, a, q')$ if $q \in Q_f$,
- $\delta_{\mathcal{M}}(q, a, q_r) = 0$ and $\delta_{\mathcal{M}}(q, a, q') = \delta(q, a, q')$ if $q \in Q \setminus Q_f$,
- $\delta_{\mathcal{M}}(q_r, a, q_r) = 1$.

It is easy to see that a word $\alpha \in \Sigma^\omega$ is rejected with probability 1 by $\mathcal{M}$ iff it is accepted with probability 1 by $\mathcal{B}$. The result now follows. □

The "partial" complementation operation has many consequences. One consequence is that the class $\mathbb{L}(\mathrm{PBA}^{=1})$ is closed under union. The class $\mathbb{L}(\mathrm{PBA}^{=1})$ is easily shown to be closed under intersection. Hence for closure properties, $\mathbb{L}(\mathrm{PBA}^{=1})$ behave like deterministic Büchi automata. The proof of these closure properties has been deferred to Appendix C. Please note that closure properties were not studied in [2].

Corollary 1. The class $\mathbb{L}(\mathrm{PBA}^{=1})$ is closed under finite union and finite intersection.

A second consequence is that unlike the case of probable semantics, almost-sure semantics of a PBA, if non-empty, is guaranteed to contain an ultimately periodic word (See Appendix C for a proof).

Corollary 2. For any PBA $\mathcal{B}$, if $\mathcal{L}_{=1}(\mathcal{B}) \neq \emptyset$ then $\mathcal{L}_{=1}(\mathcal{B})$ contains an ultimately periodic word. Furthermore, if $\mathcal{L}_{=1}(\mathcal{B}) \neq \Sigma^\omega$ then $\Sigma^\omega \setminus \mathcal{L}_{=1}(\mathcal{B})$ contains an ultimately periodic word.

4.2 Decision problems

The problem of checking whether $\mathcal{L}_{=1}(\mathcal{B}) = \emptyset$ for a given RatPBA $\mathcal{B}$ was shown to be decidable in **EXPTIME** in [2], where it was also conjectured to be **EXPTIME**-complete. The decidability of the universality problem was left open in [2]. We can leverage our "partial" complementation operation to show that a) the emptiness problem is in fact **PSPACE**-complete, thus tightening the bound in [2] and b) the universality problem is also **PSPACE**-complete.

Theorem 3. Given a RatPBA $\mathcal{B}$, the problem of deciding whether $\mathcal{L}_{=1}(\mathcal{B}) = \emptyset$ is **PSPACE**-complete. The problem of deciding whether $\mathcal{L}_{=1}(\mathcal{B}) = \Sigma^\omega$ is also **PSPACE**-complete.

Proof. (**Upper bounds**.) We first show the upper bounds. The proof of Lemma 4 shows that for any RatPBA $\mathcal{B}$, there is a RatFPM $\mathcal{M}$ constructed in polynomial time such that $\mathcal{L}_{=1}(\mathcal{B}) = \Sigma^\omega \setminus \mathcal{L}_{>0}(\mathcal{M})$. $\mathcal{L}_{=1}(\mathcal{B})$ is empty (universal) iff $\mathcal{L}_{>0}(\mathcal{M})$ is universal (empty respectively). Now, we had shown in [4, 5] that given a RatFPM $\mathcal{M}$, the problems of checking emptiness and universality of $\mathcal{L}_{>0}(\mathcal{M})$ are in **PSPACE**, thus giving us the desired upper bounds.
(**Lower bounds**.) We had shown in [4, 5] that given a RatFPM $\mathcal{M}$, the problems of deciding the emptiness and universality of $\mathcal{L}_{>0}(\mathcal{M})$ are **PSPACE**-hard respectively. Given a RatFPM $\mathcal{M} = (Q, q_s, Q_0, \delta)$ with q_r as the absorbing reject state, consider the PBA $\overline{\mathcal{M}} = (Q, q_s, \{q_r\}, \delta)$ obtained by considering the unique reject state of $\mathcal{M}$ as the only final state of $\overline{\mathcal{M}}$. Clearly we have that $\mathcal{L}_{>0}(\mathcal{M}) = \Sigma^\omega \setminus \mathcal{L}_{=1}(\overline{\mathcal{M}})$. Thus $\mathcal{L}_{>0}(\mathcal{M})$ is empty (universal) iff $\mathcal{L}_{=1}(\overline{\mathcal{M}})$ is universal (empty respectively). The result now follows. □

Even though the problems of checking emptiness and universality of almost-sure semantics of a RatPBA are decidable, the problem of deciding language containment under almost-sure semantics turns out to be undecidable, and is indeed as hard as the problem of deciding language containment under probable semantics (or, equivalently, checking emptiness under probable semantics).

Theorem 4. Given RatPBAs, $\mathcal{B}_1$ and $\mathcal{B}_2$, the problem of deciding whether $\mathcal{L}_{=1}(\mathcal{B}_1) \subseteq \mathcal{L}_{=1}(\mathcal{B}_2)$ is Σ_2^0-complete.

5 Hierarchical PBAs

We shall now identify a simple syntactic restriction on PBAs which under probable semantics coincide exactly with ω-regular languages and under almost-sure semantics coincide exactly with ω-regular deterministic languages. These restricted PBAs shall be called *hierarchical* PBAs.

Intuitively, a hierarchical PBA is a PBA such that the set of its states can be stratified into (totally) ordered levels. From a state q, for each letter a, the machine can transition with non-zero probability to at most one state in the same level as q, and all other probabilistic transitions go to states that belong to a higher level. Formally,

Definition: Given a natural number k, a PBA $\mathcal{B} = (Q, q_s, Q, \delta)$ over an alphabet Σ is said to be a *k-level hierarchical PBA (k*-PBA*)* if there is a function $\mathsf{rk} : Q \to \{0, 1, \ldots, k\}$ such that the following holds.

> Given $j \in \{0, 1, \ldots, k\}$, let $Q_j = \{q \in Q \mid \mathsf{rk}(Q) = j\}$. For every $q \in Q$ and $a \in \Sigma$, if $j_0 = \mathsf{rk}(q)$ then $\mathsf{post}(q, a) \subseteq \cup_{j_0 \leq \ell \leq k} Q_\ell$ and $|\mathsf{post}(q, a) \cap Q_{j_0}| \leq 1$.

The function rk is said to be a *compatible ranking function* of $\mathcal{B}$ and for $q \in Q$ the natural number $\mathsf{rk}(q)$ is said to be the *rank* or *level* of q. $\mathcal{B}$ is said to be a *hierarchical* PBA *(*HPBA*)* if $\mathcal{B}$ is k-hierarchical for some k. If $\mathcal{B}$ is also a RatPBA, we say that $\mathcal{B}$ is a rational hierarchical PBA (RatHPBA).

We can define classes analogous to $\mathbb{L}(\mathrm{PBA}^{>0})$ and $\mathbb{L}(\mathrm{PBA}^{=1})$; and we shall call them $\mathbb{L}(\mathrm{HPBA}^{>0})$ and $\mathbb{L}(\mathrm{HPBA}^{=1})$ respectively.

Before we proceed to discuss the probable and almost-sure semantics for HPBAs, we point out two interesting facts about hierarchical HPBAs. First is that for the class of ω-regular deterministic languages, HPBAs like non-deterministic Büchi automata can be exponentially more succinct. The proof can be found in Appendix D.

Lemma 5. Let $\Sigma = \{a, b, c\}$. For each $n \in \mathbb{N}$, there is a ω-regular deterministic property $\mathsf{L}_n \subseteq \Sigma^\omega$ such that i) any deterministic Büchi automata for L_n has at least $O(2^n)$ number of states, and ii) there are HPBAs $\mathcal{B}_n$ s.t. $\mathcal{B}_n$ has $O(n)$ number of states and $\mathsf{L}_n = \mathcal{L}_{=1}(\mathcal{B}_n)$.

The second thing is that even though HPBAs yield only ω-regular languages under both almost-sure semantics and probable semantics, we can recognize non-ω-regular languages with cut-points. The proof can be found in Appendix D.

Lemma 6. There is a HPBA $\mathcal{B}$ such that both $\mathcal{L}_{\geq \frac{1}{2}}(\mathcal{B})$ and $\mathcal{L}_{> \frac{1}{2}}(\mathcal{B})$ are not ω-regular.

Remark: We will see shortly that the problems of deciding emptiness and universality for a HPBA turn out to be decidable under both probable and almost-sure semantics. However, with cut-points, they turn out to be undecidable. The latter, however, is out of scope of this paper.

5.1 Probable semantics.

We shall now show that the class $\mathbb{L}(\mathrm{HPBA}^{>0})$ coincides with the class of ω-regular languages under probable semantics. In [3], a restricted class of PBAs called uniform PBAs was identified that also accept exactly the class of ω-regular languages. We make a couple of observations, contrasting our results here with theirs. First the definition of uniform PBA was semantic (i.e., the condition depends on the acceptance probability of infinitely many strings from different states of the automaton), whereas HPBA are a syntactic restriction on PBA. Second, we note that the definitions themselves are incomparable in some sense; in other words, there are HPBAs which are not uniform, and vice versa. Finally, HPBAs appear to be more tractable than uniform PBAs. We show that the emptiness problem for $\mathbb{L}(\mathrm{HPBA}^{>0})$ is **NL**-complete. In contrast, the same problem was demonstrated to be in **EXPTIME** and co-**NP**-hard [3].

Our main observation is the Hierarchical PBAs capture exactly the class of ω-regular languages; its proof is deferred to Appendix D.

Theorem 5. $\mathbb{L}(\mathrm{HPBA}^{>0}) = \mathsf{Regular}$.

We will show that the problem of deciding whether $\mathcal{L}_{>0}(\mathcal{B})$ is empty for hierarchical RatPBA's is **NL**-complete while the problem of deciding whether $\mathcal{L}_{>0}(\mathcal{B})$ is universal is **PSPACE**-complete. Thus "algorithmically", hierarchical PBAs are much "simpler" than both PBAs and uniform PBAs. Note that the emptiness and universality problem for finite state Büchi-automata are also **NL**-complete and **PSPACE**-complete respectively. Tthe proof can be found in Appendix D.

Theorem 6. Given a RatHPBA, $\mathcal{B}$, the problem of deciding whether $\mathcal{L}_{>0}(\mathcal{B}) = \emptyset$ is **NL**-complete. The problem of deciding whether $\mathcal{L}_{>0}(\mathcal{B}) = \Sigma^\omega$ is **PSPACE**-complete.

