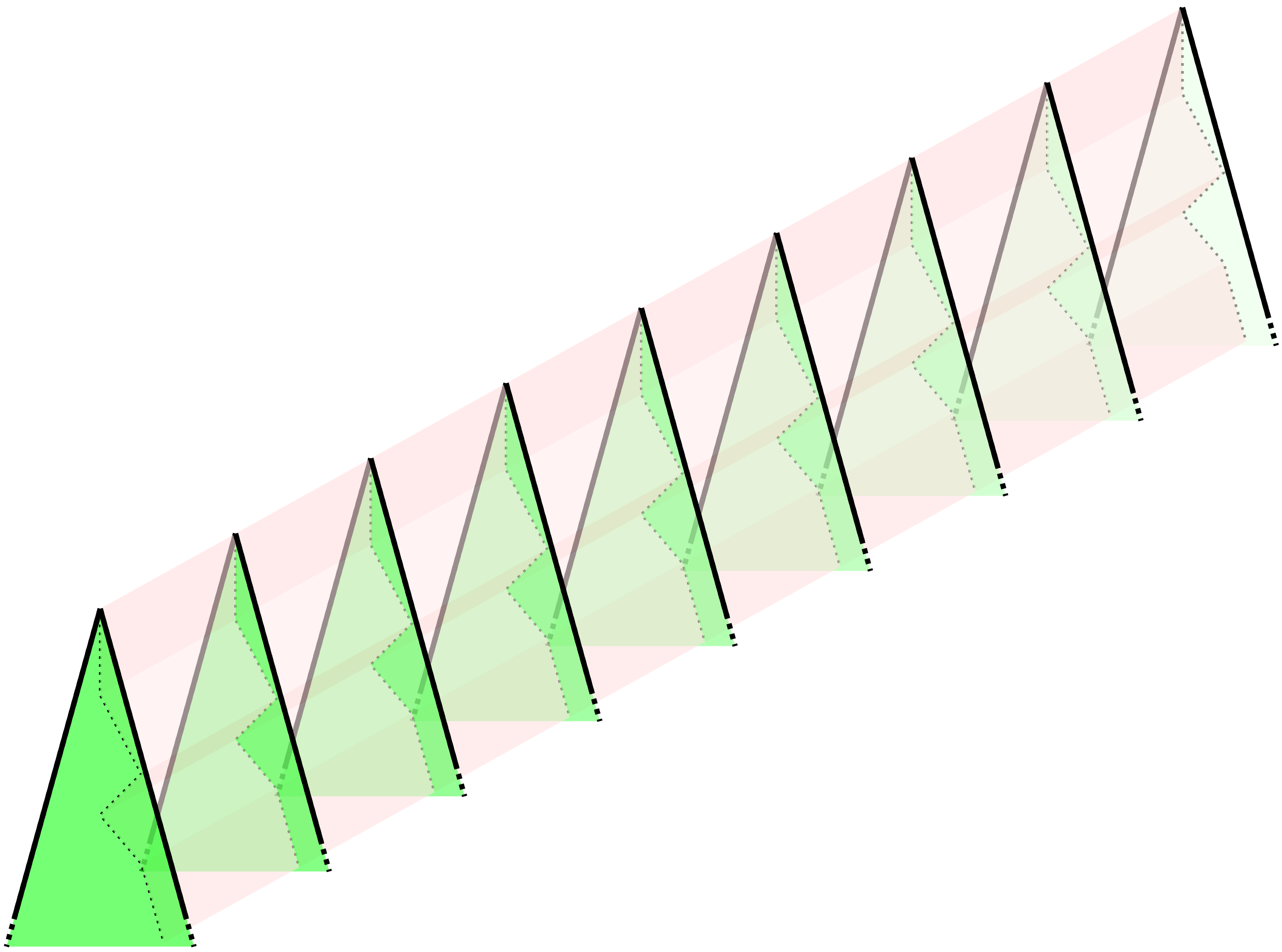
Master's Thesis

Title:	**Automata-theoretic Synthesis for Probabilistic Environments**
Author:	Christoph Welzel
Date:	April 16, 2019
Institute:	Informaik 7, RWTH Aachen
Primary Reviewer:	Dr. Christof Löding
Secondary Reviewer:	Prof. Joost-Pieter Katoen



Abstract

In this thesis we introduce and examine Weighted Descent Tree Automata, a class of tree automata for infinite trees that incorporate a notion of weighting on the individual paths of their runs and an aspect of alternation. We focus on their strength to model other probabilistic systems. Inspired by these results we obtain for specifications, given as Probabilistic Büchi Automata with almost-sure acceptance, decision procedures for deterministic strategy synthesis in Partially Observable Markov Decision Processes and antagonistic environments.

Acknowledgements

I want to thank Dr. Löding for supervising this thesis and offering helpful advice throughout its creation. As for my Bachelor's thesis, Prof. Katoen offers his aditional supervision for which I am grateful. For helpful discussions I am also indebted to Sebastian Junges and Benjamin Kaminski.

On a personal note, I extend my gratitude to my parents for their constant support, my siblings for providing a valuable change of perspective every now and then and a few fellow students: Karla, Johannes, Simon, Kerstin and Nick.

Contents

1. Introduction

1.1. Motivation

Relying on finite structures, called automata, to parse and classify finite and infinite objects, e.g. words and trees, is well established [Tho97; Rab72]. Automata on words and trees usually operate in a absolute fashion, i.e. from the current internal state of the automaton and the input the automaton proceeds into another state by some rule. These rules can be categorized to be deterministic or non-deterministic where deterministic rules uniquely identify one successor state from the current internal state and the read input while non-deterministic rules allow to choose from multiple possible successor states exactly one. The source of these choices is conceived from some omniscient oracle which provides the correct choice if one exists. Some research focuses on making the participation of the oracle more transparent. Either by finding equivalent deterministic automata [Pit07] or by considering probabilistic behavior of the oracle [CHS14; CHS11; BG05] introducing probabilistic automata.

Moreover, probabilistic behavior has proven invaluable for the analysis of various domains, e.g. decision making [CKL94] or modelling environments systems reside in [BK08]. Given such a model of a probabilistic environment it is a common task to determine how well a certain actor behaves [BK08] or construct a behavior which meets certain demands [Sch06; CJH04]. Additionally, an actor may show probabilistic behavior as well [Cha+15]. Connections between models of probabilistic environments and probabilistic automata are already established [BBG08].

Thirdly, depending on the environment it is reasonable to restrict the information an actor may observe. This concept of *incomplete or partial information* is incorporated in models for environments [CDH10] and connected to certain structural properties of automata [KV99], namely alternation.

In this thesis we discuss approaches around the theory of automata to tackle the *synthesis problem*. The synthesis problem is generally understood as the task to derive from a given specification a behavior which satisfy this specification [Chu62]. Automata on infinite words and trees are well-understood tools for reasoning about the synthesis problem. The common approach for solving synthesis problems by means of automata theory follows three steps: expressing the specification as language of an automata, determinising this automata and exectuing it in parallel on all paths of a tree [KV99; Sch06; Rab72; Zie98]. However, probabilistic environments allow for a relaxed demand on the synthesised behavior. It suffices to act according to a specification in almost-all or some situations [Sch06; BK08; CDH10]. We try to translate these approaches to probabilistic automata for words and trees. Therefore, we incorporate alternation into tree automata with probabilistic behavior and examine the resulting model and its implications on the synthesis problem for probabilistic environments with special consideration of specifications in form of probabilistic word automata.

1.2. Structure and Contributions of this Thesis

This thesis is separated in five parts. The introduction deals with the motivation of the examined approach, the related work and preliminaries introducing concepts of probability theory and notations regarding finite and infinite words. The follwing, second, chapter recalls known results of the theory of infinite words and associated word automata. Additionally, we discuss the less colloquial concepts of probabilistic automata for infinite words. In the next chapter we present results of the theory of automata on infinite trees. Notably, we introduce and use the concept of alternation and present its close connection with graph games. The rest of the chapter is dedicated to the introduction of Weighted Descent Tree Automata and the examination of basic results of the associated theory and the possibilities to model other probabilistic systems. The fourth chapter deals with the synthesis problem associated with specifications which can be described as ω-regular languages and those which are described in terms of probabilistic word automata. Both probabilistic and antagonistic environments are considered. The last chapter contains a conclusion which includes an evaluation of the results and a discussion of ideas for further research.

To the best of our knowledge this thesis achieves to contribute genuine new results for the synthesis problem where the specification is expressed as almost-surely accepting Probabilistic Büchi Automaton. Namely, we solve the strategy synthesis for such specifications for Partially Observable Markov Decision Processes and for antagonistic environments in Theorem 37 and Theorem 40 respectively. These results are inspired by the concepts of Weighted Descent Tree Automata which are also a contribution of this thesis (cp. Section 3.2). For all our results we are committed to transparently illustrate the origin of used arguments by selectively presenting proofs of known results.

1.3. Related Work

Relaxing the notion of acceptance for word automata on infinite words is a relatively recent approach [BBG09; Grö08; BG05; BBG08]. Valuation of runs of automata is also used for weighted automata [AK11] and explored in the context of qualitative satisfaction rather than absolute correctness [ABK11] which is also the basis for synthesis results in stochastic environments [AK16].

The probabilistic approach is translated to trees in [CHS14; CHS11] in two different ways. On the one hand, a weighting in form of probabilities on paths is considered and, on the other hand, also it is examined to choose transitions probabilistically which is semantically closer to the approach of probabilistic word automata. Another related concept is discussed in [BGK17]. Again, the individual paths of a run of an automaton are weighted in terms of probabilites but the acceptance conditions are more diverse. Different notions of acceptance, including a qualitative measure on accepting paths and an universal demand on accepted path are intertwined. However, both these approaches lack a concept of alternation as discussed in this thesis. Especially, both models enjoy decidable emptiness problems which contrasts results of this thesis for various weighted tree automata with alternation. A different perpective on relaxation of accepting paths in runs of tree automata is proposed in [BNN91] by considering runs accepting if the cardinality of accepted paths exceeds or deceeds a given cardinal number. The required argumentation presents in a more logic-based and set-theoretical framework.

Furthermore, notions of imperfect information are broadly applied in different settings. Used in graph games partial observability lifts the computation of winning regions even for simple winning conditions, e.g. Safety-conditions, into exponential complexity [BD08]. Also, partial observability is very prominently used in Markov Decision Processes [Cha+15; CKL94; CDH10] but entails harsh consequences on the analysis, e.g. computing almost-surely winning strategies for Parity-objectives becomes undecidable for partial observability while there are polynomial solutions in the case of complete information [CDH10; CJH04; Bai+04]. This entails interesting consequences for our model of tree automata since we are able to pinpoint a structural property of alternation in Weighted Descent Tree Automata that precisely corresponds with partial observability and therefore translates these results into our automata model. This is also suggested in [KV99] where alternating tree automata are used to model information that are inaccessible for the actor, but still are part of the specification.

Moreover, considering probabilistic automata as representation for the specification of a synthesis problem is not yet widely adopted. Logics that incorporate quantification for a probability, e.g. PCTL, state the probability of the the considered environment to satisfy a condition and do not use stoachastic behavior inherently to the specification [BK08]. Using probabilistic automata as specification is explored for timed models [NPZ17] or as concise representation of ω-regular properties [KV14]. To the best of our knowledge we pioneer in the approach to use the incomparable expressiveness of almost-surely accepting Probabilistic Büchi Automata to ω-regular languages for specifications of synthesis problems. Actually, the solution for qualitative strategy synthesis in Partially Observable Markov Decision Processes for a specification given as almost-surely accepting Probabilistic Büchi Automata fits well to the approach in [Cha+15] where reductions are examined that allow to remove probabilistic behavior for strategies or transitions in a graph by deterministic behavior. The associated randomness for these objects is given for "free". In this sense we allow for "free" randomness in the specification considering Partially Observable Markov Decision Processes.

1.4. Preliminaries

For a finite set $A = \{a_1, \ldots, a_n\}$ we consider a finite word over A as $a_{i_1}a_{i_2}\ldots a_{i_m}$ with $1 \leq i_j \leq n$ for all $1 \leq j \leq m$. ϵ denotes the unique word of length 0 and A^* the set of all finite words over A. We call $\mathcal{L} \subseteq A^*$ a language over finite words. For any two

two words $a_{i_1} \dots a_{i_m}$ and $a_{j_1} \dots a_{j_k}$ we denote with $a_{i_1} \dots a_{i_m} \cdot a_{j_1} \dots a_{j_k}$ the finite word $a_{i_1} \dots a_{i_m} a_{j_1} \dots a_{j_k}$. For two languages $\mathcal{L}_1$ and $\mathcal{L}_2$ we introduce

$$\mathcal{L}_1 \cdot \mathcal{L}_2 = \{u \cdot v : u \in \mathcal{L}_1, v \in \mathcal{L}_2\} \subseteq A^*$$

and for a word u and a language $\mathcal{L}$ we consider $u \cdot \mathcal{L}$ to be equivalent to $\{u\} \cdot \mathcal{L}$. If the semantics is still clear we might omit $\cdot$ in these expressions. Moreover, we introduce for all $i \geq 0$

$$\mathcal{L}^0 = \{\epsilon\} \text{ and } \mathcal{L}^{i+1} = \mathcal{L} \cdot \mathcal{L}^i.$$

Infinite words over A are similarly defined as finite words but requiring a countable index sequence $(i_k)_{k \in \mathbb{N}}$. We denote with A^ω the set of infinite words over A and languages of infinite words $\mathcal{L} \subseteq A^\omega$. For a language of *finite* words $\mathcal{L}_1 \subseteq A^*$ and a language of *infinite* words $\mathcal{L}_2 \subseteq A^\omega$ we define

$$\mathcal{L}_1 \cdot \mathcal{L}_2 = \{u \cdot \alpha : u \in \mathcal{L}_1, \alpha \in \mathcal{L}_2\} \subseteq A^\omega$$

and for a finite word $u \in A^*$ we allow the notation $u \cdot \mathcal{L}_2$ to express $\{u\} \cdot \mathcal{L}_2$. In general we use the convention to use latin letters to label finite words while greek letters are used for infinite words.

For two finite words $u, v \in A^*$ we denote with $\sqsubseteq$ the prefix relation, i.e.

$$u \sqsubseteq v \text{ if and only if there is } u' \in A^* \text{ such that } u \cdot u' = v.$$

We also use $\sqsubseteq$ for the notion that a finite word is a prefix of an infinite word. Hence, for $u \in A^*$ and $\alpha \in A^\omega$ we define

$$u \sqsubseteq \alpha \text{ if and only if there is } \beta \in A^\omega \text{ such that } u\beta = \alpha.$$

Considering a function $f : A \to B$ we allow to implicitly lift f to sets and finite and infinite words by elementwise application, namely

$$\begin{array}{ll} \text{for } C \subseteq A & f(C) = \{f(c) : c \in C\} \subseteq B \\ \text{for } u = a_1 \dots a_n \in A^* & f(u) = f(a_1) \dots f(a_n) \in B^* \\ \text{for } \alpha \in \alpha_1 \alpha_2 \dots \in A^\omega & f(\alpha) = f(\alpha_1) f(\alpha_2) \dots \in B^\omega. \end{array}$$

Additionally, we allow to lift a function $f : A^* \to B$ to infinite words such that

$$f(\alpha_1 \alpha_2 \alpha_3 \dots) = f(\alpha_1) f(\alpha_1 \alpha_2) f(\alpha_1 \alpha_2 \alpha_3) \dots.$$

Furthermore, we introduce for a function $f : A \to B$ the following notion

$$f\left[A' \mapsto b\right] \text{ with } f\left[A' \mapsto b\right](a) = \begin{cases} b & \text{if } a \in A', \\ f(a) & \text{otherwise.} \end{cases}$$

Also, we define $\uplus$ as an operator of disjoint union. Hence, for $A \uplus B$ we assume $A \cap B = \emptyset$. If this is not the case, we substitute the elements of B with newly defined elements such that $A \cap B = \emptyset$ holds. Also, we make use the operator Pot to construct the set of all subsets of the argument.

1.4.1. Probability Theory

In the following we introduce the basic notions of probability theory on which we rely in this thesis. We introduce these concepts in a concise fashion and refer the interested reader to [Kle06] and [Bau92]. For a given ground-set Ω we introduce a σ-algebra as a collection of sets closed under countable union and negation. Formally, we fix

Definition 1. ***σ-Algebra***:
For a ground-set Ω we call $\mathcal{F} \subseteq \mathrm{Pot}(\Omega)$ for Ω a σ-algebra if

1. $\Omega \in \mathcal{F}$,
2. for every $A \in \mathcal{F}$ we have $(\Omega \setminus A) \in \mathcal{F}$,
3. for a countable collection $(A_i)_{i\in\mathbb{N}}$ with $A_i \in \mathcal{F}$ we also have $(\cup_{i\in\mathbb{N}} A_i) \in \mathcal{F}$.

Additionally, we define

Definition 2. ***Trace***:
For a σ-algebra $\mathcal{F}$ for a ground-set Ω and a non-empty set $A \subseteq \Omega$ we define the A-trace of $\mathcal{F}$ as

$$\mathcal{F}_{|A} = \{B \cap A : B \in \mathcal{F}\}.$$

The trace of a σ-algebra is again a σ-algebra:

Lemma 1. *[Kle06, Theorem 1.26] For a σ-algebra $\mathcal{F}$ of a ground-set Ω and any non-empty set $A \subseteq \Omega$ holds that $\mathcal{F}_{|A}$ is a σ-algebra of the ground-set A.*

A probability space is defined by a set of possible results, a σ-algebra which describes those collection of events we can observe and a function which describes how probable a chosen observation is:

Definition 3. ***Probability Space***:
For a set Ω and a σ-algebra $\mathcal{F}$ for Ω we define a probability function as $\mu : \mathcal{F} \to [0, 1]$ such that

1. $\mu(\Omega) = 1$,
2. for a countable collection $(A_i)_{i\in\mathbb{N}}$ with $A_i \in \mathcal{F}$ for all $i \in \mathbb{N}$, $A_i \cap A_j = \emptyset$ for all $i, j \in \mathbb{N}$ with $i \neq j$ and $\mathbb{A} = \cup_{i\in\mathbb{N}} A_i$ we have

   $$\mu(\mathbb{A}) = \sum_{i\in\mathbb{N}} \mu(A_i).$$

   We call this property σ-additivity.

We call for such a μ the triple $(\Omega, \mathcal{F}, \mu)$ a probability space and $(\Omega, \mathcal{F})$ a measurable space. For any finite A we can additionally define

Probability Distribution as a function $P : A \to [0,1]$ such that $\sum_{\omega \in A} P(\omega) = 1$.

Support of a probability distribution P as the set of possible outcomes, i.e.

$$\mathrm{support}(P) = \{\omega \in A \mid P(\omega) > 0\}\,.$$

Distribution Set the set of probability distributions over A as $\mathcal{D}(A)$.

Dirac distribution / measure as a special probability distribution $P_a \in \mathcal{D}(A)$ or probability measure μ_a for every $a \in A$ such that (cp. [Kle06, Example 1.30] or [CD14])

$$P(\omega) = \begin{cases} 1 & \text{if } \omega = a, \\ 0 & \text{otherwise,} \end{cases} \quad \text{or } \mu(B) = \begin{cases} 1 & \text{if } a \in B, \\ 0 & \text{otherwise.} \end{cases}$$

In order to obtain a probability measure on a σ-algebra it suffices to declare the measure for a sufficiently rich family of elements which induce the measure on the complete σ-algebra. We quote for this

Theorem 1 (Carathéodory's Extension Theorem)**.** *[Bau92, Theorem 2.4, Theorem 5.6] For a ground-set Ω and every collection $\mathcal{E} \subseteq \mathrm{Pot}(\Omega)$ that is closed under intersection and which contains $\emptyset$ and a sequence $(E_i)_{i\in\mathbb{N}}$ such that $\cup_{i\in\mathbb{N}} E_i = \Omega$ any σ-additiv function $\mu' : \mathcal{E} \to [0,1]$ with $\mu'(\emptyset) = 0$ and $\mu'(\Omega) = 1$ can be* uniquely *extended to a probability measure μ on the smallest σ-algebra containing $\mathcal{E}$ (denoted by $\sigma(\mathcal{E})$). This entails a probability space*

$$(\Omega, \sigma(\mathcal{E}), \mu)\,.$$

The following theorem is used to formalize the intuition that sufficiently probable events happen again and again if an infinite amount of time passes.

Theorem 2 (Borel-Cantelli Lemma)**.** *[Kle06, Theorem 2.7][ER59] For a probability space $(\Omega, \mathcal{F}, \mu)$ and a sequence $(A_i)_{i>0}$ with $A_i \in \mathcal{F}$ and*

$$\mathcal{A} = \bigcap_{i>0} \bigcup_{j>m} A_j$$

such that $\mu(A_i \cap A_j) = \mu(A_i) \cdot \mu(A_j)$ for all $i \neq j$ and $i, j > 0$ it holds that $\mu(\mathcal{A}) = 1$ if

$$\sum_{i>0} \mu(A_i) = \infty.$$

The property that $\mu(A_i \cap A_j) = \mu(A_i) \cdot \mu(A_j)$ for all $i \neq j$ and $i, j > 0$ is called pairwise independence *of all A_i for $i > 0$.*

The following notions are standard for systems that show probabilistic movements through a finite state space (cp. [Cha+15; Ras+07]). We introduce cylindric sets over words. Consider a finite alphabet A and for any $u \in A^*$ we introduce the cylinder of u by all infinite words in A^ω that respect u as prefix. Formally, we get

$$\mathrm{cyl}(u) = u \cdot A^\omega.$$

Based upon these cylindric sets we construct the smallest σ-algebra containing all these sets as *Borel*-algebra (denoted by $\mathcal{B}(A)$). We can obtain $\mathcal{B}(A)$ by including all cylindric sets and consider the transitive closure under negation and countably union of this set. This is defined as

Definition 4. ***Borel-algebra***:
For a finite set A we call $\mathcal{B}(A) \subseteq \mathrm{Pot}(A^\omega)$ the smallest σ-algebra containing $\mathrm{cyl}(w)$ for all $w \in A^*$. We call a set $C \subseteq A^\omega$ *Borel* if $C \in \mathcal{B}(A)$.

Note that the collection of all cylindric sets $\mathcal{E}$ over a finite set A together with $\emptyset$ satisfy the conditions for the family in Theorem 1.

Additionally, we discuss measurable and integrable functions with helpful related results [Bau92, Chapter 7].

Definition 5. ***Measurable Function***:
For two measurable spaces $(\Omega_1, \mathcal{F}_1)$, $(\Omega_2, \mathcal{F}_2)$ we call a function $f : \Omega_1 \to \Omega_2$ $\mathcal{F}_1$-$\mathcal{F}_2$ measurable if

$$f^{-1}(A) = \{\omega \in \Omega_1 \mid f(\omega) \in A\} \in \mathcal{F}_1 \text{ for all } A \in \mathcal{F}_2.$$

We consider particularly measurable function from a measurement space $(\Omega, \mathcal{F})$ to the measurment space $([0,1], \mathcal{A})$ where $\mathcal{A}$ is the smallest σ-algebra which contains all intervals $[0, a)$ for any $a \in [0, 1]$ (cp. [Bau92, Chapter 4, Chapter 6]). These functions are called numerical functions and we quote the following helpful

Theorem 3. *[Bau92, Theorem 9.2] For a measurable space $(\Omega, \mathcal{F})$ and a numerical function $f : \Omega \to [0, 1]$ the $\mathcal{F}$-measurability of f is equivalent to one of the following conditions*

1. *for all $a \in [0, 1]$ holds $\{p \in \Omega \mid f(p) < a\} \in \mathcal{F}$,*
2. *for all $a \in [0, 1]$ holds $\{p \in \Omega \mid f(p) \leq a\} \in \mathcal{F}$,*
3. *for all $a \in [0, 1]$ holds $\{p \in \Omega \mid f(p) > a\} \in \mathcal{F}$,*
4. *for all $a \in [0, 1]$ holds $\{p \in \Omega \mid f(p) \geq a\} \in \mathcal{F}$.*

Moreover, we obtain for all numerical functions the following

Theorem 4. *[Bau92, Theorem 11.6] For a probability space $(\Omega, \mathcal{F}, \mu)$ and a numerical $\mathcal{F}$-measurable function $f : \Omega \to [0,1]$ the integral*

$$\int_{\omega \in \Omega} f(\omega) d\mu(\omega)$$

is well-defined in the usual sense.