5.2 Almost-sure semantics.

For a hierarchical PBA, the "partial" complementation operation for almost-sure semantics discussed in Section 4 yields a hierarchical PBA. Therefore using Theorem 5, we immediately get that a language $\mathcal{L} \in \mathbb{L}(\mathrm{HPBA}^{=1})$ is ω-regular. Thanks to the topological characterization of $\mathbb{L}(\mathrm{HPBA}^{=1})$ as a sub-collection of deterministic languages, we get that $\mathbb{L}(\mathrm{HPBA}^{=1})$ is exactly the class of languages recognized by deterministic finite-state Büchi automata (the proof is moved to Appendix D).

Theorem 7. $\mathbb{L}(\mathrm{HPBA}^{=1}) = \mathsf{Regular} \cap \mathrm{Deterministic}$.

The "partial" complementation operation also yields the complexity of emptiness and universality problems (the proof is moved to Appendix D).

Theorem 8. Given a RatHPBA, $\mathcal{B}$, the problem of deciding whether $\mathcal{L}_{=1}(\mathcal{B}) = \emptyset$ is **PSPACE**-complete. The problem of deciding whether $\mathcal{L}_{=1}(\mathcal{B}) = \Sigma^\omega$ is **NL**-complete.

6 Conclusions

In this paper, we investigated the power of randomization in finite state automata on infinite strings. We presented a number of results on the expressiveness and decidability problems under different notions of acceptance based on the probability of acceptance. In the case of decidability, we gave tight bounds for both the universality and emptiness problems. As part of future work, it will be interesting to investigate the power of randomization in other models of computations on infinite strings such as pushdown automata etc. Since the universality and emptiness problems are PSPACE-complete for almost-sure semantics, their application to practical systems needs further enquiry.

6.1 Acknowledgements.

Rohit Chadha was supported in part by NSF grants CCF04-29639 and NSF CCF04-48178. A. Prasad Sistla was supported in part by NSF CCF-0742686. Mahesh Viswanathan was supported in part by NSF CCF04-48178 and NSF CCF05-09321.

References

1. B. Alpern and F. Schneider. Defining liveness. *Information Processing Letters*, 21:181–185, 1985.
2. C. Baier, N. Bertrand, and M. Größer. On decision problems for probabilistic Büchi automata. In *11th International Conference on Foundations of Software Science and Computational Structures, FoSSaCS*, pages 287–301, 2008.
3. C. Baier and M. Größer. Recognizing ω-regular languages with probabilistic automata. In *Proceedings of the IEEE Symposium on Logic in Computer Science*, pages 137–146, 2005.
4. R. Chadha, A. P. Sistla, and M. Viswanathan. On the expressiveness and complexity of randomization in finite state monitors. In *LICS 2008- 23rd Annual IEEE Symposium on Logic in Computer Science*, pages 18–29. IEEE Computer Society, 2008.
5. R. Chadha, A. P. Sistla, and M. Viswanathan. On the expressiveness and complexity of randomization in finite state monitors. *Journal of the ACM*, To appear. Available at http://www.cs.uiuc.edu/homes/rch/TechReports/MonitoringTechReport.pdf.
6. A. Condon and R. J. Lipton. On the complexity of space bounded interactive proofs (extended abstract). In *30th Annual Symposium on Foundations of Computer Science*, pages 462–467, 1989.
7. M. Groesser. *Reduction Methods for Probabilistic Model Checking*. PhD thesis, TU Dresden, 2008.
8. G. J. Holzmann and D. Peled. The state of spin. In *Proceedings of the 8th International Conference on Computer Aided Verification*, 1996.
9. J. Kemeny and J. Snell. *Denumerable Markov Chains*. Springer-Verlag, 1976.
10. R. P. Kurshan. *Computer Aided Verification of the Coordinated Processes: The Automata Theoretic Approach*. Princeton University Press, 1994.
11. L. Lamport. Logical foundation, distributed systems- methods and tools for specification. *Springer-Verlag Lecture Notes in Computer Science*, 190, 1985.
12. A. Paz. *Introduction to Probabilistic Automata*. Academic Press, 1971.
13. D. Perrin and J-E. Pin. *Infinite Words: Automata, Semigroups, Logic and Games*. Elsevier, 2004.

14. M. O. Rabin. Probabilitic automata. *Information and Control*, 6(3):230–245, 1963.
15. J. M. Rutten, M. Kwiatkowska, G. Norman, and D. Parker. *Mathematical Techniques for Analyzing Concurrent and Probabilistic Systems*. AMS, 2004.
16. A. Salomaa. *Formal Languages*. Academic Press, 1973.
17. A. P. Sistla. *Theoretical Issues in the Design and Verification of Distributed Systems*. PhD thesis, Harvard University, 1983.
18. W. Thomas. Automata on infinite objects. In *Handbook of Theoretical Computer Science*, volume B, pages 133–192. 1990.
19. M. Vardi. Automatic verification of probabilistic concurrent systems. In *26th annual Symposium on Foundations of Computer Science*, pages 327–338. IEEE Computer Society Press, 1985.
20. M. Vardi and P. Wolper. An automata theoretic approach to automatic program verification. In *Proceedings of the first IEEE Symposium on Logic in Computer Science*, 1986.
21. M. Vardi, P. Wolper, and A. P. Sistla. Reasoning about infinite computation paths. In *Proceedings of the 24th IEEE Symposium on Foundations of Computer Science*, 1983.

A Some more preliminary definitions for Section 2

Arithmetical Hierarchy. Let Δ be a finite alphabet. A language L over Δ is a set of finite strings over Δ. Arithmetical hierarchy consists of classes of languages $\Sigma^0_n,\ \Pi^0_n$ for each integer $n > 0$. Fix an $n > 0$. A language $L \in \Sigma^0_n$ iff there exists a recursive predicate $\phi(u, \boldsymbol{x}_1, ..., \boldsymbol{x}_n)$ where u is a variable ranging over Δ^*, and for each i,$0 < i \leq n$, $\boldsymbol{x}_i$ is a finite sequence of variables ranging over integers such that

$$L \ = \ \{u \in \Delta^* \mid \exists \boldsymbol{x}_1, \forall \boldsymbol{x}_2, \ldots, Q_n \boldsymbol{x}_n \ \phi(u, \boldsymbol{x}_1, ..., \boldsymbol{x}_n)\}$$

where Q_n is an existential quantifier if n is odd, else it is a universal quantifier. Note that the quantifiers in the above equation are alternating starting with an existential quantifier. The class Π^0_n is exactly the class of languages that are complements of languages in Σ^0_n. $\Sigma^0_1,\ \Pi^0_1$ are exactly the class of **R.E.**-sets and **co-R.E.**-sets. A well known language in Σ^0_1 is the set of encodings of Turing machines that halt on empty input. A well known language in Π^0_2 is the set of encodings of deterministic Turing machines that halt on infinite number of inputs.

Borel Hierarchy on the Cantor space. For a class $\mathcal{L}$ of languages, we define $\mathcal{L}_\delta = \{\cap_{i \in \mathbb{N}} \mathsf{L}_i \mid \mathsf{L}_i \in \mathcal{L}\}$ and $\mathcal{L}_\sigma = \{\cup_{i \in \mathbb{N}} \mathsf{L}_i \mid \mathsf{L}_i \in \mathcal{L}\}$. The set of open sets of the Cantor space is closed under arbitrary unions but only finite intersections. Similarly the set of closed sets of the Cantor union is closed arbitrary intersections but only finite unions. The Borel hierarchy of the Cantor space is obtained by the means of countable unions, intersections and complementation. This yields a transfinite hierarchy, but we will restrict our attention to the first few levels. At the lowest level of this hierarchy is the collection $\mathcal{G} \cap \mathcal{F}$ which is strictly contained in both $\mathcal{G}$ and $\mathcal{F}$ which form the next level of the hierarchy. Both $\mathcal{G}$ and $\mathcal{F}$ are strictly contained in the collection $\mathcal{G}_\delta \cap \mathcal{F}_\sigma$ which forms the next level. The collection $\mathcal{G}_\delta \cap \mathcal{F}_\sigma$ is strictly contained in $\mathcal{G}_\delta$ and $\mathcal{F}_\sigma$ which is at the next level. $\mathcal{G}_\delta$ and $\mathcal{F}_\sigma$ are strictly contained in $\mathcal{G}_{\delta\sigma} \cap \mathcal{F}_{\sigma\delta}$ which itself is strictly contained in both $\mathcal{G}_{\delta\sigma}$ and $\mathcal{F}_{\sigma\delta}$.

Construction of measure space. Given a PBA, $\mathcal{B} = (Q, q_s, Q_f, \delta)$ on the alphabet Σ and a word $\alpha \in \Sigma^\omega$, the *probability space generated* by $\mathcal{B}$ and α is the probability space $(Q^\omega, \mathcal{F}_{\mathcal{B},\alpha}, \mu_{\mathcal{B},\alpha})$ where

– $\mathcal{F}_{\mathcal{B},\alpha}$ is the smallest σ-algebra on Q^ω generated by the collection $\{\mathsf{C}_\eta \mid \eta \in Q^+\}$ where $\mathsf{C}_\eta = \{\rho \in Q^\omega \mid \eta \text{ is a prefix of } \rho\}$.
– $\mu_{\mathcal{B},\alpha}$ is the unique probability measure on $(Q^\omega, \mathcal{F}_{\mathcal{B},\alpha})$ such that $\mu_{\mathcal{B},\alpha}(\mathsf{C}_{q_0\ldots q_n})$ is
  - 0 if $q_0 \neq q_s$,
  - 1 if $n = 0$ and $q_0 = q_s$, and
  - $\delta(q_0, \alpha(1), q_1) \ldots \delta(q_{n-1}, \alpha(n), q_n)$ otherwise.

B Proofs from Section 3

B.1 Proof of Theorem 1

We just need to prove the claim in the proof. Given $\mathcal{I} \subseteq \{1, \ldots, n\}, j \in \mathcal{I}$, let $\mathsf{L}_{\mathcal{I},j} = \mathcal{L}_{=1}(\mathcal{R}_\mathcal{I}) \cap (\Sigma^\omega \setminus \mathcal{L}_{=1}(\mathcal{R}^j_\mathcal{I}))$. We will say that a run ρ of PRA $\mathcal{R}$ satisfies the Rabin pair (B_r, G_r) if all states in B_r occur only finitely many times in ρ and at least one state in G_r occurs infinitely often in ρ.