And additionally

Lemma 2. *[CHS14, Lemma 40] Let $(\Omega, \mathcal{F}, \mu)$ be a probability space and f a measureable function from Ω to $[0,1]$, then $\int_\Omega f d\mu = 1$ if and only if $\mu(f^{-1}(1)) = 1$,*

We occasionally use product spaces which are defined as follows [Bau92, Chapter 22]:

Definition 6. *Product Space*:
For two measurable spaces $(\Omega_1, \mathcal{F}_1)$ and $(\Omega_2, \mathcal{F}_2)$ we fix

$$\Omega = \Omega_1 \times \Omega_2$$

and consider

$$p_i : \Omega \to \Omega_i \text{ with } p_i((\omega_1, \omega_2)) = \omega_i \text{ for } i = 1, 2.$$

The smalles σ-algebra $\mathcal{F}$ such that p_1, p_2 are $\mathcal{F}$-$\mathcal{F}_1$, $\mathcal{F}$-$\mathcal{F}_2$ measurable respectively is called product of the σ-algebras $\mathcal{F}_1$ and $\mathcal{F}_2$ denoted by

$$\mathcal{F} = \mathcal{F}_1 \otimes \mathcal{F}_2.$$

We obtain for product algebras that it is sufficient to examine generating families of sets, namely

Theorem 5. *[Bau92, Theorem 22.1] Given two collections $\mathcal{E}_1$ and $\mathcal{E}_2$ which generate σ-algebras $\mathcal{F}_1$ and $\mathcal{F}_2$ for ground-sets Ω_1 and Ω_2 respectively such that there are sequences $\left(E_j^i\right)_{j \in \mathbb{N}}$ respectively with $E_j^i \in \mathcal{E}_i$ for $i = 1, 2$ and $j \in \mathbb{N}$ and $\Omega_i = \cup_{j \in \mathbb{N}} E_j^i$. Then the product σ-algebra $\mathcal{F}_1 \otimes \mathcal{F}_2$ is generated by $E_1 \times E_2$ for all pairs $E_1 \in \mathcal{E}_1$ and $E_2 \in \mathcal{E}_2$.*

Analogously to [CHS14, Remark 35], we consider product algebras of two Borel-algebras and observe the following

Lemma 3. *For two finite sets A, B the product algebra $\mathcal{A} \otimes \mathcal{B}$ is generated by the sets*

$$\mathrm{cyl}(u) \times \mathrm{cyl}(v) \textit{ for } u \in A^n, v \in B^n \textit{ for all } n > 0.$$

Proof. From Theorem 5 we know that $\mathcal{B}(A) \otimes \mathcal{B}(B)$ is generated by $\mathrm{cyl}(u) \times \mathrm{cyl}(v)$ for all $u \in A^*$ and $v \in B^*$. W.l.o.g. we assume $|u| < |v|$. From $m = |v| - |u|$ we generate the set $E = \{u \cdot y : y \in A^m\}$. Since A is finite so is E and we may use the generating sets $\mathrm{cyl}(u') \times \mathrm{cyl}(v)$ for all $u' \in E$ to obtain the same set as $\mathrm{cyl}(u) \times \mathrm{cyl}(v)$. Therefore, all sets $\mathrm{cyl}(u) \times \mathrm{cyl}(v)$ are part of algebra which is generated by balanced cylinders. By construction and minimality of Borel-algebras the claim follows. □

Moreover regarding probability measures in product spaces we have

Lemma 4. *[Bau92, Lemma 23.2, Theorem 23.3] For two probability spaces*

$$(\Omega_1, \mathcal{F}_1, \mu_1) \text{ and } (\Omega_2, \mathcal{F}_2, \mu_2)$$

holds that for every $Q \in \mathcal{F}_1 \otimes \mathcal{F}_2$ the functions

$$\omega_i \mapsto \mu_{3-i}(\{\omega_{3-i} \in \Omega_{3-i} \mid (\omega_i, \omega_{3-i}) \in Q\}) \text{ for } i = 1, 2$$

defined on Ω_i and measurable in $\mathcal{F}_i$ for $i = 1, 2$ respectively. Moreover, there is a unique measure π (denoted by $\mu_1 \otimes \mu_2$) for the measurable space $(\Omega_1 \times \Omega_2, \mathcal{F}_1 \otimes \mathcal{F}_2)$ with

$$\pi(A_1 \times A_2) = \mu_1(A_1) \cdot \mu_2(A_2) \text{ for all } A_1 \in \mathcal{F}_1, A_2 \in \mathcal{F}_2.$$

Also, for these product spaces and measurable functions we have

Theorem 6 (Tonelli's Theorem)**.** *[Bau92, Theorem 23.6] For two probability spaces $(\Sigma_i, \mathcal{F}_i, \mu_i)$ $(i = 1, 2)$ and a $\mathcal{F}_1 \otimes \mathcal{F}_2$-measureable numerical function $f : \Sigma_1 \times \Sigma_2 \to [0, 1]$ then we obtain measurability in $\mathcal{A}_1$ and $\mathcal{A}_2$ respectively of*

$$\omega_1 \mapsto \int_{\Omega_2} f(\omega_1, \cdot) d\mu_2 \text{ and } \omega_2 \mapsto \int_{\Omega_1} f(\cdot, \omega_2) d\mu_1.$$

Additionally, it holds that

$$\int f d(\mu_1 \otimes \mu_2) = \int_{\omega_1 \in \Omega_1} (\int_{\Omega_2} f(\omega_1, \cdot) d\mu_2) d\mu_1 = \int_{\omega_2 \in \Omega_2} (\int_{\Omega_1} f(\cdot, \omega_2) d\mu_1) d\mu_2.$$

Moving away from measurable functions we consider a more involved way to relay two measurable spaces in the means of [Kle06, Definition 8.25]

Definition 7. ***Markov-Kernel***:
For two measurable spaces $(\Omega_1, \mathcal{F}_1)$ and $(\Omega_2, \mathcal{F}_2)$ a function

$$K : \Omega_1 \times \mathcal{F}_2 \to [0, 1]$$

is called a Markov-kernel if

1. $K(\cdot, A)$ is measurable in $\mathcal{F}_1$ for all $A \in \mathcal{F}_2$,

2. $K(\omega, \cdot)$ is a probability measure on $(\Omega_2, \mathcal{F}_2)$ for every $\omega \in \Omega_1$.

Moreover, we have

Lemma 5. *[Kle06, Remark 8.26] Given two measurable spaces $(\Omega_1, \mathcal{F}_1)$, $(\Omega_2, \mathcal{F}_2)$ and a collection $\mathcal{E}$ that is closed under intersection and contains a sequence of sets $(E_i)_{i\in\mathbb{N}}$ such that $\cup_{i\in\mathbb{N}} E_i = \Omega_2$. A function*

$$K : \Omega_1 \times \mathcal{F}_2 \to [0, 1]$$

is a Markov-kernel if

1. *$K(\cdot, E)$ is measurable in $\mathcal{F}_1$ for all $E \in \mathcal{E}$,*
2. *$K(\omega, \cdot)$ is a probability measure on $(\Omega_2, \mathcal{F}_2)$ for every $\omega \in \Omega_1$.*

Similar to Lemma 4 we have

Theorem 7. *[Kle06, Corollary 14.23] For a probability space $(\Omega_1, \mathcal{F}_1, \mu)$, a measurable space $(\Omega_2, \mathcal{F}_2)$ and a Markov-kernel $K : \Omega_1 \times \mathcal{F}_2$ there exists a unique probability measure $\mu \otimes K$ on $(\Omega_1 \times \Omega_2, \mathcal{F}_1 \otimes \mathcal{F}_2)$ with*

$$\mu \otimes K(A_1, A_2) = \int_{\omega\in A_1} K(\omega, A_2) d\mu \text{ for all } A_1 \in \mathcal{F}_1, A_2 \in \mathcal{F}_2.$$

A stronger connection between probability spaces are isomorphisms. We define analogously to [Kle06, Definition 8.34]

Definition 8.
We call two probability spaces $(\Omega_1, \mathcal{F}_1, \mu_1)$ and $(\Omega_2, \mathcal{F}_2, \mu_2)$ isomorphic if there is a bijective function $\rho : \Omega_1 \to \Omega_2$ such that ρ is $\mathcal{F}_1$-$\mathcal{F}_2$ measurable and ρ^{-1} is $\mathcal{F}_2$-$\mathcal{F}_1$ measurable and $\mu_2 = \mu_1 \circ \rho^{-1}$.

We introduce a certain class of isomorphic probability spaces which we occasionally make use of: for two finite non-empty sets A, B and a function $f : A \to B$, we introduce the operator

$$\mathrm{lift}_f : A^\omega \to (A \times B)^\omega \text{ with } \mathrm{lift}_f(\alpha_1\alpha_2 \dots) = (\alpha_1, f(\alpha_1))\,(\alpha_2, f(\alpha_2)) \dots.$$

We infer

Lemma 6. *Given finite and non-empty A, B and a function $f : A \to B$ then the probability space $(A^\omega, \mathcal{B}(A), \mu)$ is isomorphic to*

$$\left(\mathrm{lift}_f(A^\omega), \mathcal{B}(A \times B)_{|_{\mathrm{lift}_f(A^\omega)}}, \mu'\right) \text{ with } \mu' = \mu \circ \mathrm{lift}_f^{-1},$$

for the isomorphism lift_f.

Moreover, $((A \times B)^\omega, \mathcal{B}(A \times B), \mu')$ *forms a probability space itself if we set*

$$\mu'(A) = 0 \text{ for all } A \in (A \times B)^\omega \setminus \mathrm{lift}_f(A^\omega).$$

Proof. At first, we show that lift_f and lift_f^{-1} are $\mathcal{B}(A)$-$\mathcal{B}(A \times B)_{|_{\mathrm{lift}_f(A^\omega)}}$ measurable and $\mathcal{B}(A \times B)_{|_{\mathrm{lift}_f(A^\omega)}}$-$\mathcal{B}(A)$ measurable respectively. However, this is an immediate consequence of the observation that for every $a_1 \dots a_n \in A^*$ holds

$$\mathrm{lift}_f(\mathrm{cyl}(a_1 \dots a_n)) = \mathrm{cyl}((a_1, f(a_1)) \dots (a_n, f(a_n))) \cap \mathrm{lift}_f(A^\omega)$$

and that $\mathcal{B}(A \times B)_{|_{\mathrm{lift}_f(A^\omega)}}$ is generated by

$$\mathrm{cyl}((a_1, f(a_1)) \dots (a_n, f(a_n))) \cap \mathrm{lift}_f(A^\omega) \text{ for all } a_1 \dots a_n \in A^*.$$

Hence the generating sets of both algebras are bijected by lift_f. This implies the measurability of lift_f and lift_f^{-1} by construction of the respective σ-algebras. Hence, the isomorphism follows by definition of μ'.

By the measurability of lift_f we obtain that $\mathrm{lift}_f(A^\omega) \in \mathcal{B}(A \times B)_{|_{\mathrm{lift}_f(A^\omega)}}$. Therefore, by the closure properties of the σ-algebra we have $\mathcal{B}(A \times B)_{|_{\mathrm{lift}_f(A^\omega)}} \subseteq \mathcal{B}(A \times B)$. Hence, extending μ' as proposed in the claim indeed yields a probability measure on $((A \times B)^\omega, \mathcal{B}(A \times B), \mu')$. □

2. Word Automata

In this chapter we discuss automata for infinite words. We present results form the well-established theory of ω-regular languages and introduce the lesser known concept of Probabilistic Büchi Automata (PBAs).

2.1. ω-regular Languages

Following [Tho97] (respectively [GTW02, Chapter 1] for the Parity-condition below) we introduce word automata with the following structural definition

Definition 9. ***Word Automaton***:
We define a word automaton as $\mathcal{A} = (Q, \Sigma, q_0, \Delta)$ where Q is a set of states, Σ a finite alphabet, $q_0 \in Q$ is the initial state and $\Delta \subseteq Q \times \Sigma \times Q$ is the transition relation. It is required that for every $q \in Q$ and $\sigma \in \Sigma$ exists one $p \in Q$ such that $(q, \sigma, p) \in \Delta$. We define some associated notions as follows:

Run For a word $\alpha = \alpha_0\alpha_1 \cdots \in \Sigma^\omega$ we call a sequence $\pi = \pi_0\pi_1 \cdots \in Q^\omega$ a run of $\mathcal{A}$ on α if $\pi_0 = q_0$ and for every $i \in \mathbb{N}$ holds that $(\pi_i, \alpha_i, \pi_{i+1}) \in \Delta$.

Determinism We call $\mathcal{A}$ deterministic if for every pair $q \in Q$ and $\sigma \in \Sigma$ the set $\{p \in Q \mid (q, \sigma, p) \in \Delta\}$ has exactly one element. In that case we occasionally substitute the relation Δ with a function $\delta : Q \times \Sigma \to Q$.

This allows for automata to yield for an input word a corresponding run, i.e. attach onto the input word a word of states that is consistent with the structure of the automaton. These attached words of states are runs of word automata on words and are categorized as accepting or non-accepting by the following conditions:

Definition 10. ***Acceptance Conditions***:
For a word automaton $\mathcal{A} = (Q, \Sigma, q_0, \Delta)$ we define different acceptance conditions. For this we define for a run $\pi = \pi_0\pi_1 \cdots \in Q^\omega$ the operator Inf as the set of states that occur infinitely in a run, i.e.

$$\mathrm{Inf}(\pi) = \{q \in Q | \text{ there are infinitely many } i \in \mathbb{N} \text{ s.t. } \pi_i = q\}.$$

With this notion we define the following acceptance conditions:

Büchi A Büchi-condition is defined by a set of final states $F \subseteq Q$ and we call a run π accepting if $\mathrm{Inf}(\pi) \cap F \neq \emptyset$.

Muller Muller-conditions are given as a family of state sets, i.e. $\mathcal{F} \subseteq 2^F$. A run π is called accepting if $\mathrm{Inf}(\pi) \in \mathcal{F}$.

Rabin This acceptance condition is represented by a set of pairs $\Omega = \{(E_0, F_0), \ldots, (F_n, E_n)\}$ and we call a run π accepting if there is an i s.t. $\mathrm{Inf}(\pi) \cap F_i \neq \emptyset$ but $\mathrm{Inf}(\pi) \cap E_i = \emptyset$.

Parity This condition is defined by a function $\mathrm{par} : Q \to \mathbb{N}$. We call a run π accepting if the maximum of the set $\mathrm{par}(\mathrm{Inf}(\pi))$ is even (note that due to the finiteness of Q, $\mathrm{Inf}(\pi)$ is finite as well and thus the maximum of $\mathrm{par}(\mathrm{Inf}(\pi))$ exists).

For any of these objects $\Gamma \in \{F, \mathcal{F}, \{(E_0, F_0), \ldots, (E_n, F_n)\}, \mathrm{par}\}$ we use $\mathrm{Acc}(\Gamma)$ to refer to the accepted language. If necessary we use subscripts to Acc to make the interpretation of its argument clear.

We categorize word automata by their acceptance condition and if they are deterministic. For example we call a word automaton $\mathcal{A}$ equipped with a Büchi-condition a Non-Deterministic Büchi Automaton (NBA) or respectively Deterministic Büchi Automaton (DBA) if $\mathcal{A}$ is deterministic. And analogously we obtain word automata for Muller-, Rabin- and Parity-conditions as Non-Deterministic Muller Automaton (NMA), Deterministic Muller Automaton (DMA), Non-Deterministic Rabin Automaton (NRA), Deterministic Rabin Automaton (DRA), Non-Deterministic Parity Automaton (NPA) and Deterministic Parity Automaton (DPA).

The theory to these word automata, which we call ω-regular automata, is well established. In the following we recall some of the central results and selectively provide proofs. Our arguments follow, in some condensed and occasionally less formal form, [GTW02, Chapter 1]. We start with

Example 1. We consider the language $\mathcal{L}$ as the set of all infinite words over the alphabet $\{a, b\}$ such that a occurs only finitely often. This language can be accepted using e.g. an NBA $\mathcal{A}$ as defined in Figure 1. The argument that this NBA accepts $\mathcal{L}$ is as follows: The non-determinism allows the automaton to "guess" one moment from which on no more a is read. For any word in $\mathcal{L}$ such a moment exists and can therefore be correctly guessed while every word that is accepted by $\mathcal{A}$ needs to move to its q_F state and it is straightforward to argue that no more a can occur afterwards.

Alternatively, we can accept $\mathcal{L}$ by a DPA $\mathcal{B}_P$, DRA $\mathcal{B}_R$, DMA $\mathcal{B}_M$ which all share the same structure as defined in Figure 1. We use the different acceptance conditions to model the same restriction, namely that q_a only occurs finitely often while q_b occurs infinitely often.

$\mathcal{B}_P$ We define $\text{par}(q_a) = 1$ and $\text{par}(q_b) = 0$, thus - since the parity of q_a trumps the parity of q_b but is odd - q_a must only occur finitely often.

$\mathcal{B}_R$ Here we define $\{(E = \{q_a\}, F = \{q_b\})\}$, which models the restriction very clearly by explicitly stating which states are "good" respectively "bad" to visit infinitely often.

$\mathcal{B}_M$ The acceptance family is fixed with $\mathcal{F} = \{\{q_b\}\}$, thus the only way to achieve an accepting run is to eventually stay in q_b.

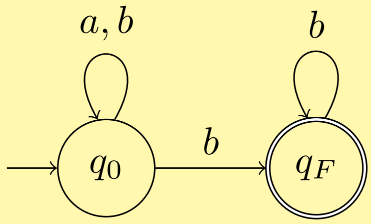
It is noteworthy that all these conditions are essentially modelling the abscence of a state. This is occasionally done by defining a set of states that must only be visited finitely often. Such an acceptance condition is called a *co-Büchi-condition* with $F = \{q_a\}$ and describes the inverse of a Büchi-condition; that is the lack of occurences of elements in F from one point onwards.

Note that on the other hand we used a structural way to model this negative restriction for $\mathcal{A}$.

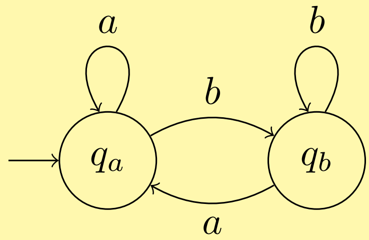
We radically compress the rich theory of ω-automata by stating the following

Figure 1. In (a) an NBA is illustrated which accepts the language of words with finitely many a. The states of the Büchi-condition F are marked by doubling the outline, i.e. $F = \{q_F\}$. In (b) a deterministic automaton is defined which starting in q_a simply moves to q_a (q_b) if an a (b) is read.

(a)



(b)



Theorem 8. *[GTW02, Theorem 1.19, Theorem 1.24, Section 1.3.2, Theorem 3.2] The class of recognizable languages coincides for NBAs, NMAs, DMAs, NRAs, DRAs, NPAs and DPAs and is called ω-regular languages. DBAs are strictly less expressive.*

In the following we present parts of the proof of this theorem. Mainly, we introduce and discuss common concepts and arguments for automata which are also valuable in more involved contexts.

Firstly, we provide an argument to show that DBAs are less expressive. Consider again $\mathcal{L}$ from Example 1 and the following argument: the deterministic nature of any DBA $\mathcal{A}$

yields one possible run for every word of $\mathcal{A}$. This means that words with common finite prefixes share common finite prefixes of their individual runs. Assume to have one DBA $\mathcal{A}$ which accepts $\mathcal{L}$ and consider the following family of words with common prefixes of increasing length

$$b^\omega, b^{n_0}ab^\omega, b^{n_0}ab^{n_1}ab^\omega, \dots$$

where every n_i is chosen such that the unique run in $\mathcal{A}$ just visited a state in F (which needs to happen since every element of this word-family is in $\mathcal{L}$ and therefore accepted by $\mathcal{A}$). Thus, by iterating "sliding in" a after a visit in F we construct a word which is accepted but contains infinitely many a contradicting that $\mathcal{A}$ actually recognizes $\mathcal{L}$.

Secondly, we argue that Muller-, Rabin- and Parity-conditions are equally expressive. Therefore, we initially consider that all deterministic automata are non-deterministic automata which do not use their non-determinism. Furthermore it is easy to see, that Rabin- and Parity-conditions are captured by Muller-conditions by enumerating all Inf-sets that satisfy the corresponding conditions. In a second step we show that we can translate NMAs to NPAs as well as NRAs. This translation makes use of the following construct, called Latest Appearance Record (LAR):

Definition 11. ***Latest Appearance Record***:
For a finite state set Q define the set of permutations of elements in Q as $\mathrm{Perm}(Q) \subseteq Q^{|Q|}$ and

$$\mathrm{LAR}(Q) = \{[w, h] \mid w \in \mathrm{Perm}(Q) \text{ and } 1 \leq h \leq |Q|\}$$

Additionally we define an update function $\mathrm{update} : \mathrm{LAR}(Q) \times Q \to \mathrm{LAR}(Q)$ with

$$\mathrm{update}([q_1 \dots q_n, h], q) = [qq_1 \dots q_{i-1}q_{i+1} \dots q_n, i] \text{ for the unique } i \text{ s.t. } q = q_i$$

and conclude with auxillary definitions of

Hit-set for $\ell = [q_1 \dots q_n, h]$ we call $\{q_1, \dots, q_h\}$ the hit-set of ℓ,

State partition the states of $\mathrm{LAR}(Q)$ can be partitioned into those elements that resulted from an update by $\mathrm{update}(\cdot, q)$ for every $q \in Q$. We define $\mathrm{LAR}(Q)[q] = \{[q_1 \dots q_n] \in \mathrm{LAR}(Q) \mid q_1 = q\} = \{\mathrm{update}(\ell, q) : \ell \in \mathrm{LAR}(Q)\}$.

For a word $q_0q_1q_2 \dots \in Q^\omega$ we can obtain one associated sequence of elements in $\mathrm{LAR}(Q)$ by picking one arbitrary starting value $\ell_0 \in \mathrm{LAR}(Q)[q_0]$ and successively applying the update operation, i.e. $\ell_0\ell_1\ell_2 \dots \in \mathrm{LAR}(Q)^\omega$ with $\ell_{i+1} = \mathrm{update}(\ell_i, q_{i+1})$ for all $i \geq 0$. We can now state that $\mathrm{Inf}(q_0q_1q_2 \dots)$ moves to the beginning of the $\mathrm{LAR}(Q)$ elements in the associated sequence, formalized in

Lemma 7. *[GTW02, Lemma 1.21]* $\mathrm{Inf}(q_0q_1q_2 \dots) = F$ *if and only if from one* $k > 0$ *the hit-set of all* ℓ_i *for* $i > k$ *is a subset of* F *and there are infinitely many* $i > k$ *s.t. the hit-set of* ℓ_i *coincides with* F.