We claim first that $\mathsf{L}_{\mathcal{I},j} \subseteq \mathcal{L}_{>0}(\mathcal{R})$. Fix any $\alpha \in \mathsf{L}_{\mathcal{I},j}$. Since $\mathsf{L}_{\mathcal{I},j} \subseteq \mathcal{L}_{=1}(\mathcal{R}_\mathcal{I})$, it follows that on input α the measure of runs that visit the set $\mathsf{Good}_\mathcal{I} = \cup_{i\in\mathcal{I}}(G_i)$ infinitely often must be 1. On the other hand, as $\mathsf{L}_{\mathcal{I},j} \cap \mathcal{L}_{=1}(\mathcal{R}^j_\mathcal{I}) = \emptyset$, it follows that on input α the measure of runs that visit $\mathsf{Bad}_{\mathcal{I},j} = B_j \cup (\cup_{i\in\mathcal{I},i\neq j}(G_i))$ only finitely many times has strictly positive measure. Since $\mathsf{Good}_\mathcal{I} \setminus \mathsf{Bad}_{\mathcal{I},j} \subseteq G_j$, it now follows from the previous two observations that the measure of runs that visit G_j infinitely often but visit $\mathsf{Bad}_{\mathcal{I},j}$ only finitely many times is strictly positive. Since $B_j \subseteq \mathsf{Bad}_{\mathcal{I},j}$, we get that the set of runs that satisfy the Rabin pair (B_j, G_j) has non-zero measure on input α . Therefore, we have that $\mathsf{L}_{\mathcal{I},j} \subseteq \mathcal{L}_{>0}(\mathcal{R})$. But, we have that $\mathcal{L}_{>0}(\mathcal{R}) = \mathcal{L}_{=1}(\mathcal{R}) = \mathcal{L}_{>0}(\mathcal{B})$. Hence, we get

$$\bigcup_{\mathcal{I}\subseteq\{1,\ldots,n\},j\in\mathcal{I}} \mathsf{L}_{I,j} \subseteq \mathcal{L}_{>0}(\mathcal{B}).$$

We will be done if we can show the reverse inclusion. Thus, given word α in $\mathcal{L}_{>0}(\mathcal{B})$, we have to construct $\mathcal{I}$ and j such that $\alpha \in \mathsf{L}_{I,j}$. We construct them as follows.

First, let $\widetilde{\mathcal{I}}$ be the set of all indices r such that the measure of all runs that satisfy the Rabin pair (B_r, G_r) on input α is > 0. $\widetilde{\mathcal{I}}$ is non-empty (since $\alpha \in \mathcal{L}_{=1}(\mathcal{R})$). Clearly, we have that on input α, the measure of runs such that $\mathsf{Good}_{\widetilde{\mathcal{I}}}$ is visted infintely often is 1 (again, since $\alpha \in \mathcal{L}_{=1}(\mathcal{R})$). In other words, $\alpha \in \mathcal{L}_{=1}(\mathcal{R}_{\widetilde{\mathcal{I}}})$. Required $\mathcal{I}$ will be a subset of $\widetilde{\mathcal{I}}$ and will be constructed by induction as follows.

At step 1 of the induction, we pick an arbitrary index r in $\widetilde{\mathcal{I}}$. Then we check if it is the case that on α, the probability of visting G_r infinitely often in $\mathcal{R}$ is 1. Note that it is the case that the probability that B_r is visited infinitely often in $\mathcal{R}$ is < 1 (as α satisfies (B_r, G_r) with non-zero probability). Note that this implies that $\alpha \in \mathcal{L}_{=1}(\mathcal{R}_{\{r\}}) \cap (\Sigma^\omega \setminus \mathcal{L}_{=1}(\mathcal{R}_{\{r\},r})$ and the induction stops at this point. If it is not the case, then let $\mathcal{I}_1 = \{r\}$.

Proceed by induction. At step m, we would have produced an index set $\mathcal{I}_m \subseteq \widetilde{\mathcal{I}}$ such that on α, we have that $\alpha \notin \mathcal{L}_{=1}(\mathcal{R}_{\mathcal{I}_m})$ (meaning the set of runs which visit $\mathsf{Good}_{\mathcal{I}_m}$ infinitely often have probability < 1). Now since α is accepted by PRA $\mathcal{R}$

with probability 1, there must be some index r in $\widetilde{\mathcal{I}} \setminus \mathcal{I}_m$ such that the set of runs that satisfy (B_r, G_r) and visit Good_{I_m} only finitely many times is > 0. Fix one such r. Now, there are two cases.

1. On the input α, the set of runs that visit $\mathsf{Good}_{\mathcal{I}_m} \cup G_r$ infinitely often has measure 1. In that case, by construction, we also have that $\alpha \in \mathsf{L}_{\mathcal{I}_m \cup \{r\}, r}$ and induction stops.
2. Otherwise, we let $\mathcal{I}_{m+1} = \mathcal{I}_m \cup \{r\}$ and proceed.

The induction must stop at a finite point at which we will satisfy the required condition (since $\alpha \in \mathcal{L}_{=1}(\mathcal{R}_{\widetilde{\mathcal{I}}})$).

B.2 Proof of Lemma 1.

Let $\mathcal{B} = (Q, q_s, Q_f, \delta)$. Now given $k > 0$, let $\mathsf{Paths}^k \subseteq \Sigma^\omega$ be the set of all infinite runs which start at the state q_s and visit the set of final states at least k-times. Let Paths^ω be the set of all infinite runs which start at the state q_s and visit the final states infinitely often. Formally, $\mathsf{Paths}^k = \{\rho \in \Sigma^\omega \mid \rho(0) = q_s \text{ and } |\{i \in \mathbb{N} \mid \rho(i) \in Q_f\}| \geq k\}$ and $\mathsf{Paths}^\omega = \{\rho \in \Sigma^\omega \mid \rho(0) = q_s \text{ and } |\{i \in \mathbb{N} \mid \rho(i) \in Q_f\}| = \omega\}$. We have that $Path^k, k > 0$ form a decreasing sequence and

$$\cap_{k \in \mathbb{N}, k>0} \mathsf{Paths}^k = \mathsf{Paths}^\omega.$$

From standard probability theory, we get that for any word α,

$$\lim_{j \to \infty} \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) = \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^\omega)$$

where $\mu_{\mathcal{B},\alpha}$ is the probability measure generated by the infinite word α and PBA $\mathcal{B}$. From this, we immediately see that an infinite word α is accepted with probability at least x iff for all $k > 0$ the probability of visiting the set of final states on input α at least k-times $\geq x$. In other words,

$$\{\alpha \in \Sigma^\omega \mid \mu^{acc}_{\mathcal{B},\,\alpha} \geq x\} = \cap_{k \in \mathbb{N}, k>0} \{\alpha \in \Sigma^\omega \mid \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) \geq x\}.$$

Hence, it suffices to show that for each $k \in \mathbb{N}, k > 0$ the set $\{\alpha \in \Sigma^\omega \mid \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) \geq x\}$ is a $\mathcal{G}_\delta$ set. Note that for each $k > 0$,

$$\{\alpha \in \Sigma^\omega \mid \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) \geq x\} = \cap_{n \in \mathbb{N}} \{\alpha \in \Sigma^\omega \mid \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) > x - \frac{1}{n}\}.$$

Hence, it suffices to show that for each $k \in \mathbb{N}, n \in \mathbb{N}, k > 0$ the set $\{\alpha \in \Sigma^\omega \mid \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) > x - \frac{1}{n}\}$ is $\mathcal{G}$-set. In order to see this, consider the PBA $\mathcal{B}^k = \{Q^k, q^k_s, Q^k_f, \delta^k\}$ constructed as follows. The set of states, Q^k, of $\mathcal{B}^k$ are going to be $Q \times [0, 1, \ldots, k-1] \cup \{q_a\}$, where q_a is a new state. Intuitively, the state (q, l) will encode that "the set Q_f has been visited l times" and q_a will encode that "the set Q_f has been visited at least k times." The initial state q^k_s will be $(q_s, 0)$. The set of final states Q^k_f will be $\{q_a\}$. The transition function δ^k is defined as follows. For any state of the form (q, l) where $q \notin Q_f$, transition will be like that in $\mathcal{B}$ and the machine will stay in the "level l" and from any state of the form (q, l) where $q \in Q_f$, transition will be again like in $\mathcal{B}$ except that l will be incremented (with the understanding that incrementing $k-1$ leads to state q_a). The state q_a will be absorbing. Formally, for each $a \in \Sigma, q \in Q, q' \in Q, l \in \{0, 1, \ldots k-1\}$—

- if $q \notin Q_f$ then $\delta^k((q,l), a, (q', l)) = \delta(q, q')$, $\delta^k((q,l), a, (q', l')) = 0$ if $l' \neq l$ and $\delta^k((q,l), a, q_a) = 0$;
- if $q \in Q_f$ and $l < k-1$ then $\delta^k((q,l), a, (q', l+1)) = \delta(q, q')$, $\delta^k((q,l), a, (q', l')) = 0$ if $l' \neq l+1$ and $\delta^k((q,l), a, q_a) = 0$;
- if $q \in Q_f$ and $l = k-1$ then $\delta^k((q,l), a, q_a) = 1$ and $\delta^k((q,l), a, (q', l)) = 0$; and
- $\delta^k(q_a, a, q_a) = 1$ and $\delta^k(q_a, a, (q, l)) = 0$.

It is easy to see that for any word α, probability of visiting the set of final states at least k times is the same as the probability of accepting of $\mathcal{B}^k$ accepting word α. In other words,

$$\mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) = \mu^{acc}_{\mathcal{B}^k,\,\alpha}.$$

Now, it is also easy to see that for any word α and $m \in \mathbb{N}$, the probability of being in q_a on input $\alpha[0, m]$ is an increasing function of m and that

$$\lim_{m\to\infty} \delta^k_{\alpha[0,m]}((q_s, 0), q_a) = \mu^{acc}_{\mathcal{B}^k,\,\alpha}.$$

Therefore given k and given n, $\mu^{acc}_{\mathcal{B}^k,\,\alpha} > x - \frac{1}{n}$ iff there is a finite prefix $\alpha[0, m]$ of α such that $\delta^k((q_s, 0), q_a) > x - \frac{1}{n}$. Therefore, it follows,

$$\{\alpha \in \Sigma^\omega \mid \mu_{\mathcal{B},\alpha}(\mathsf{Paths}^k) > x - \frac{1}{n}\} = L\Sigma^\omega$$

where $L \subseteq \Sigma^+$ is set of finite words $\{u \in \Sigma^+ \mid \delta^k_u(q_s, q_a) > x - \frac{1}{n}\}$. Since $L\Sigma^\omega$ is a $\mathcal{G}$ set, the result now follows.