This Lemma induces that the sequence of hit-sets stabilises to the set $\mathrm{Inf}(q_1q_2 \dots)$. Thus, $q_1q_2 \dots$ satisfies a Muller-condition $\mathcal{F}$ if and only if the hit-sets stabilises to an element of $\mathcal{F}$. Since this implies that the Inf-set is the biggest occuring hit-set from one point onwards, this can be captured within a Parity-condition by associating to $\ell = [p_1 \dots p_n, h]$ with hit-set $H = \{p_1, \dots, p_h\}$ a Parity-condition

$$\mathrm{par}(\ell) = \begin{cases} 2 \cdot |H| & \text{if } H \in \mathcal{F}, \\ 2 \cdot |H| - 1 & \text{if } H \notin \mathcal{F}. \end{cases}$$

Thus, winning (losing) hit-sets do have an even (odd) parity and notably the highest parity value is even if and only if $\mathrm{Inf}(q_1 q_2 \dots) \in \mathcal{F}$. Therefore, we can define a NPA that has as state set the elements $\mathrm{LAR}(Q)$ and mirrors the transitions of the original NMA through the application of the update function, i.e. for every (q, a, p) there is a corresponding $(\ell_q, a, \mathrm{update}(\ell_q, p))$ for every $\ell_q \in \mathrm{LAR}(Q)$ that is produced by $\mathrm{update}(\cdot, q)$. Every run in the NPA can be translated back to a run in the original NMA and vice versa and by Lemma 7 and the definition of par we obtain the acceptance equivalence of both runs. We close this argument by stating that any Parity-condition can be expressed by a Rabin-condition as follows: we define for every occuring even parity $k \in \mathrm{par}(Q)$ a pair (E_k, F_k) where $F_k = \mathrm{par}^{-1}(k)$ and $E_k = \cup_{p>k} \mathrm{par}^{-1}(p)$; hence, an accepting run regarding the Parity-condition satisfies the pair associated with the highest parity and on the other hand, any run that is accepting by the Rabin-condition, namely by the pair (E_k, F_k), has k as the highest parity which is the even witness rendering the run accepting.

Thirdly, we conclude by stating (without proof) two results. Firstly, that there is for every NMAs an equivalent NBA and the other way around (this is actually easy since Muller-conditions as well as Rabin- and Parity-conditions can emulate Büchi-conditions):

Theorem 9. *[GTW02, Theorem 1.10] For every NBAs exists an equivalent NMA and for every NMAs exists an equivalent NBA.*

Secondly, we also avoid the lengthy technicalities of determinization of an NBA by simply stating

Theorem 10. *[GTW02, Theorem 3.6] [Pit07, Section 3.2] [Löd99, Theorem 6] For every NBAs $\mathcal{A}$ exists an equivalent DPA $\mathcal{P}$.*

Given that $\mathcal{A}$ has n states then $\mathcal{P}$ can be constructed with $n^{2 \cdot n+2}$ states. There are $\mathcal{A}$ such that $\mathcal{P}$ necessarily has states in the magnitude of $n!$.

Intuitively, we observe that determinism is only a real restriction for Büchi-conditions, but that constructing deterministic automata can be very costly.

Additionally, we state regarding the closure properties of ω-regular languages the following

Theorem 11. *[GTW02, Consequence from Theorem 1.5 and Theorem 1.24] The ω-regular languages form a Boolean-algebra, i.e. are closed under union, intersection and negation. The transformations can be effectively constructed.*

Proof. For this proof we give operations that take two (respectively one) word automata and construct a word automaton that accepts the union or intersection (or respectively the negation). Beginning with the union, we make use of non-determinism such that the resulting automaton guesses at the beginning in which language the word is and simulates the "correct" automaton. Hence, we use a copy of both automata and introduce a new distinct initial state q^0. q^0 is equipped with every transition from the two original inital states. Therefore, the initial decision determines which automaton is run on the input word. If the word is part of the language of one of these automata an omniscient oracle is able to provide an accepting run by moving into the correct automaton in the beginning. If the word is not element of either language any initial guess is doomed to produce a non-accepting run.

The closure under intersection is obtained by a natural product construction, namely

Definition 12.
For $\mathcal{A}_1 = (Q, \Sigma, q_0, \Delta)$ and $\mathcal{A}_2 = (P, \Sigma, p_0, \nabla)$ we define

$$\mathcal{A}_1 \otimes \mathcal{A}_2 = (Q \times P, \Sigma, (q_0, p_0), \Delta \otimes \nabla)$$

with

$$\Delta \otimes \nabla = \left\{ \left(\left(q^1, p^1\right), \sigma, \left(q^2, p^2\right) \right) : \left(q^1, \sigma, q^2\right) \in \Delta, \left(p^1, \sigma, p^2\right) \in \nabla \right\}.$$

Every run in this product can be separated into runs in $\mathcal{A}_1$ and $\mathcal{A}_2$. W.l.o.g. we can assume the original two automata to be equipped with Muller-conditions $\mathcal{F}_1, \mathcal{F}_2$ and define for the product automaton the Muller-condition

$$\mathcal{F} = \left\{ \left\{ \left(q^1, p^1\right), \ldots, \left(q^n, p^n\right) \right\} : \left\{q^1, \ldots, q^n\right\} \in \mathcal{F}_1, \left\{p^1, \ldots, p^n\right\} \in \mathcal{F}_2 \right\}.$$

Hence the run in the product automaton is accepted if and only if the associated runs in the original automata are accepted.

The closure under negation can be expressed very elegantly (cp. [GTW02, Transformation 1.25.]). W.l.o.g. we consider the original automaton to be a DPA. By setting a new Parity-condition with $\mathrm{par}'(q) = \mathrm{par}(q) + 1$ for all $q \in Q$ (denoted by $\mathrm{par}' = \mathrm{par} + 1$) and keeping the original structure we exchange non-accepting and accepting runs and therefore obtain a structural equivalent DPA that precisely accepts the complement of the original automaton. $\square$

At last, we want to shortly present that deciding emptiness for ω-regular languages can be decided by means of graph algorithms. The central idea is to observe that for any NBA $\mathcal{A} = (Q, q_0, \Sigma, \Delta, F)$ and an accepting run π at least one $f \in F$ has to occur infinitely often in π. It is therefore sufficient to examine if the graph

$$(Q, E_\Delta) \text{ with } E_\Delta = \{(q, p) \in Q \times Q \mid \text{exists } \sigma \in \Sigma \text{ such that } (q, \sigma, p) \in \Delta\},$$

contains an $f \in F$ which is reachable from q_0 and f itself. Moving from q_0 to f and then loop in f yields one accepting word (see Figure 2). This allows to state

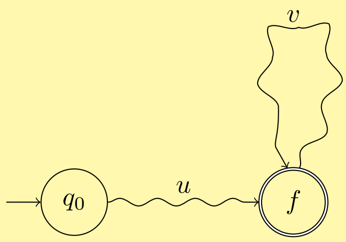
Theorem 12. *[Büc90; Tar72] Emptiness for a given NBA*

$$\mathcal{A} = (Q, q_0, \Sigma, \Delta, F)$$

can be decided.

It is possible to compute in time linear to the size of the automaton, i.e. in $\mathcal{O}(|Q|+|\Delta|)$*, if an NBA accepts a word. If so,* $\mathcal{A}$ *also accepts an* ultimatively periodic word*, i.e. a word of the form* $u \cdot v^\omega$ *for* $u, v \in \Sigma^+$*.*

Figure 2. Illustration of an accepting run of an NBA.



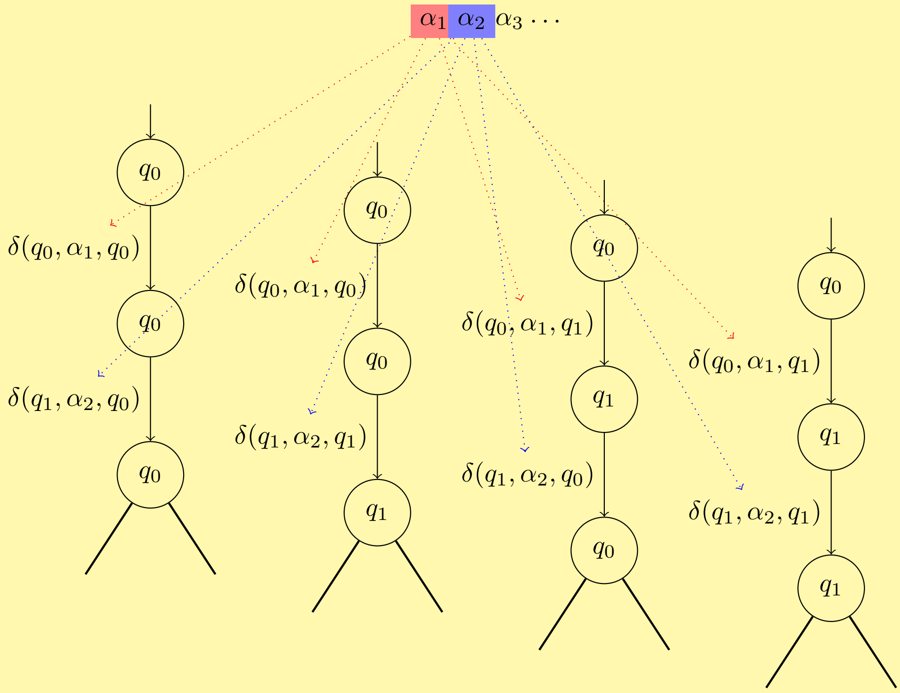
2.2. Probabilistic Büchi Automata

Replacing non-deterministic choices by probabilistic ones in NBAs yields a new class of word automata, namely Probabilistic Büchi Automata (cp. [BG05; BBG08; Grö08]). A PBA essentially is an NBA where the non-deterministic oracle acts probabilistically. Therefore, we define analogously to [Grö08]

Definition 13. ***Probabilistic Büchi Automata***:
A PBA $\mathcal{A}$ over a finite alphabet Σ is defined by a tuple $(Q, \Sigma, \delta, q_0, F)$ where Q is a finite state set, $q_0 \in Q$ the initial state, $\delta : Q \times \Sigma \times Q \to [0, 1]$ a transition probability function such that for all pairs $q \in Q$ and $\sigma \in \Sigma$ we have $\sum_{p \in Q} \delta(q, \sigma, p) = 1$ and $F \subseteq Q$ is the set of final states.

Conceptually, a PBA $\mathcal{P}$ processes an input word similarly as other word automata by sequentially reading the letters of the input word and moving along its states. Provided the current state is q and the next letter of the input word is a then $\mathcal{P}$ consults δ to determine its next state. Specifically, $\mathcal{P}$ chooses a state p with probability $\delta(q, a, p)$. Naturally, any run of $\mathcal{P}$ starts in its initial state q_0. Considering one fixed input word α the PBA $\mathcal{P}$ provides a process which produces corresponding state sequences with certain probabilities (see Figure 3 for an illustration). Its acceptance is defined by the notion of how probable the produced state sequences are to satisfy the associated Büchi-condition F. For a PBA $\mathcal{P} = (Q, q_0, \delta, F)$ we are interested in a probability space that

Figure 3. Illustration of the stochastic process of a PBA $\mathcal{P}$ with two states q_0, q_1 on a word $\alpha \in \Sigma^\omega$ with transition probabilities defined by δ.



is induced by the run of $\mathcal{P}$ on the base set Q^ω while reading the input word α. For the cylindric sets $\mathrm{cyl}(u)$ for any $u \in Q^*$ (see Figure 4 for an illustration) we obtain the probability to stay within this cylinder while reading a finite prefix of the input word. Let $v = u_0 \ldots u_n$ be such a finite prefix of α. For any sequence $q_1 q_2 \ldots q_n$ the run of $\mathcal{P}$ on α stays within $\mathrm{cyl}(q_1 \ldots q_n)$ with the probability to move along the path $q_0 q_1 \ldots q_n$ while reading $u_1 \ldots u_n$. This is the product of the probabilities to move from q_{i-1} to q_i while reading u_i for $1 \leq i \leq n$: $\prod_{1 \leq i \leq n} \delta(q_{i-1}, u_i, q_i)$.

Given an input word α and using the Borel-algebra $\mathcal{B}(Q)$ and the probability for staying in a cylinder for α allows to use Theorem 1 to fix a probability space

$$(Q^\omega, \mathcal{B}(Q), \mu_\alpha)\,.$$

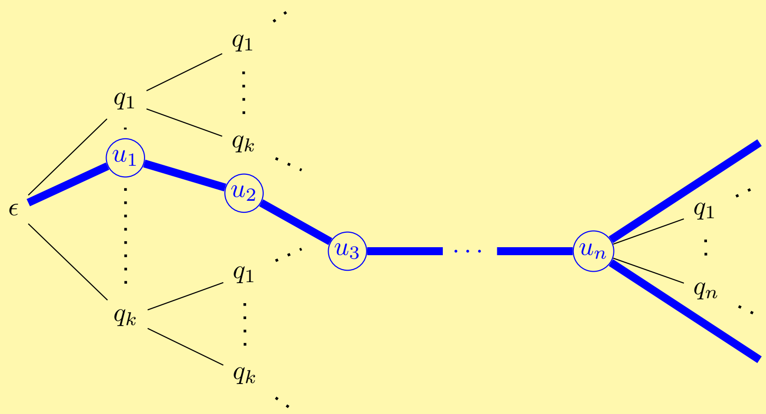
In which the acceptance of α is encoded by the probability of the set

$$\mathrm{Acc}(F) = \{\alpha \in Q^\omega \mid \alpha \text{ satisfies the Büchi-condition } F\}\,.$$

Therefore, it is necessary to show that this set indeed is measureable. This is known from e.g. [Grö08, Chapter 4.1.1.] but we present a proof here which follows [CHS14, Proposition 6] since this argument can be easily presented and is re-used later in a more involved context.

Lemma 8 (Measurability)**.** *The set* $\mathrm{Acc}(F)$ *for a Büchi-condition* $F \subseteq Q$ *is measurable in* $\mathcal{B}(Q)$.

Figure 4. All infinite sequences of states $\{q_1, \ldots, q_k\}$ can be organised in a tree. In this tree the set $\mathrm{cyl}(u_1 \ldots u_n)$ are all possible prolongations of the initial sequence $u_1 \ldots u_n$ as illustrated by the blue path and attached cylinder.



Proof. In order to show the measureability of $\mathrm{Acc}(F)$ we formulate it at as Boolean combination of cylinders inducing its membership in the Borel-algebra by the closure properties of a σ-algebra. Initially, we start with a co-Büchi-condition (recall Example 1), i.e. those words which eventually do not visit a target set T anymore. More precisely, such a set T defines the accepted language as

$$\mathrm{Acc}_{\text{co-Büchi}}(T) = Q^* \left(Q \setminus T\right)^\omega .$$

We claim that

$$\mathrm{Acc}_{\text{co-Büchi}}(T) = \bigcup_{u \in Q^*} \left(\mathrm{cyl}(u) \setminus \bigcup_{v \in u \cdot Q^* T} \mathrm{cyl}(v)\right) \tag{2.1}$$

which renders $\mathrm{Acc}_{\text{co-Büchi}}(T)$ measurable as countable union of measurable sets. For $\beta \in \mathrm{Acc}_{\text{co-Büchi}}(T)$ we know that there is eventually no occurence of T anymore. This induces that there is one finite prefix $u \sqsubseteq \beta$ such that any prolongation of u in β does not contain a state of T, hence it holds that $\beta \notin \bigcup_{v \in u \cdot Q^* T} \mathrm{cyl}(v)$. Consequently, β is element of the right hand side of Equation 2.1. On the other hand if $\beta \notin \mathrm{Acc}_{\text{co-Büchi}}(T)$ we know that for every finite prefix $u \sqsubseteq \beta$ there is a (finite) prolongation $u \sqsubseteq v \sqsubseteq \beta$ such that $v \in u \cdot Q^* T$. This v removes β from $\mathrm{cyl}(u)$. This gives that β is not element of the right hand side of Equation 2.1 which proves the announced equality. For any Büchi-condition $F \subseteq Q$ we use its duality to the co-Büchi-condition $T = Q \setminus F$, i.e.

$$\mathrm{Acc}_{\text{Büchi}}(F) = Q^\omega \setminus \mathrm{Acc}_{\text{co-Büchi}}(T),$$

to obtain the measureability of $\mathrm{Acc}(F)$ by the closure of $\mathcal{B}(Q)$ under complementation and the proven measureability of $\mathrm{Acc}_{\text{co-Büchi}}(T)$. $\square$

Since we explicitly defined PBAs with Büchi-conditions this suffices to show their definition to be useful. But it is a natural escalation to consider more elaborate acceptance conditions, e.g. Muller-, Parity- or Rabin-conditions. We show the measureability of those sets here as well, since it is a easy consequence of Lemma 8 (cp. again [CHS14, Proposition 6]).

Corollary 1. *For a Muller-condition, a Parity-condition and a Rabin-condition defined by* $\mathcal{F} \subseteq \mathrm{Pot}(Q)$, par, $R = \{(E_0, F_0), \ldots, (E_n, F_n)\}$ *respectively, the sets* $\mathrm{Acc}(\mathcal{F})$, $\mathrm{Acc}(\mathrm{par})$ *and* $\mathrm{Acc}(R)$ *are measureable in* $\mathcal{B}(Q)$.

Proof. Since Rabin- and Parity-conditions can be expressed by fitting Muller-conditions it suffices to show the measureability of any $\mathrm{Acc}(\mathcal{F})$ for a Muller-condition $\mathcal{F} \subseteq \mathrm{Pot}(Q)$. Fix one Muller-condition $\mathcal{F} = \{F_1, \ldots, F_n\}$. We claim that for one $F = \{f_1, \ldots, f_k\} \in \mathcal{F}$

$$\mathrm{Acc}_{\mathrm{Muller}}(\{F\}) = \left(\bigcap_{1 \leq i \leq k} \mathrm{Acc}_{\text{Büchi}}(\{f_i\}) \right) \setminus \mathrm{Acc}_{\text{co-Büchi}}(Q \setminus F).$$

The individual Büchi-conditions on f_i for $1 \leq i \leq k$ ensure that every of these states is visited infinitely often while the co-Büchi-condition on all other states ensure that they are only visited finitely often rendering the Inf-set to be F. By simply setting

$$\mathrm{Acc}_{\mathrm{Muller}}(\mathcal{F}) = \bigcup_{F \in \mathcal{F}} (\mathrm{Acc}_{\mathrm{Muller}}(\{F\}))$$

we obtain the measureability of $\mathrm{Acc}_{\mathrm{Muller}}(\mathcal{F})$ by the closure properties of the σ-algebra $\mathcal{B}(Q)$. $\square$

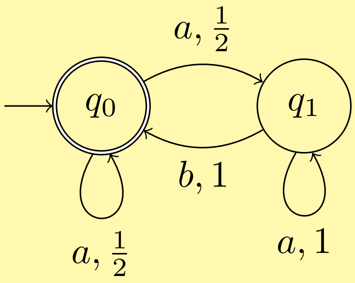
This finally allows us to define the semantics of a PBA $\mathcal{P}$ formally with

Definition 14. *Probabilistic Büchi Automaton - Acceptance*:
A PBA $\mathcal{P} = (Q, \Sigma, \delta, q_0, F)$ positively (almost-surely) accepts a word $\alpha \in \Sigma^\omega$ if $\mu_\alpha(\mathrm{Acc}(F)) > 0$ ($\mu_\alpha(\mathrm{Acc}(F)) = 1$).

With Corollary 1 we may define analogously Probabilistic Parity Automaton (PPA), Probabilistic Muller Automaton (PMA) and Probabilistic Rabin Automaton (PRA). All probabilistic word automata can be separated into positively and almost-surely accepting. We want to provide a small discussion of the theory regarding PBAs. Therefore, we present an example of a PBA which allows us to get a feel for the behavior while also providing a witness to separate the expressiveness of ω-regular languages and languages positively accepted by PBAs.

Example 2. [BG05] We examine the PBA $\mathcal{P}$ as defined in Figure 5. The accepted

Figure 5. A PBA accepting under positive acceptance a non-ω-regular language depicted by the same notions as finite word automata before. Omitted transitions (for example a b-transition from state q_0) implicitly lead to a non-accepting sink state.



language is

$$\mathcal{L}_{>0} = \left\{ a^{k_1} b a^{k_2} b \cdots : k_i > 0 \text{ for all } i > 0 \text{ and } \prod_{i>0} (1 - \left(\frac{1}{2}\right)^{k_i}) > 0 \right\}.$$

This can be explained the following way: intially $\mathcal{P}$ starts in q_0, thus the whole *probability mass* lays within q_0. Every read a dissipates half of the mass in q_0 to q_1 while the mass in q_1 stays in q_1. Every read b discards all mass in q_0 while moving all mass in q_1 to q_0. Thus, reading a gradually saves away probability mass into q_1 and b restarts this process. Note that any word ending in a^ω is not accepted since an infinite dissipatation leaves no mass in q_0 behind but only the mass visiting q_0 infinitely often is considered accepting. Thus, we need to have some probability mass which moves infinitely often through the dissipatation step but gradually, due to the loss of mass, steps need to "save" more mass in order to not let it diminish too fast.

As stated by the caption in Figure 5 the PBA $\mathcal{P}$ from Example 2 accepts a not ω-regular language. This gives

Lemma 9. *Follows from the proof of [BG05, Theorem 4]. The language*

$$\left\{ a^{k_1} b a^{k_2} b \cdots : k_i > 0 \textit{ for all } i > 0 \textit{ and } \prod_{i>0} (1 - \left(\frac{1}{2}\right)^{k_i}) > 0 \right\}$$

can be positively accepted by a PBA but is not ω-regular.

Proof. From Theorem 12 we know that every ω-regular language contains an ultimately periodic word. We argue that no ultimately periodic word is part of the language. Since any word ending in a^ω is not part of the language (as argued in Example 2) any ultimatively periodic word that might be part of the language contains at least one b in its period v. But this induces that the sequences of a are bound by a natural number k yielding that for this word

$$\prod_{i>0}(1-\left(\frac{1}{2}\right)^{k_i}) \leq \underbrace{\prod_{i>0}(1-\left(\frac{1}{2}\right)^{k})}_{\text{infinite product of a constant } 0\leq c<1} = 0.$$

Hence, every ultimatively periodic word is not accepted by $\mathcal{P}$ rendering its language not ω-regular. $\square$

Furthermore, we quote

Lemma 10. *[BG05, Lemma 5] For any NBA $\mathcal{A}$ exists a positively accepting PBA $\mathcal{P}$ which recognizes the same language.*

And therefore we may conclude that positively accepting PBAs are more expressive than ω-regular automata:

Theorem 13. *[BG05, Theorem 4] The class of languages that can be accepted by a PBA strictly contains the class of ω-regular languages.*

Additionally, we know that the class of languages that are positively accepted by a PBA forms a Boolean algebra. This is captured in

Theorem 14. *[Grö08, Chapter 4.3.] The class of languages that can be accepted by a PBA form a Boolean algebra. The transformations can be effectively constructed.*

Proof-sketch. A union operator is very easily obtained by the following idea: given two PBAs $\mathcal{P}_1, \mathcal{P}_2$ with initial states q_0^1, q_0^2 respectively, then we can construct a PBA $\mathcal{P}$ which effectively uses a initial distribution $\iota : Q^1 \cup Q^2 \to [0,1]$ such that $\iota(q_0^1) = \iota(q_0^2) = \frac{1}{2}$ while $\iota(p) = 0$ for any other state p. By introducing a new initial state q_0 and "skipping over" q_0^1, q_0^2 in the first step, i.e. setting

$$\delta(q_0, \sigma, p) = \frac{1}{2} \cdot \delta^i(q_0^i, \sigma, p) \text{ with } i = \begin{cases} 1 & \text{if } p \in Q^1, \\ 2 & \text{if } p \in Q^2 \end{cases}$$

we can emulate such an initial distribution ι. This yields that the resulting $\mathcal{P}$ accepts α with a probability $\frac{1}{2} \cdot \mu_\alpha^1(\mathrm{Acc}(F^1)) + \frac{1}{2} \cdot \mu_\alpha^2(\mathrm{Acc}(F^2))$ which is greater than 0 if and only if at least one of the probabilities is greater than 0.