B.3 Proof of Lemma 2

Thanks to Lemma 1, $\mathbb{L}(\mathrm{PBA}^{=1}) \subseteq \mathcal{G}_\delta$. Since $\mathbb{L}(\mathrm{PBA}^{>0}) = \mathsf{BCl}(\mathbb{L}(\mathrm{PBA}^{=1}))$ (see Theorem 1), we get that $\mathbb{L}(\mathrm{PBA}^{>0}) \subseteq \mathsf{BCl}(\mathcal{G}_\delta)$. We only have to show that the containment is strict. The proof of this fact utilizes the following result which shows that for any $\mathsf{L} \in \mathbb{L}(\mathrm{PBA}^{>0})$, the smallest safety language containing L is guaranteed to be ω-regular even if L is not.

Lemma 7. For any PBA $\mathcal{B}$, let $\mathsf{cl}(\mathsf{L})$ be the smallest safety language containing $\mathcal{L}_{>0}(\mathcal{B})$. Then $\mathsf{cl}(\mathsf{L})$ is ω-regular.

Proof. Without loss of generality, we can assume that $\mathsf{L} \neq \emptyset$. Let $\mathcal{B} = (Q, q_s, Q_f, \delta)$. Given $q \in Q$, let $\mathcal{B}_q$ be the PBA which is exactly like $\mathcal{B}$, except that the initial state is q. That is $\mathcal{B}_q = (Q, q, Q_f, \delta)$. Let $Q_{>0} \subseteq Q$ be the set of states $\{q \mid \exists \alpha. \mu^{acc}_{\mathcal{B}_q,\,\alpha} > 0\}$. Consider the finite state Büchi automata $\mathcal{A} = (Q_{>0}, q_s, Q_{>0}, \Delta)$ where $(q_1, a, q_2) \in \Delta$ iff $\delta(q_1, a, q_2) > 0$. It is easy to see that $\mathsf{cl}(\mathsf{L})$ is exactly the language recognized by $\mathcal{A}$. □

We proceed as follows. Fix two letters a, b of the alphabet Σ and consider the language L consisting of exactly one word $\alpha = abaabb \ldots a^i b^i a^{i+1} b^{i+1} \ldots$. Now, $\mathsf{cl}(\mathsf{L}) = \mathsf{L}$ (every single element set in a metric space is a safety language) and L in not ω-regular (since L does not contain any periodic word). Therefore, the closed set L is not in the class $\mathbb{L}(\mathrm{PBA}^{>0})$ (note that $\mathcal{F} \subseteq \mathcal{G}_\delta$).

B.4 Proof of Lemma 3.

We need to show the ($\Leftarrow$)-direction of the lemma and the two claims in the ($\Rightarrow$)-direction.

We start by showing the ($\Leftarrow$)-direction. Note that it is well-known fact that the product $\prod_{j=1}^{\infty}(1 - \frac{1}{2^j})$ converges and is > 0. Assume now that $\exists C \subseteq Q$ such that $Reach(\mathcal{B}, C, x)$ is true and $\forall j > 0$ there is a finite word u_j such that $\forall q \in C\ .\delta_{u_j}^{Q_f}(q, C) > (1 - \frac{1}{2^j})$. Since $Reach(\mathcal{B}, C, x)$ is true, there is a finite word u such that $\delta_u(q_s, C) > x$. Fix u. Also for each $j > 0$, fix u_j such that $\forall q \in C\ .\delta_{u_j}^{Q_f}(q, C) > (1 - \frac{1}{2^j})$. There are two cases–

1. The first case is when $x = 0$. In this case, it is easy to see that the infinite word $uu_1u_2 \ldots$ is accepted by $\mathcal{B}$ with probability > 0.
2. The second case is when $x > 0$. In this case let $z = \delta_u(q_s, C)$. Let $y = \frac{x}{z}$. We have that $y < 1$. Since the product $\prod_{j=1}^{\infty}(1 - \frac{1}{2^j}) > 0$, and $y < 1$ there is a $j_0 > 0$ such that $\prod_{j=j_0}^{\infty}(1 - \frac{1}{2^j}) > y$. Now it is easy to see that the word $\alpha = uu_{j_0}u_{j_0+1} \ldots$ is accepted by $\mathcal{B}$ with probability $> zy$. But zy is x and the result follows.

Now, we prove the claim in the ($\Rightarrow$)-direction.
Proof of Claim. We have to show two things.

1. For each $k > 0$, let $C_k = \mathsf{post}(q_s, \gamma[0 : k])$. Since the set of states Q is finite, there must be some C such that $C_k = C$ for infinitely many k's. Fix one such C. We claim that C is a good set of states. We need to show that C satisfies the definition of good set of states. So we need to construct ϵ, Paths and the infinite sequence $i_1 < i_2 < \ldots$ as in the definition of good set of states. We will pick $\epsilon > 0$ such that $\mu_{\mathcal{B},\gamma}^{acc} = x + 2\epsilon$. We construct Paths and the sequence $i_1 < i_2 < \ldots$ as follows.
First let Paths_0 be the set of all runs starting in q_s and visiting the final states infinitely often. We have that Paths_0 is measurable and $\mu(\mathsf{Paths}_0) = x + 2\epsilon$. Now, for each $j > 0$, let $\mathsf{Paths}_0^j \subseteq \mathsf{Paths}_0$ be the set of runs that visit the set of final states at least one time before j. Formally, $\mathsf{Paths}_0^j = \{\rho \in \mathsf{Paths}_0 \mid \exists i.(i < j \wedge \rho[i] \in Q_f)\}$. Clearly Paths_0^j is an increasing sequence of measurable sets and $\cup_{j \in \mathbb{N}} \mathsf{Paths}_0^j = \mathsf{Paths}_0$ (each run in Paths_0 visits the set of final states infinitely often). Since $\mu(\mathsf{Paths}_0) = x + 2\epsilon$, there must exist a j_0 such that $\mu(\mathsf{Paths}_0^{j_0}) > x + \epsilon + \frac{\epsilon}{2}$. Fix j_0 and let $i_1 > j_0$ be the smallest integer such that $\mathsf{post}(q_s, \gamma[0 : i_1]) = C$. Let $\mathsf{Paths}_1 = \mathsf{Paths}_0^{j_0}$. Clearly Paths_1 is measurable, $\mu(\mathsf{Paths}_1) > x + \epsilon + \frac{\epsilon}{2}$, and for each $\rho \in \mathsf{Paths}_1$, there is some $i < i_1$ such that $\rho(i) \in Q_f$.
Now, in a similar fashion we can construct a $i_2 > i_1$ and a measurable set $\mathsf{Paths}_2 \subseteq \mathsf{Paths}_1$ such that a) $\rho[i_2] \in C$, b) for each $\rho \in \mathsf{Paths}_2$ there is some i such that $i_1 < i < i_2$ and $\rho[i] \in Q_f$, and c) $\mu(\mathsf{Paths}_2) > x + \epsilon + \frac{\epsilon}{4}$.

We can continue this process and construct a sequence $i_1 < i_2 < \ldots$ and a sequence of measurable sets $\mathsf{Paths}_1 \subseteq \mathsf{Paths}_2 \subseteq \ldots$ such that a) $\rho(i_j) \in C$, b) for each $j > 1$ and $\rho \in \mathsf{Paths}_j$ there is some i such that $i_{j-1} < i < i_j$ and $\rho[i] \in Q_f$, and c) for each j, $\mu(\mathsf{Paths}_j) > x + \epsilon + \frac{\epsilon}{2^j}$.

Let $\mathsf{Paths} = \cap_{j\in\mathbb{N}}\mathsf{Paths}_j$. Now, it is easy to see that Paths and the sequence $i_1 < i_2 < \ldots$ are the desired set of runs and the desired sequence respectively.

2. We have that C is minimal good set of states. Note that as C is finite, we only need to show that for each $q \in Q$, $\inf_{j>0} \mu(\mathsf{Paths}_{j,q}) > 0$. We proceed by contradiction. Assume that there is some q such that $\inf_{j>0} \mu(\mathsf{Paths}_{j,q}) = 0$. Fix one such q. We will obtain a contradiction to minimality if we can show that $C \setminus \{q\}$ is also a good set of states.

   In order to show that $C \setminus \{q\}$, we have to satisfy the definition of a good set of states.

   Now, since $\inf_{j>0} \mu(\mathsf{Paths}_{j,q}) = 0$, there is some j_1 such that $\mu(\mathsf{Paths}_{j_1,q}) < \frac{\epsilon}{4}$. Let $\mathsf{Paths}^1 = \mathsf{Paths} \setminus \mathsf{Paths}_{j_1,q}$. We have that $\mathsf{Paths}^1 \subseteq \mathsf{Paths}$, $\mu(\mathsf{Paths}^1) \geq x + \frac{\epsilon}{2} + \frac{\epsilon}{4}$ and for each $\rho \in \mathsf{Paths}^1$, $\rho(i_{j_1}) \in C \setminus \{q\}$.

   Now, again as $\inf_{j>0} \mu(\mathsf{Paths}_{j,q}) = 0$, there is some $j_2 > j_1$ such that $\mu(\mathsf{Paths}_{j_2,q}) < \frac{\epsilon}{8}$. Let $\mathsf{Paths}^2 = \mathsf{Paths}^1 \setminus \mathsf{Paths}_{j_2,q}$. We have that $\mathsf{Paths}^2 \subseteq \mathsf{Paths}^1$, $\mu(\mathsf{Paths}^2) \geq x + \frac{\epsilon}{2} + \frac{\epsilon}{8}$ and for each $\rho \in \mathsf{Paths}^2$, $\rho(i_{j_2}) \in C \setminus \{q\}$. Note also that as $j_2 > j_1$ and $\mathsf{Paths}^1 \subseteq \mathsf{Paths}$, we have that for each each $\rho \in \mathsf{Paths}^2$ there is some i such that $i_1 < i < i_2$ and $\rho(i) \in Q_f$.

   We can continue and obtain a sequence $\mathsf{Paths}^{j_1} \supseteq \mathsf{Paths}^{j_2} \supseteq \ldots$ of measurable sets, and sequence $i_{j_1} < i_{j_2} < \ldots$ such that for each $l > 0$, $\mu(\mathsf{Paths}^{j_l}) \geq x + \frac{\epsilon}{2} + \frac{\epsilon}{2^l}$ and for each $\rho \in \mathsf{Paths}^{j_l}$, $\rho(i_{j_l}) \in C \setminus \{q\}$. Furthermore for each $l > 1$ and each $\rho \in \mathsf{Paths}^l$ there is some i such that $i_{l-1} < i < i_l$ and $\rho(i) \in Q_f$.