Complementation requires a rather involved operation which we do not present here. Nevertheless, we mention an intermediate step: for every PBA $\mathcal{P}$ we can construct a PRA $\mathcal{R}$ such that $\mathcal{R}$ acceptes the same languages as $\mathcal{P}$. Moreover, $\mathcal{R}$ induces a binary probability measure on its acceptance condition, i.e. for its Rabin-condition R we have $\mu_\alpha(\mathrm{Acc}(R)) \in \{0,1\}$. The main idea of its construction is to sample partial runs of the original PBA and keeping track of those that are accepting. These partial runs can be then "glued together". Therefore, we state here

Theorem 15. *[Grö08, Theorem 4.3.2] For each PBA $\mathcal{P}$ there exists a PRA $\mathcal{R}$ that accepts the same language, but for every word the acceptance measure for $\mathcal{R}$ is either* 0 *or* 1*.*

Since we soon see that PBAs with almost-sure acceptance are strictly less expressive than PBAs with positive acceptance we note the correspondance to Theorem 8 in the sense that more expressive acceptance conditions allow for a "stuctural" difference, which in this case is the acceptance measure (in contrast to determinsm before). Also, we note that all "strong" conditions again coincide which is easily obtained by using LARs which we prove for a more general model in Chapter 3. Nevertheless, we state here

Theorem 16. *The expressiveness for PRA, PMA and PPA coincides for positive (almost-sure) acceptance measures.*

□

The gain in expressiveness for PBA with positive acceptance over ω-regular languages is unfortunately accompanied by some undesired effects. Notably, while emptiness for a NBA can be decided (see Theorem 12) we have that given a PBA $\mathcal{P}$ it is undecideable to find if there is any word positively accepted by $\mathcal{P}$.

Theorem 17. *[BBG08, Theorem 2] The emptiness problem for positively accepting PBA is undecideable.*

By Theorem 15 and Theorem 16 we obtain undecideability of emptiness for probabilistic word automata with almost-sure accpetance of more expressiv acceptance conditions. Moreover, since Büchi-conditions can be emulated by Muller-, Rabin- and Parity-conditions we get

Corollary 2. *The emptiness problem for PRAs, PPAs and PMAs is undecideable for positive and almost-sure acceptance.*

However, this is different for almost-sure acceptance of Büchi-conditions. Notably, we have

Theorem 18. *[BBG08, Theorem 6] The emptiness problem for PBA with almost-sure acceptance is decideable.*

Again, we consider an example for a PBA with almost-sure acceptance:

Example 3. [BBG08] Let $\mathcal{P}$ be a PBA as defined in Figure 6. The language accepted by $\mathcal{P}$ is

$$L_\lambda = \left\{ a^{k_1} b a^{k_2} b \cdots \mid k_i > 0 \text{ for } i > 0 \text{ and } \prod_{i>0} \left(1 - \lambda^{k_i}\right) = 0 \right\}.$$

Every a-sequence $w \in \{a\}^*$ models an experiment which determines how probable it is to stay in q_2, namely $\lambda^{|w|}$. Every experiment is concluded with the occurence of a single b. Naturally, the probability to "fail" such an experiment is $1 - \lambda^{|w|}$. Due to the equivalent behaviour of q_3 and q_0 all these experiments are independent and therefore, consistenly failing these experiments with a-sequences of length $k_1, k_2, \ldots$ starting from the m-th experiment happens with probability

$$p_m = \prod_{i>m}(1-\lambda^{k_i}) \text{ which implies } \prod_{i>0}(1-\lambda^{k_i}) = \underbrace{\prod_{m\geq i>0}(1-\lambda^{k_i})}_{=c_m} \cdot p_m.$$

Since $c_m > 0$ implies $c_m \cdot p_m = 0$ if and only if $p_m = 0$ we observe

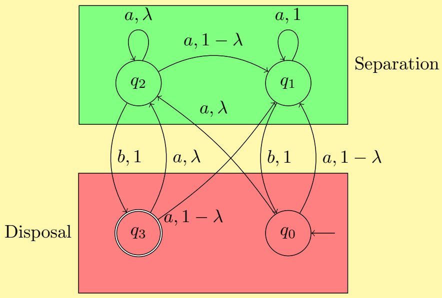
$$p_m = 0 \text{ for all } m > 0 \text{ if and only if } \prod_{i>0}(1-\lambda^{k_i}) = 0.$$

In order to accept almost-surely the set of non-accepting runs needs to be *negligible*, i.e. carrying a probability of 0. Thus, indeed $\mathcal{P}$ accepts L_λ.

If λ is set to $\frac{1}{2}$, the resulting language $\mathcal{L}_{=1}$ is - to a certain degree - dual to the language in Example 2 (without considering all words ending in a^ω or those words starting in b or ommitting an "a-phase"). This duality is rooted in structural similarities. Examining the *Separation* part, it mirrors the a-transitions of the PBA in Figure 5. But an occurences of b does not discard the probability mass in q_2 (q_0, respectively) but saves it in an accepting state and re-introduces into the circulation (as seen in the *Disposal* part).

Considering Example 3 it is a natural question if the choice of λ matters. Quoting the

Figure 6. A PBA which accepts a non-ω-regular language under almost-sure acceptance. We conceptually separate it into two regions, namely the "Separation" and "Disposal" region (highlighted by the green or red box respectively).



following

Lemma 11. *[BBG08, Lemma 1] For each $n \in \mathbb{N}_{>1}$ there exists a sequence $(k_i)_{i>0}$ such* *that*

$$\prod_{i>0}(1-\lambda^{k_i}) > 0 \textit{ if and only if } \lambda < \frac{1}{n},$$

allows us to answer this question positively. Choosing λ_1 and λ_2 differently, we can find n such that $\lambda_1 < \frac{1}{n} < \lambda_2$ and then the languages of $\mathcal{P}$ with λ_1 and λ_2 do not agree on the sequence given by Lemma 11 for n since λ_1 renders this sequence non-accepting in $\mathcal{P}$ while λ_2 renders it accepting (as described in [BBG08]).

Regarding the relationship of ω-regularity to languages which are accepted by a PBA with almost-sure acceptance we find that both classes are incomparable. We examine both directions of possible inclusions. Firstly, as noted in Figure 6 the defined PBA $\mathcal{P}$ almost-surely accepts a non-ω-regular language, hence

Lemma 12. *There is a non-ω-regular language that is accepted by any PBA with almost-sure acceptance.*

Proof. As expected, we use the language $\mathcal{L}_{=1}$ from Example 3 to witness this claim. In Example 3 we already mentioned the "almost duality" of this language to $\mathcal{L}_{>0}$ from Example 2. By defining

$$\mathcal{L}_\omega = \left\{ \alpha_1\alpha_2\cdots \in \{a,b\}^\omega \,\middle|\, \text{s.t.} \begin{array}{ll} \alpha_1 = b & \text{or} \\ \alpha_i = \alpha_{i+1} = b & \text{for one } i > 0 \text{ or} \\ \alpha_i = \alpha_{i+1} = \cdots = b & \text{for one } i > 0 \text{ holds} \end{array} \right\},$$

we can state that $\{a,b\}^\omega \setminus \mathcal{L}_{>0} = \mathcal{L}_{=1} \cup \mathcal{L}_\omega$. $\mathcal{L}_\omega$ can be separated into its individual parts where each part is ω-regular and so is $\mathcal{L}_\omega$ by closure under union for ω-regular languages. Thus, assuming $\mathcal{L}_{=1}$ ω-regularity renders $\mathcal{L}_{=1} \cup \mathcal{L}_\omega$ ω-regular and by closure under complement of ω-regular languages also $\mathcal{L}_{>0}$ which is a contradiction to Lemma 9. Therefore, we state that $\mathcal{L}_{=1}$ is not ω-regular. □

On the other hand, we get

Lemma 13. *[Grö08, Theorem 4.4.9 (b)] There is an ω-regular language that cannot be accepted by any PBA with almost-sure acceptance.*

Proof. We examine the ω-regular language $\mathcal{L}$ over the alphabet $\{a,b\}$ of those words that contain only finitely many a (cp. Example 1). By similar arguments as for DBA we can derive the impossibility to accept this language with an almost-sure accepting PBA. To provoke a contradiction we assume the existence of a PBA $\mathcal{P} = (Q, q_0, \{a,b\}, \delta, F)$ which accepts $\mathcal{L}$. W.l.o.g. we can assume that all states in Q are reachable from q_0 and hence reading b^ω starting in any state must yield an acceptance measure of 1 for $\mathrm{Acc}(F)$. Moreover, every state p must allow to read an accepting state with a positive probability ϵ_p after a sequence of b^{n_p}; if we take the maximum of all the lengths of these sequences $n = \max\{n_p : p \in Q\}$ we know that every state must visit with positive probability (at least) $\epsilon = \min\{\epsilon_p : p \in Q\}$ an accepting state after reading b^n. This implies that for every state that is possible after a sequence $(b^n a)^k$ (for $k = 1, 2, \dots$) it is probable to read another accepting state within the next $b^n a$ sequence. Hence, fixing A_k as the observation that in the k-th reading of $b^n a$ from every state at least once a final state is visited allows to use Theorem 2 for the sequence $(A_k)_{k>0}$ and yields an almost-sure probability for reading infinitely many states in F for runs of $\mathcal{P}$. Formally, we set

$$A_k = Q^{(n+1)^k} \cdot \bigcup_{1 \le i \le n} Q^{i-1} F \cdot Q^{n-i+1} Q^\omega,$$

i.e. the occurence of at least one state F in the k-th occurence of $b^n a$. We observe that

$$\bigcap_{i>0} \bigcup_{j>i} A_j = \mathrm{Acc}(F),$$

since there must not be an index from which onwards F never again occurs. The pairwise independence of all A_k for $k > 0$ follows from the history-independence of the probability functions of the individual states[1]. □

And by combination of Lemma 13 and Lemma 12 we get

Theorem 19. *[BBG08, Theorem 4, (b), (c)] The classes of ω-regular languages and languages accepted by a PBA with almost-sure acceptance are incomparable.*

Considering this theorem and that using Dirac distributions at every state emulate determinstic choices the following result is transparent:

Proposition 1. *[Grö08, Proof of Theorem 4.4.9 (d)] The class of languages recognizable by PBAs with almost-sure acceptance strictly includes the class of languages recognizable by DBAs.*

This entails by the observation that $\{a, b\}^* b^\omega$ cannot be recognized by a PBA with almost-sure acceptance but the set of all words that contain infinitely many a is (since it is DBA recognizable)

Proposition 2. *[Grö08, Theorem 4.4.9 (d)] The class of languages recognizable with PBAs with almost-sure acceptance is not closed under complementation.*

However, the union construction for positively accepting PBAs can be used to obtain an intersection operator for almost-surely accepting PBAs.

Proposition 3. *[Grö08, Consequence of Section 4.3.] The class of languages recognizable with PBAs with almost-sure acceptance is closed under intersection.*

[1] This property of history-independence of the probability distributions is called *Markov*-property (cp. e.g. [Grö08, Page 8] or [Kle06, Chapter 17]).

Proof. For the union operator for positively accepting PBAs we observed that

$$\mu(\mathrm{Acc}(F)) = \frac{1}{2} \cdot \mu^1_\alpha(\mathrm{Acc}(F^1)) + \frac{1}{2} \cdot \mu^2_\alpha(\mathrm{Acc}(F^2))$$

for $F = F_1 \cup F_2$. Naturally, this entails

$$\mu(\mathrm{Acc}(F)) = 1 \text{ if and only if } \mu^1_\alpha(\mathrm{Acc}(F^1)) = 1 \text{ and } \mu^2_\alpha(\mathrm{Acc}(F^2)) = 1.$$

Hence, the claim follows. □

2.2.1. Markov Decision Processes

In the following we introduce Markov Decision Processes (MDPs), a model of interaction between a probabilistic domain and an actor within this domain. We call this actor Eve and treat her grammatically as female. The actions of Eve impact the behavior of the domain and depending on the behavior of the domain she may adapt her behavior. It shares significant structural similarities to PBAs but allows for a rather different interpretation. Nevertheless, we reuse various concepts from PBAs to define MDPs. The following definitions follow [CHS14]:

Definition 15. ***Markov Decision Process***:
A Markov Decision Process is modelled as tuple $\left(S, A, (\tau_a)_{a\in A}, s_0\right)$ where S is a set of states and A a set of actions. Given an action $a \in A$ the corresponding transition function $\tau_a : S \times S \to [0, 1]$ satisfies for every $q \in S$ that $\sum_{p\in S} \tau_a(q, p) = 1$. s_0 is the initial state. Additionally, we define a few helpful auxillary notions:

Plays We consider all infinite sequences of states valid plays. All plays implicitly start in s_0 and are gathered in $\mathrm{Plays} = S^\omega$.