   Let $\mathsf{Paths}' = \cap_{l>0}\mathsf{Paths}^l$. We have that $\mu(\mathsf{Paths}') \geq x + \frac{\epsilon}{2}$. Clearly $\frac{\epsilon}{2}$, Paths' and the sequence $i_{j_1} < i_{j_2} < \ldots$ witness the fact that $C \setminus \{q\}$ is a good set of states.

B.5 Proof of Theorem 2

First observe that since $\mathbb{L}(\mathrm{PBA}^{>0})$ is closed under complementation and that the complementation is recursive [2, 7], the universality problem will be Σ_2^0-complete iff emptiness problem is. Also observe that given $\mathcal{B}_1$ and $\mathcal{B}_2$, we have that $\mathcal{L}_{>0}(\mathcal{B}_1) \subseteq \mathcal{L}_{>0}(\mathcal{B}_2)$ iff $\mathcal{L}_{>0}(\mathcal{B}_1) \cap (\Sigma^\omega \setminus \mathcal{L}_{>0}(\mathcal{B}_2)) = \emptyset$. Now, results of [2, 7] show that there is a constructible $\mathcal{B}_3$ such that $\mathcal{L}_{>0}(\mathcal{B}_3) = \mathcal{L}_{>0}(\mathcal{B}_1) \cap (\Sigma^\omega \setminus \mathcal{L}_{>0}(\mathcal{B}_2))$. Therefore, language containment will be in Σ_2^0 if emptiness is. Note that the universality problem also ensures that langauage containment is Σ_2^0-hard if universality is Σ_2^0-hard. Thus, we only need to consider the emptiness problem.

(Upper bound.) We show first the upper bound.

Let us fix a PBA $\mathcal{B} = (Q, q_s, Q_f, \delta)$, and $x \in [0, 1)$. Now Lemma 3 says that the non-emptiness of $\mathcal{L}_{>x}(\mathcal{B})$ is equivalent to the following property

$$\varphi = \exists C \subseteq Q.\ \exists u \in \Sigma^*.\ ((\delta_u(q_s, C) > x) \wedge \\ (\forall j.\ \exists u_j \in \Sigma^*.\ (\forall q \in C.\ \delta_{u_j}^{Q_f}(q, C) > (1 - \tfrac{1}{2^j})))$$

which can be rewritten as (by moving quantifiers out)

$$\varphi = \exists C \subseteq Q.\ \forall j.\ \exists u \in \Sigma^*.\ \exists u_j \in \Sigma^*.\ ((\delta_u(q_s, C) > x) \wedge \\ (\forall q \in C.\ \delta_{u_j}^{Q_f}(q, C) > (1 - \tfrac{1}{2^j})))$$

Now consider the property ψ given as

$$\psi = \forall j.\ \exists C_j \subseteq Q.\ \exists u \in \Sigma^*.\ \exists u_j \in \Sigma^*.\ ((\delta_u(q_s, C_j) > x) \wedge \\ (\forall q \in C_j.\ \delta_{u_j}^{Q_f}(q, C_j) > (1 - \tfrac{1}{2^j})))$$

Clearly, ψ logically follows from φ. However, in our specific case, it turns out that in fact, ψ is equivalent to φ due to the following observations. First note, that since there are only finitely many subsets of Q, there must be a $C \subseteq Q$ such that $C = C_j$ for infinitely many j (if ψ holds). Further observe that if $\exists u_j.(\forall q \in C.\ \delta_{u_j}^{Q_f}(q, C) > (1 - \frac{1}{2^j}))$ for some j then $\exists u_i.(\forall q \in C.\ \delta_{u_i}^{Q_f}(q, C) > (1 - \frac{1}{2^i}))$ holds for all $i \leq j$. From these it follows that φ logically follows from ψ.

Observe that $(\delta_u(q_s, C) > x)$ and $(\forall q \in C.\ \delta_{u_j}^{Q_f}(q, C) > (1 - \frac{1}{2^j}))$ are recursive predicates. Thus, ψ demonstrates that the non-emptiness problem is in Π_2^0, which means that emptiness is in Σ_2^0.

(Lower bound.) Now, we show the lower bound. For this it suffices to show that non-emptiness checking is Π_2^0-hard.

For any PBA B, let $L(B)$ denote the set of infinite strings accepted by B with non-zero probability. Here is an outline of the proof.

Theorem: The set $\mathcal{C}_0 = \{< M > | M$ is a PBA that accepts at least one infinite string with probability greater than $0\}$ is Π_2^0-hard.

Proof: The idea is similar to the proof techniques used in [2], where they showed this set to be r.e.-hard. By modifying it substantially we can show the $\mathcal{C}_0$ to be Π_2^0-hard.

Consider a deterministic two counter machine M which has two counters and which also reads input from the input tape which is one way and read only. We can capture the computation of M, as a sequence of configurations where each configuration is a 4-tuple (q, x, a^i, b^j, m) where q is the state of the finite state control that M changed to, x is the input symbol that is read and i, j are the new counter values and m indicates whether the input head stayed in the same place, or moved right and read a new input symbol. Here $m \in \{same, right\}$. Note that the two counter values are represented in unary form having a string of as and bs, respectively. A halting computation is a sequence of configurations ending in a halting state. The following set $\mathcal{D}$ is Π_2^0-hard: $\mathcal{D} = \{< M > | M$ is a deterministic counter machine that halts on infinite number of inputs.$\}$. We denote computations as strings over the alphabet Σ' which includes exactly the states of M, the input symbols of M, the symbols $a, b, same, right, (,)$.

We reduce $\mathcal{D}$ to $\mathcal{C}_0$. As indicated earlier, we will be using the construction [2] with some modification. In their proof, they assume that they are given a PFA $\mathcal{R}$ that has the following property: for some ϵ, $0 \leq \epsilon < \frac{1}{2}$, $\mathcal{R}$ accepts at least one input with probability greater than or equal to $1 - \epsilon$ or it accepts all inputs with probability less than or equal to ϵ. From $\mathcal{R}$, they construct two PBAs $\mathcal{P}_1$ and $\mathcal{P}_2$ such that $L(\mathcal{P}_1) \cap L(\mathcal{P}_2)$ is empty iff $\mathcal{R}$ does not accept any string with probability $\geq 1 - \epsilon$. We use their construction as follows.

Let ϵ be a constant such that $0 \leq \epsilon < \frac{1}{2}$. Consider a counter machine M. We make the assumption that on every input M reads all the input symbols. Thus the number of steps in the computation is at least as much as the length of the input. Given a counter machine M, using the construction of [6], we obtain a PFA $\mathcal{R}$ as follows. The input alphabet Σ of $\mathcal{R}$ includes Σ' together with the new symbol @ which is used to separate computations. Throughout, we use @ to separate successive computations of M in an input string of $\mathcal{R}$. The PFA $\mathcal{R}$ satisfies the following properties.

1. There exists an integer constant $d \geq 2$ such that if w is a valid and halting computations of M of length n, then the input string $(w@)^{d^n}$ is accepted by $\mathcal{R}$ with probability $\geq (1-\epsilon)$; that is, the string obtained by concatenating w, d^n number of times, where successive concatenations are separated by @, is accepted with the above probability bound.
2. If w is a concatenation of one or more computations, none of which is a valid and halting computation of M, then w is accepted by $\mathcal{R}$ with probability $\leq \epsilon$.

For any input string w, let $Pr(w)$ denote the probability of acceptance of w by $\mathcal{R}$.

Using $\mathcal{R}$, as in the construction of [2], we obtain two PBAs, $\mathcal{P}_1, \mathcal{P}_2$. Let Σ be the input alphabet of $\mathcal{R}$. The input symbols to $\mathcal{P}_1$ and $\mathcal{P}_2$ include all the input symbols to $\mathcal{R}$ together with two additional symbols $\#, \$$. Both $\mathcal{P}_1$ and $\mathcal{P}_2$ take input strings of the form $w_1^1\#w_2^1\#...w_{k_1}^1\$\$w_1^2\#w_2^2...w_{k_2}^2\$\$...$ where w_i^j is an input string of $\mathcal{R}$.

The language $L(\mathcal{P}_1)$ is given below:
$L(\mathcal{P}_1) = \{w_1^1\#w_2^1\#...w_{k_1}^1\$\$w_1^2\#w_2^2...w_{k_2}^2\$\$... \mid w_i^j \in \Sigma^*$ and $\prod_{j\geq 1}(1-(\prod_{i=1}^{k_j-1}(1-Pr(w_i^j)))) > 0\}$.

The language $L(\mathcal{P}_2)$ is given as follows:
$L(\mathcal{P}_2) = \{v_1\$\$v_2\$\$... \mid v_i \in (\Sigma \cup \{\#\})^*$ and $\prod_{i\geq 1}(1-(1-\epsilon)^{g(v_i)}) = 0\}$ where $g(v_i)$ is the number of $\#$ symbols in v_i.

We use the two PBAs $\mathcal{P}_1$ and $\mathcal{P}_2$ as specified above. We also use a third PBA $\mathcal{P}_3$ which take same input strings as $\mathcal{P}_1, \mathcal{P}_2$. The purpose of $\mathcal{P}_3$ is to make sure that in any infinite input u, the computations of M that are present in u are on infinite number of different inputs. It makes sure of this by requiring that the lengths of the computations, in the input string, grow unboundedly. $\mathcal{P}_3$ looks like $\mathcal{P}_\lambda$ as given [2]. It has three states s_0, s_1, s_r where s_r is the reject state and s_0 is the initial state. s_r is an absorbing state. In state s_0, if it gets any symbol other than $@, \#, \$\$$ then it goes to both s_0 and s_1 each with probability λ and $1-\lambda$, respectively; if it gets $@, \#, \$\$$ then it goes to s_r with probability 1; on all other input symbols it stays in s_0 with probability 1. In state s_1 it behaves as follows. On input $@, \#, \$\$$ it goes to state s_0 with probability 1; on all other inputs, it remains in s_1 with probability 1. It can be shown that all input strings u, in which the maximum length of a computation in u is bounded, are rejected by $\mathcal{P}_3$ with probability 1. (Note that a computation is any sub-string between consecutive symbols x, y such that $x, y \in \{@, \#, \$\$\}$).

For the counter machine M, let $L(M)$ be the set of input strings on which M halts. Now we have the following claim.

Claim: $L(M)$ is a finite set iff $L(\mathcal{P}_1) \cap L(\mathcal{P}_2) \cap L(\mathcal{P}_3)$ is the empty set.

Proof of the claim: Let $L'(M)$ be the set of valid halting computations of M. From our assumption about M, we see that if M halts on an input string u then the length of it's computation on u is at least as long as the length of u. From this and the fact that M is deterministic, it is easy to see that $L(M)$ is finite iff $L'(M)$ is finite. Assume $L(M)$ is a finite set. Now we show that $L(\mathcal{P}_1) \cap L(\mathcal{P}_2) \cap L(\mathcal{P}_3)$ is the empty set. Consider any $u \in L(\mathcal{P}_2) \cap L(\mathcal{P}_3)$. Since $u \in L(\mathcal{P}_3)$ and $L'(M)$ is finite, it can be shown that there exists a suffix u' of u such that none of computations of M in u' is a valid and halting computation. Now using this fact, and the fact that $u \in L(\mathcal{P}_2)$ and using a similar reasoning as in the paper of [2], it can be shown that $u \notin L(\mathcal{P}_1)$.

Now assume that $L(M)$ is an infinite set and hence $L'(M)$ is also an infinite set. In the proof given in [2], the authors construct an infinite sequence $\tilde{w}$ of inputs to $\mathcal{P}_1$ and $\mathcal{P}_2$, which is a concatenation of sub-strings separated by $$; call these sub-strings as super blocks. The i^{th} super block is a concatenation of k_i sub-strings separated by $\#$; call these sub-strings as blocks. In their construction, each block is the same string w. In our construction, the blocks in a super block are all identical, but blocks in different super blocks are different. Each block in the ith super block is a concatenation of m_i number of a valid halting computation w_i of length n_i (separated by the symbol $@$) where $m_i = c \cdot d^{n_i}$ and c, d are constants given earlier. These constants ensure that each block is a finite input string to $\mathcal{R}$ which is accepted with probability at least $(1-\epsilon)$. Now, the constants $k_1, ..., k_i, ..$ are chosen as in [2] so that $\tilde{w}$ is accepted by both $\mathcal{P}_1$ and $\mathcal{P}_2$ with non-zero probability. Now, we chose the computations $w_1, ..., w_i, ...$ such that $\tilde{w}$ is also accepted by $\mathcal{P}_3$ with non-zero probability. It should be easy to see that the probability that $\mathcal{P}_3$ accepts $\tilde{w}$ is given $\prod_{i>0}(p_i)$ where $p_i = (1-\lambda^{n_i})^{m_i \cdot k_i}$. We chose n_i so that the i^{th} term in the above product is $> (1-\frac{1}{2^i})$. From this it would follow that the above product has non-zero value. Assuming λ to be very small, it is easily seen that $p_i > (1-m_i \cdot \lambda^{n_i})^{ki}$. By substituting for $m_i = d^{n_i}$, we see that $p_i > (1-(d \cdot \lambda)^{n_i})^{k_i}$. Here d, k_i are given and λ is a small constant such that $d \cdot \lambda << \frac{1}{2}$. Now, it should be easy to see that we can chose a sufficiently large n_i so that $(1-(d \cdot \lambda)^{n_i})^{k_i} > (1-\frac{1}{2^i})$ and such that there is a valid halting computation w_i of length n_i. $\square$

C Proofs of Section 4

C.1 Proof of Proposition 4

Lemma 1, Theorem 1 and Lemma 2 already imply that $\mathbb{L}(\mathrm{PBA}^{=1}) \subsetneq \mathcal{G}_\delta = \mathrm{Deterministic}$. We only need to show that $\mathsf{Regular} \cap \mathbb{L}(\mathrm{PBA}^{=1}) = \mathsf{Regular} \cap \mathrm{Deterministic}$. Since every language in $\mathbb{L}(\mathrm{PBA}^{=1})$ is deterministic, we get immediately that $\mathsf{Regular} \cap \mathbb{L}(\mathrm{PBA}^{=1}) \subseteq \mathsf{Regular} \cap \mathrm{Deterministic}$. For the reverse inclusion, note that every ω-regular, deterministic language is recognizable by a finite-state deterministic Büchi automaton. It is easy to see that any language recognized by a deterministic finite-state Büchi automaton is in $\mathbb{L}(\mathrm{PBA}^{=1})$. The result follows.

C.2 Proof of Corollary 1

Let $\mathcal{B}_1 = (Q^1, q_s^1, Q_f^1, \delta^1)$ and $\mathcal{B}_2 = (Q^2, q_s^2, Q_f^2, \delta^2)$ be two PBAs, and we assume without loss of generality that $Q^1 \cap Q^2 = \emptyset$. We will present construction of PBAs that recognize the union and intersection of these languages under the almost sure semantics.

We begin by first considering the construction for union. Now by Lemma 4, we know that there are FPMs $\mathcal{M}_1$ and $\mathcal{M}_2$ such that $\mathcal{L}_{=1}(\mathcal{B}_i) = \Sigma^\omega \setminus \mathcal{L}_{>0}(\mathcal{M}_i)$. Now, we had shown in [5] that there is a FPM $\mathcal{M} = (Q, q_s, Q_f, \delta)$ such that for any word α, $\mu^{acc}_{\mathcal{M},\alpha} = \mu^{acc}_{\mathcal{M}_1,\alpha} \times \mu^{acc}_{\mathcal{M}_2,\alpha}$. It is easy to see that $\mathcal{L}_{>0}(\mathcal{M}) = \mathcal{L}_{>0}(\mathcal{M}_1) \cap \mathcal{L}_{>0}(\mathcal{M}_2)$.

Now, the FPM $\mathcal{M}$ can be easily "complemented". If q_r is the reject state of $\mathcal{M}$, then let Consider the PBA $\overline{\mathcal{M}} = (Q, q_s, \{q_r\}, \delta)$; clearly $\mathcal{L}_{=1}(\overline{\mathcal{M}}) = \mathcal{L}_{>0}(\mathcal{M})$. Thus, by DeMorgan Laws, $\mathcal{L}_{=1}(\overline{\mathcal{M}}) = \mathcal{L}_{=1}(\mathcal{B}_1) \cup \mathcal{L}_{=1}(\mathcal{B}_2)$.

The PBA recognizing the intersection of the languages recognized by $\mathcal{B}_1$ and $\mathcal{B}_2$ with respect to almost-sure semantics does the following: on an input α, with probability $\frac{1}{2}$ it runs $\mathcal{B}_1$ on α, and with probability $\frac{1}{2}$ it runs $\mathcal{B}_2$. Clearly, such a machine will accept (with respect to almost-sure semantics) iff both $\mathcal{B}_1$ and $\mathcal{B}_2$ accept. Formally, $\mathcal{B} = (Q, q_s, Q_f, \delta)$ is given by

- $Q = Q^1 \cup Q^2 \cup \{q_s\}$ where $q_s \notin Q^1 \cup Q^2$
- $Q_f = Q^1_f \cup Q^2_f$
- The transition relation δ is defined as follows
  - For $q \in Q^1$, $\delta(q_s, a, q) = \frac{1}{2}\delta^1(q^1_s, a, q)$, and for $q \in Q^2$, $\delta(q_s, a, q) = \frac{1}{2}\delta^2(q^2_s, a, q)$
  - For $q, q' \in Q^1$, $\delta(q, a, q') = \delta^1(q, a, q')$ and for $q, q' \in Q^2$, $\delta(q, a, q') = \delta^2(q, a, q')$.

C.3 Proof of Theorem 4

Observe first that given PBAs $\mathcal{B}_1$ and $\mathcal{B}_2$, there are (constructible) FPMs $\mathcal{M}_1$ and $\mathcal{M}_2$ such that $\mathcal{L}_{=1}(\mathcal{B}_i) = \Sigma^\omega \setminus \mathcal{L}_{>0}(\mathcal{M}_i)$ for $i = 1, 2$ (see Lemma 4). Thus, $\mathcal{L}_{=1}(\mathcal{B}_1) \subseteq \mathcal{L}_{=1}(\mathcal{B}_2)$ iff $\mathcal{L}_{>0}(\mathcal{M}_2) \subseteq \mathcal{L}_{=1}(\mathcal{M}_1)$. The upper bound then follows from the upper bound of the containment of PBAs under probable semantics.

The lower bound is shown by a reduction from emptiness-checking of probable semantics. Recall from the proof of the fact that $\mathbb{L}(\mathrm{PBA}^{>0}) = \mathsf{BCl}(\mathbb{L}(\mathrm{PBA}^{=1}))$ (Theorem 1) that given a PBA $\mathcal{B}$, there are $\mathcal{B}^+_1, \mathcal{B}^+_2, \ldots \mathcal{B}^+_m$ and $\mathcal{B}^-_1, \mathcal{B}^-_2 \ldots \mathcal{B}^-_m$ such that

$$\mathcal{L}_{>0}(\mathcal{B}) = \bigcup_{1 \le i \le m} \mathcal{L}_{=1}(\mathcal{B}^+_i) \cap (\Sigma^\omega \setminus \mathcal{L}_{=1}(\mathcal{B}^-_i)).$$

Furthermore, results of [7] and the construction in the proof of Theorem 1 implies that $\mathcal{B}^+_i$ and $\mathcal{B}^-_i$ are constructible. Now, $\mathcal{L}_{>0}(\mathcal{B}) = \emptyset$ iff for each i, $\mathcal{L}_{=1}(\mathcal{B}^+_i) \cap (\Sigma^\omega \setminus \mathcal{L}_{=1}(\mathcal{B}^-_i)) = \emptyset$. The lower bound now follows from the observation that $\mathcal{L}_{=1}(\mathcal{B}^+_i) \cap (\Sigma^\omega \setminus \mathcal{L}_{=1}(\mathcal{B}^-_i)) = \emptyset$ iff $\mathcal{L}_{=1}(\mathcal{B}^+_i) \subseteq \mathcal{L}_{=1}(\mathcal{B}^-_i)$.

D Proofs from Section 5

D.1 Proof of Lemma 5

Given $n \in \mathbb{N}$, let L_n be the safety language in which for every a there is a c after exactly n-steps. In other words, $\mathsf{L}_n = \Sigma^\omega \setminus (\Sigma^* a \Sigma^n \{a, b\} \Sigma^\omega)$. This could model, for instance, the property "every request a is answered after exactly n-steps". We can build a deterministic Büchi automaton for L_n and the number of states of such a automaton is $O(2^n)$. We could build a HPBA $\mathcal{B}_n$ with $O(n)$ state such that $\mathsf{L}_n = \mathcal{L}_{=1}(\mathcal{B}_n)$. The HPBA $\mathcal{B}_n$ will be an FPM also. The construction of $\mathcal{B}_n$ is as follows— $\mathcal{B}_n$ scans the input and upon encountering a, $\mathcal{B}_n$ decides with probability $\frac{1}{2}$ to check if there is a c after n steps and with probability $\frac{1}{2}$, $\mathcal{B}_n$ decides to continue scanning the rest of the input. In the former case, if the check $\mathcal{B}_n$ reveals an error then $\mathcal{B}_n$ rejects the input; otherwise $\mathcal{B}_n$ accepts the input.

D.2 Proof of Lemma 6

The HPBA we will construct will actually an FPM. The following construction is given in [4]. Let $\Sigma = \{\mathbf{0}, \mathbf{1}\}$. Let $Q = \{q_0, q_1, q_r\}$ and $\delta : Q \times \Sigma \times Q \to [0,1]$ be defined as follows. The states q_r and q_1 are absorbing, *i.e.*, $\delta(q_r, \mathbf{0}, q_r) = \delta(q_r, \mathbf{1}, q_r) = \delta(q_1, \mathbf{0}, q_1) = \delta(q_1, \mathbf{1}, q_1) = 1$. For transitions out of q_0, $\delta(q_0, \mathbf{0}, q_0) = \delta(q_0, \mathbf{0}, q_r) = \delta(q_0, \mathbf{1}, q_0) = \delta(q_0, \mathbf{1}, q_1) = \frac{1}{2}$ Consider the *FPM* $\mathcal{M}_{\mathsf{Id}} = (Q, q_0, \{q_0, q_1\}, \delta)$. $\mathcal{M}_{\mathsf{Id}}$ can be seen to be 2-hierarchical with $\mathsf{rk}(q_0) = 0, \mathsf{rk}(q_1) = 1$ and $\mathsf{rk}(q_r) = 2$. Given $\alpha = a_0 a_1 \ldots$, it can be shown that $\mu^{acc}_{\mathcal{M}_{\mathsf{Id}}, \alpha} = \mathsf{bin}(\alpha)$ where $\mathsf{bin}(\alpha)$ is the real number: $\sum_i \frac{\mathsf{num}(a_i)}{2^{i+1}}$ where $\mathsf{num}(\mathbf{0})$ is the integer 0 and $\mathsf{num}(\mathbf{1})$ is the integer 1.

Now, consider the FPM $\mathcal{M}_{\mathsf{Id}} \circ \mathcal{M}_{\mathsf{Id}}$ constructed as follows. The states of this FPM are $\{q_0, q_1\} \times \{q_0, q_1\} \cup q_{r_{\mathsf{new}}}$. The initial state is (q_0, q_0) and the reject state is $q_{r_{\mathsf{new}}}$. The transition probabilities, from the state (q_{i_1}, q_{j_1}) on input $a \in \{\mathbf{0}, \mathbf{1}\}$ is defined as follows— to state (q_{i_2}, q_{j_2}) the transition probability is $\delta(q_{i_1}, a, q_{i_2}) \times \delta(q_{j_1}, a, q_{j_2})$ and to state $q_{r_{\mathsf{new}}}$ the transition probability is $1 - \sum_{i_2, j_2 \in \{0,1\}} \delta(q_{i_1}, a, q_{i_2}) \times \delta(q_{j_1}, a, q_{j_2})$. The state $q_{r_{\mathsf{new}}}$ is absorbing. The FPM $\mathcal{M}_{\mathsf{Id}} \circ \mathcal{M}_{\mathsf{Id}}$ can be seen to be hierarchical with $\mathsf{rk}(q_{i_1}, q_{i_2}) = i_1 + i_2$. Furthermore, it can be shown that on word α, $\mu^{acc}_{\mathcal{M}_{\mathsf{Id}} \circ \mathcal{M}_{\mathsf{Id}}, \alpha} = (\mathsf{bin}(\alpha))^2$. Thus, $\mathcal{L}_{>\frac{1}{2}}(\mathcal{M}_{\mathsf{Id}} \circ \mathcal{M}_{\mathsf{Id}}) = \{\alpha \mid \mathsf{bin}(\alpha) > \sqrt{\frac{1}{2}}\}$ and $\mathcal{L}_{\geq\frac{1}{2}}(\mathcal{M}_{\mathsf{Id}} \circ \mathcal{M}_{\mathsf{Id}}) = \{\alpha \mid \mathsf{bin}(\alpha) \geq \sqrt{\frac{1}{2}}\}$; both of which are not ω-regular.

D.3 Proof of Theorem 5

We begin by showing that every ω-regular language can be recognized by a hierarchical PBA; this is the content of the next Lemma.

Lemma 8. For every ω-regular language L, there is a hierarchical PBA $\mathcal{B}$ such that $\mathsf{L} = \mathcal{L}_{>0}(\mathcal{B})$.

Proof. Let $\mathcal{R} = (Q, q_s, F, \Delta)$ be a deterministic Rabin automaton recognizing L, where $F = \{(B_1, G_1), \ldots (B_k, G_k)\}$. The hierarchical PBA will, intuitively, in the first step choose the pair (B_i, G_i) that will be satisfied in the run, and then ensure that the measure of paths that visit B_i infinitely often is 0. Formally, $\mathcal{B} = (Q', q'_s, Q'_f, \delta')$ is given as follows.

- $Q' = \{q'_s, q'_r\} \cup (\{1, \ldots k\} \times Q)$, where $q'_s, q'_r \notin Q$
- $Q'_f = \bigcup_{i=1}^{k} (\{i\} \times G_i)$
- The transition relation δ' is given by

- $\delta'(q'_s, a, (i,q)) = \frac{1}{k}$ iff $(q_s, a, q) \in \Delta$
- For $q \notin B_i$, $\delta'((i,q), a, (i,q')) = 1$ iff $(q, a, q') \in \Delta$
- For $q \in B_i$, $\delta'((i,q), a, q'_r) = \frac{1}{2}$ for all $a \in \Sigma$, and $\delta'((i,q), a, (i,q')) = \frac{1}{2}$ iff $(q, a, q') \in \Delta$
- Finally, $\delta'(q'_r, a, q'_r) = 1$ for all $a \in \Sigma$.

Observe that on input α, there is only one run from a state (i,q). Runs from (i,q) that visit a state in (i, B_i) infinitely often have measure 0. Therefore, runs from (i,q) that visit (i, G_i) infinitely often with strictly positive measure must visit (i, B_i) only finitely many times. From this, it is easy to see that $\mathcal{L}_{>0}(\mathcal{B}) = \mathcal{L}(\mathcal{R}) = \mathsf{L}$. Finally, we point out that $\mathcal{B}$ is a $k+1$-level hierarchical PBA. This is witnessed by the ranking function rk defined as follows — $\mathsf{rk}(q'_s) = 0$, $\mathsf{rk}((i,q)) = i$, and $\mathsf{rk}(q'_r) = k+1$. (**End proof of the lemma**.) □

We will now complete the proof of Theorem 5 by showing the other direction. Thus, we need to show that every language in $\mathbb{L}(\mathrm{HPBA}^{>0})$ is ω-regular. The proof depends on the following Claim.

Claim: For any hierarchical PBA $\mathcal{B} = (Q, q_s, Q_f, \delta)$ and any word $\alpha \in \Sigma^\omega$, $\alpha \in \mathcal{L}_{>0}(\mathcal{B})$ $q_i \in Q_f$ for infinitely many $i \in \mathbb{N}$, $\delta(q_i, \alpha[i], q_{i+1}) > 0$ for all $i \in \mathbb{N}$ and $\exists j \geq 0$ such that $\delta(q_i, \alpha[i], q_{i+1}) = 1$ all $i \geq j$.

Proof of the claim: Let $\mathcal{B}$ be a k-level hierarchical PBA with compatible ranking function rk. Let $Q_j = \{q \in Q \,|\, \mathsf{rk}(q) = j\}$. The proof will proceed by induction on the level k.

Base Case: Suppose $k = 0$. Based on the definition of hierarchical PBAs, this means that $\mathcal{B}$ is a deterministic Büchi automaton, i.e., for all $q, q' \in Q$ and $a \in \Sigma$, either $\delta(q, a, q') = 1$ or $\delta(q, a, q') = 0$. Thus, the lemma clearly holds in this case.

Induction Step: Let $\alpha \in \Sigma^\omega$ be such that $\alpha \in \mathcal{L}_{>0}(\mathcal{B})$, with $\mu^{acc}_{\mathcal{B},\alpha} = x > 0$. Observe that for every i, $|\mathsf{post}(q_s, \alpha[0,i]) \cap Q_0| \leq 1$. There are two cases to consider.

Case 1 Suppose $|\mathsf{post}(q_s, \alpha[0,i]) \cap Q_0| = 1$ for all i; let us denote the unique state in $\mathsf{post}(q_s, \alpha[0,i]) \cap Q_0$ by q_i. Suppose in addition, there is a j such that for all $\ell > j$, $\delta(q_\ell, \alpha[\ell], q_{\ell+1}) = 1$. Then clearly the sequence $q_0, q_1, \ldots$ satisfies the conditions of the lemma.

Case 2 Suppose Case 1 does not hold. Then there are two possibilities. The first possibility is that there is a i_0 such that $\mathsf{post}(q_s, \alpha[0,i_0]) \cap Q_0 = \emptyset$. The second possibility is that for every j, there is an $\ell > j$ such that $\delta(q_\ell, \alpha[\ell], q_{\ell+1}) < 1$, where once again we are denoting the unique state of Q_0 in $\mathsf{post}(q_s, \alpha[0,\ell])$ by q_ℓ. In this second subcase, there must then exist an i_0 such that $\delta_u(q_s, q_{i_0}) < x$, where $u = \alpha[0, i_0]$.

Now, based on the definition of i_0 given for the two subcases above, it must be the case that for some state $q \in \mathsf{post}(q_s, \alpha[0,i_0]) \setminus Q_0$, the measure of accepting runs from q on the word $\alpha[i_0{+}1]\alpha[i_0{+}2]\cdots$ is non-zero. Consider the hierarchical PBA $\mathcal{B}' = (Q', q, Q'_f, \delta')$, where $Q' = Q \setminus Q_0$, $Q'_f = Q_f \setminus Q_0$ and $\delta' = \delta|_{Q' \times \Sigma \times Q'}$. Clearly, $\mathcal{B}'$ is a $k-1$-level hierarchical PBA, and thus by induction hypothesis, the string $\alpha[i_0+1]\alpha[i_0+2]\cdots$ has a run ρ satisfying the conditions in the lemma. The desired run for α (in PBA $\mathcal{B}$) satisfying the conditions in the lemma is obtained by concatenating the run from q_s to q on $\alpha[0, i_0]$ with ρ. (**End proof of claim).** □

We now proceed with the main theorem. Let $\mathcal{B} = (Q, q_s, Q_f, \delta)$. Consider the finite-state Büchi automaton $\mathcal{A} = (Q', q'_s, Q'_f, \Delta')$ where Q' is the set $Q \times \{0,1\}$, $q'_s = (q_s, 0)$, $Q'_f = \{(q,1) \,|\, q \in Q_f\}$, and Δ' is defined as follows. For each $q_1, q_2 \in Q$,

- $((q_1,0), a, (q_2,0)) \in \Delta'$ iff $\delta(q_1,q_2) > 0$.
- $((q_1,0), a, (q_2,1)) \in \Delta'$ iff $\delta(q_1,q_2) > 0$.
- $((q_1,1), a, (q_2,1)) \in \Delta'$ iff $\delta(q_1,q_2) = 1$.
- $((q_1,1), a, (q_2,0)) \in \Delta'$ iff never.

The claim above immediately implies that $\mathcal{L}_{>0}(\mathcal{B})$ is the language recognized by $\mathcal{A}$ and hence is ω-regular.

D.4 Proof of Theorem 6

(Upper Bounds). First note since $\mathcal{B}$ is hierarchical, the language $\mathcal{L}_{>0}(\mathcal{B})$ is ω-regular (see Theorem 5). The proof of Theorem 5 also allows us to construct a finite-state Büchi automata $\mathcal{A}$ such that a) $\mathcal{L}_{>0}(\mathcal{B})$ is the language recognized by $\mathcal{A}$ and b) the size of the automaton $\mathcal{A}$ is at-most twice the size of the automaton $\mathcal{B}$. Furthermore, the construction can be carried out in $\mathbf{NL}$. Since the emptiness problem of finite-state Büchi automata is in $\mathbf{NL}$ and the universality problem is in $\mathbf{PSPACE}$, we immediately get that the desired upper bounds.

(Lower Bounds). Please note that the $\mathbf{NL}$-hardness of the emptiness problem can be proved easily from the emptiness problem of deterministic finite state machines. For the universality problem, we make the following claim.

Claim: Given an FPM $\mathcal{M}$ such that the $\mathcal{M}$ is also a hierarchical PBA, the problem of deciding whether $\mathcal{L}_{=1}(\mathcal{M})$ is empty is **PSPACE**-hard.

Before, we proceed to prove the claim, we first show how the lower bound follows from the reduction. Given an FPM $\mathcal{M} = (Q, q_s, Q_f, \delta)$ with reject state q_r, consider the PBA $\overline{\mathcal{M}} = (Q, q_s, \{q_r\}, \delta)$ obtained by taking the reject state of $\mathcal{M}$ as the unique final state of $\overline{\mathcal{M}}$. Clearly,

1. $\overline{\mathcal{M}}$ is HPBA if $\mathcal{M}$ is.
2. $\mathcal{L}_{>0}(\overline{\mathcal{M}})$ is universal iff $\mathcal{L}_{=1}(\mathcal{M})$ is empty.

From these two observations the desired result will follow if we can prove the claim. We now prove the claim.
Proof of the claim. We show that there is a polynomial time bounded reduction from every language in PSPACE to the language

$$\{(\mathcal{M}, \Sigma) \mid \mathcal{M} \text{ is an FPM on } \Sigma, \mathcal{M} \text{ is a HPBA and } \mathcal{L}_{=1}(\mathcal{M}) = \emptyset\}.$$

Consider a language $L \in PSPACE$ and $\mathbb{T}$ be a single tape deterministic Turing machine that accepts L in space $p(n)$ for some polynomial p where n is the length of its input. We assume that $\mathbb{T}$ accepts an input by halting in a specific final state q_f and $\mathbb{T}$ rejects an input by not halting. Let $\mathbb{T}$ be given by the tuple $(Q, \Delta, \Gamma, \delta, q_0, q_f)$. Here Q is the set of states of the finite control of $\mathbb{T}$; Δ, Γ are the input and tape alphabets and

$\Delta \subseteq \Gamma$ and the blank symbol # is in $\Gamma - \Delta$; $\delta : Q \times \Gamma \rightarrow \Gamma \times Q \times \{Left, Right\}$; q_0 is the initial state and q_f is the final state. Each tuple $\delta(q, a) = (a', q', d)$ indicates that when $\mathbb{T}$ is in state q, scanning a cell containing the symbol a, then $\mathbb{T}$ writes value a' in the current cell, changes to state q' and moves in the direction d.

Let $\Phi' = \Gamma \times Q$ and $\Phi = \Phi' \cup \Gamma$. We call members of Φ' composite symbols. A configuration of $\mathbb{T}$, on an input of length n, is a string of symbols, of length $p(n)$, drawn from Φ. We can define a valid configuration in the standard way. In each valid configuration there can be only one composite symbol (i.e.,from Φ') and that indicates the head position of $\mathbb{T}$. A computation of $\mathbb{T}$ is a sequence of configurations which is either finite or infinite depending on whether the input is accepted or not. A computation starts in an initial configuration and each succeeding configuration is obtained by one move of $\mathbb{T}$ from the previous configuration. The initial configuration contains the input string and the first symbol in it is from $\Gamma \times Q$ indicating its head position is on the first cell.

For given input σ, we construct a FPM $\mathcal{M}_\sigma$ such that $\mathcal{M}_\sigma$ is a 2-HPBA and $\mathcal{M}_\sigma$ accepts some infinite input with probability 1 iff $\mathbb{T}$ rejects σ, i.e., $\mathbb{T}$ does not halt on σ. Let σ be an input to $\mathbb{T}$ of length n and let $m = p(n)$. A state of the automaton $\mathcal{M}_\sigma$ is a pair of the form (i, a) where $0 \leq i < m$ and $a \in \Phi$, or is in $\{q_s, q_r\}$; here q_s is the initial state and is of rank 0 and q_r is the reject state and is of rank 2. The rank of states $\{(i, a) \mid 0 \leq i < m \text{ and } a \in \Phi\}$ will be 1. Intuitively, if $\mathcal{M}_\sigma$ is in state (i, a) that denotes that i^{th} element of the current configuration of the computation of $\mathbb{T}$ has value a. Note that a is in Φ' or is in Γ. The input alphabet to $\mathcal{M}_\sigma$ is the set $\{0, ..., m-1\} \times \Phi' \times \{left, right\}$ together with an additional input symbol τ; that is each input to the automaton is of the form $(i, (b, q), d)$ or is τ where $d \in \{left, right\}$.

Let $\sigma = \sigma_0, ..., \sigma_{n-1}$ be the input to $\mathbb{T}$. The transitions of $\mathcal{M}_\sigma$ are defined as follows. From the initial state q_s, on input τ, there are transitions to the states (i, r_i) , for each $i \in \{0, ..., m-1\}$ where $r_i = \sigma_i$ for $i < n$, and is the blank symbol otherwise; the probability of each of these transitions is $\frac{1}{m}$. Thus the input τ sets up the initial configuration when $\mathcal{M}_\sigma$ is in the initial state q_r. From every other state on input τ there is a transition to the reject state q_r with probability 1. From any state of the form (j, b), where $b \in \Gamma$, on input symbol of the form $(i, (a, q'), d)$ the transitions are defined as follows: if either $i = j - 1$ and $d = right$, or $i = j + 1$ and $d = left$ then the transition is to the state $(j, (b, q'))$; otherwise the transition is back to (j, b); in both cases the probability of the transition is 1. From any state of the form $(j, (b, q))$ on input of the form $(i, (a, q'), d)$ the transition is defined as follows: if $i = j$ and $\mathbb{T}$ changes its state from q to q' and writes a in the scanned cell, on seeing the symbol b, then there is a transition to the automaton state (j, a); otherwise, the transition is to q_r; in either case, the probability of the transition is 1. Note that if $\mathbb{T}$ halts then also there is a transition to q_r.

Suppose σ is rejected, i.e., $\mathbb{T}$ does not terminate on σ and the composite symbols in each successive configuration of the infinite computation are $a_0, a_1, ...$ and they occur in positions $i_0, ...$ and the direction of the head movement is given by $d_0, ...$ respectively then $\mathcal{M}_\sigma$ accepts the infinite string $(i_0, a_0, d_0), .., (i_k, a_k, d_k), ...$ with probability 1 and accepts all others with probability less than 1. It is not difficult to see that if σ is accepted by $\mathbb{T}$, all input strings are accepted by $\mathcal{M}_\sigma$ with probability less than 1. The above reduction is clearly polynomial time bounded. **(End proof of the claim.)** □

D.5 Proof of Theorem 7

The inclusion $\mathsf{Regular} \cap \mathrm{Deterministic} \subseteq \mathbb{L}(\mathrm{HPBA}^{=1})$ follows immediately from the fact that any language in $\mathsf{Regular} \cap \mathrm{Deterministic}$ is recognizable by a finite-state deterministic Büchi automaton. For the reverse inclusion $\mathbb{L}(\mathrm{HPBA}^{=1}) \subseteq \mathsf{Regular} \cap \mathrm{Deterministic}$, note that since $\mathbb{L}(\mathrm{PBA}^{=1}) \subseteq \mathrm{Deterministic}$, it suffices to show that $\mathbb{L}(\mathrm{HPBA}^{=1}) \subseteq \mathsf{Regular}$. Now, given $\mathsf{L} \in \mathbb{L}(\mathrm{HPBA}^{=1})$, Lemma 4 immediately implies that there is an FPM $\mathcal{M}$ such that $\mathcal{L}_{>0}(\mathcal{M}) = \Sigma^\omega \setminus \mathsf{L}$. Furthermore, it is easy to see from the proof of Lemma 4 that we can take $\mathcal{M}$ to be hierarchical given that $\mathsf{L} \in \mathbb{L}(\mathrm{HPBA}^{=1})$. Now, thanks to Theorem 5, $\mathcal{L}_{>0}(\mathcal{M})$ is ω-regular which implies that L is also ω-regular.

D.6 Proof of Theorem 8

(Upper Bounds.) The upper bounds are obtained by constructing the FPM $\mathcal{M}$ such that $\mathcal{L}_{=1}(\mathcal{B}) = \Sigma^\omega \setminus \mathcal{L}_{>0}(\mathcal{M})$ as in the proof of Lemma 4. Now, $\mathcal{M}$ is hierarchical if $\mathcal{B}$ is hierarchical. The result now follows immediately from Theorem 6.

(Lower Bounds.) The $\mathbf{NL}$-hardness of checking universality can be shown from $\mathbf{NL}$-hardness of checking emptiness of deterministic finite state machines. Please recall that in the proof of Theorem 6, we had shown that given an FPM $\mathcal{M}$ such that $\mathcal{M}$ is a HPBA, the problem of checking whether $\mathcal{L}_{=1}(\mathcal{M})$ is empty is **PSPACE**-hard. Thus, it follows immediately that checking emptiness of $\mathcal{L}_{=1}(\mathcal{B})$ for a HPBA is **PSPACE**-hard.