Strategy We define $f : S^* \to A$ as strategy for a MDP.

A strategy f for an MDP $\mathcal{M} = \left(S, A, (\tau_a)_{a\in A}, s_0\right)$ induces a probability measure μ_f on the Borel-algebra $\mathcal{B}(S)$. Formally, we use the unique extension (in the sense of Theorem 1) of the measure μ_f from

$$\mu_f(\mathrm{cyl}(s_1 \dots s_n)) = \tau_{f(\epsilon)}(s_0, s_1) \prod_{1\le i<n} \tau_{f(s_1\dots s_i)}(s_i, s_{i+1}).$$

PBAs can be interpreted as MDPs where the decisions in every state are induced by a word $\alpha = \alpha_0\alpha_1 \dots \in \Sigma^\omega$ rather than a player. Any PBA $\mathcal{P}$ can be interpreted as MDP and for any α we can define a corresponding strategy $\varphi_\alpha : Q^* \to \Sigma$ with $\varphi_\alpha(w) = \alpha_{|w|}$. On the other hand, strategies for MDPs allow for a more refined interaction between Eve and the environment. Notably, Eve might react for plays of equal length differently based on which state she currently resides in, e.g.

Example 4. We interpret the PBA pictured in Figure 5 as MDP $\mathcal{M}$. Furthermore, we challenge Eve with the task to visit states in F infinitely often. She trivially has a strategy f that allows her to satisfy this task. Initially, she starts in q_0 and therefore plays $f(\epsilon) = a$ and in the resulting plays she uses

$$f(p_1 \dots p_n) = \begin{cases} b & \text{if } p_n = q_1, \\ a & \text{if } p_n = q_0. \end{cases}$$

Eve is not restricted (as words are) to be "consistent" regarding the length of the play. A word has exactly one letter at each position while Eve may provide different actions (or letters) after the same length of the play depending on the state she currently resides in (or even the current history of the play). Eve's strategy can therefore be encoded as a tree where the root is her initial move and moving down edges which are labelled with her observations leads to nodes which contain her decisions regarding that partial play.

We want to examine Eve's behavior given she does not observe the precise state of the game, but only some information. Conceptually, we color the state of the game and only reveal to Eve the color of the current state but not the precise state. This leads to the following definition of Partially Observable Markov Decision Processes (POMDPs). A POMDP is a MDP $\mathcal{M}$ with an associated equivalence relation $\sim$ which relates equally colored states. Hence, the equivalence relation models the restriction of Eve to observe the state of $\mathcal{M}$. For every $s \in S_{\mathcal{M}}$ we define its color by its equivalence class regarding $\sim$: $[s]_\sim = \{r \in S \mid r \sim s\}$ and $[S]_\sim$ as set of all these equivalence classes. A strategy for a POMDP $(\mathcal{M}, \sim)$ is defined as $f : ([S]_\sim)^* \to A$.

Example 5. Again, consider a PBA $\mathcal{P}$ and interpret it as an MDP $\mathcal{M}$ we observed that every word α induces a corresponding strategy f_α. But not every strategy induces a word because strategies on $\mathcal{M}$ may choose depending on the state of the play different letters for plays of the same length which creates an inconsistency with respect to an - hypothetical - associated word (cp. Example 4). We add to $\mathcal{M}$ an equivalence relation $\sim = S \times S$ (as suggested in [BBG08]), hence there is exactly one equivalence class. Firstly, we observe that f_α is still a valid strategy for the resulting POMDP $\mathcal{N} = (\mathcal{M}, \sim)$. Secondly, since there is exactly one equivalence class every strategy for $\mathcal{N}$ observes at any time only the length of the history. Hence, we can provide an equivalent strategy of the form $f : \mathbb{N} \to \Sigma$. This in turn means that every strategy in $\mathcal{N}$ induces a word $f(0)f(1)\dots$ in Σ^ω. Furthermore, the measure of one strategy f and an associated word α_f coincide, i.e. the language of $\mathcal{P}$ can be understood as strategy space for $\mathcal{N}$ and vice versa.

The definitions of MDPs and POMDPs are for now probabilistic transition systems and strategies induce a behavior of such a transition system. To model goals or task for the player to satisfy we again use Büchi-, Rabin-, Parity- and Muller-conditions (cp. Example 4). We can then ask if Eve has a strategy to positively or almost-surely satisfy the associated condition in an MDP or POMDP. The interpretation of PBAs as uniformly colored POMDPs described in Example 5 directly induces some undecideability results, namely

Corollary 3. *[BBG08, Corollary 3 (a)] [CDH10, Theorem 5] The question whether a strategy for a POMDP exists that either positively or almost-surely satisfies a Rabin-, Parity- or Muller-condition is undecideable. Moverover, the question whether a strategy for a POMDP that positively satisfies a Büchi-condition exists is also undecidable.*

Proof. Any algorithm to compute a strategy for a POMDP that positively satisfies a Büchi-condition immediately solves the emptiness problem for PBAs with positive acceptance which is undecideable by Theorem 17. The same argument holds for almost-sure and positive acceptance of Rabin-, Muller- or Parity-conditions and Corollary 2. □

On the other hand, there are some positive results regarding computation of strategies for MDPs and for almost-sure satisfaction of Büchi-conditions in POMDPs:

Theorem 20. *[CDH10, Theorem 5] It is possible to compute a strategy for a POMDP that almost-surely satisfies an associated Büchi-condition.*

This strategy can be computed in time exponential in the number of states of the POMDP and in general cannot be computed in less time.

and

Theorem 21. *[CJH04, Theorem 1.1] It is possible to compute a strategy for a MDP that almost-surely or positively satisfies an associated Büchi- or Parity-condition.*

This can be done in time polynomial in the number of states of the MDP.

The problem of obtaining such strategies is explored in more detail in Chapter 4. For now we are content with the existence of these decision procedures. Additionally, note that Theorem 20 induces a maximal complexity for deciding the emptiness problem for almost-surely accepting PBAs (Theorem 18).

3. Tree Automata

3.1. Automata on Infinite Trees

In Section 2.1 it is noted that word automata attach to a word a sequence of states. The idea of classic tree automata is to attach states to more complex structures, namely trees. For a finite set of directions D and a finite alphabet Σ we define a D-ary Σ-tree $t : D^* \to \Sigma$. A tree automaton produces a tree of states which is attached to the input tree, hence a D-ary Q-tree where Q is the set of states. The following introduction conceptually follows [GTW02, Chapter 8].

Definition 16. ***Tree automaton***:
Let Q be a finite set of states and $q_0 \in Q$ be one initial state. Furthermore, let Δ denote a set of transitions of the form $\left(q, \sigma, (q_d)_{d \in D}\right)$ where $q \in Q, q_d \in Q$ for all $d \in D$ and $\sigma \in \Sigma$. We define a tree automaton

$$\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{Acc} \subseteq Q^\omega)\,.$$

And its semantics with the following auxillary notions

Run a run of $\mathcal{A}$ on a D-ary Σ-tree t is a D-ary Q-tree r such that $r(\epsilon) = q_0$ and for every $u \in D^*$ there is one $\left(q, \sigma, (q_d)_{d \in D}\right) \in \Delta$ with $r(ud) = q_d$,

Acceptance Condition the acceptance condition Acc is considered to be represented as Büchi-, Parity-, Muller- or Rabin-condition,

Projection of a tree for any $\alpha = \alpha_1\alpha_2 \cdots \in D^\omega$ and any D-ary Σ-tree t let $t(\alpha)$ denote the word $\beta = \beta_0\beta_1 \cdots \in \Sigma^\omega$ such that for every $i \in \mathbb{N}_{>0}$ holds $\beta_i = t(\alpha_1 \dots \alpha_i)$ and $\beta_0 = t(\epsilon)$.

$\mathcal{A}$ accepts a tree t if there exists a run r of $\mathcal{A}$ on t such that for every $\alpha \in D^\omega$ $r(\alpha) \in \mathrm{Acc}$.

Again, we introduce acronyms for the considered tree automata, i.e. Parity Tree Automaton (PTA), Rabin Tree Automaton (RTA), Muller Tree Automaton (MTA) and Büchi Tree Automaton (BTA), and present an easy example to familiarize ourselves with the notions of tree automata

Example 6. Consider $\mathcal{L}_{\exists a}$ as the language of those binary trees over $\{a, b\}$, i.e. $\{0, 1\}$-ary $\{a, b\}$-trees, that contain at least one a. This language can be accepted by a tree automaton by non-deterministically choosing a path to an occurence of an a and rendering every other path accepted. From the point onwards where an a occured the chosen path is rendered accepted. The corresponding tree automaton can be described by two states q_s, q_a, where q_s is a state that searches for an a while q_a is a state that renders all its subtrees accepting. q_a reproduces itself into all its subtrees, while q_s reproduces itself only in one of its subtrees and "dismissing" the other one by sending q_a in it. The acceptance condition is described as a Büchi condition by the set $F = \{q_a\}$. Thus, since q_a only produces q_a in all following subtrees, it can be also understood as a Reachability-condition, i.e. a condition which is satisfied as soon as one desired element occurs. This yields the tree automaton

$$\mathcal{A}_{\exists a} = (Q = \{q_s, q_a\}, q_s, \Delta, F = \{q_a\})$$

with

$$\Delta = \{(q_a, a, q_a, q_a), (q_a, b, q_a, q_a), (q_s, b, q_s, q_a), (q_s, b, q_a, q_s), (q_s, a, q_a, q_a)\}.$$

For all trees in $\mathcal{L}_{\exists a}$ the automaton $\mathcal{A}_{\exists a}$ can produce an accepting run by "guessing" a path to the occurence of a while every tree accepted by $\mathcal{A}_{\exists a}$ does need to contain an a since otherwise q_s is never transformed into q_a which yields one non-accepting path in any run. Enforcing, that q_s always reproduces into both subtrees yields an automaton which recognizes the language of those trees which do have an a on every branch. We denote this language as $\mathcal{L}_{\forall a}$.

It is noteworthy that $\mathcal{A}_{\exists a}$ uses a very simple acceptance condition, namely a Reachability-condition which is emulated by a Büchi-condition. The following example on the other hand uses a more elaborate acceptance condition:

Example 7. This example is similiar to e.g. [GTW02, Exercise 8.3 and the proof of Theorem 8.6]. Let $\mathcal{L}_{\not\infty a}$ be the language of all binary trees over the set $\{a, b\}$ that contain on every path only finitely many a. Note the similarity to Example 1 for word automata. This language can be recognized by tree automata with a very easy structure, namely there are two states q_a, q_b which only are visited if the corresponding letter is read. Thus, we define

$$\Delta = \{(q_a, a, q_a, q_a), (q_b, a, q_a, q_a), (q_a, b, q_b, q_b), (q_b, b, q_b, q_b)\}.$$

Any Rabin-, Muller- or Parity-condition as described in Example 1 accepts the correct language since these precisely define the abscence of infinitely many occurences of q_a.

For the corresponding NBA in Example 1 we used a structural way to model the abscence of further occurences of a from one point onwards. This argument of guessing the moment when no further a appears cannot be used in the case of trees since every such guess involves the complete subtree. Considering the tree $t : \{0,1\}^* \to \{a,b\}$ with $t(w) = a$ if and only if $w = 1^n0$ for any $n \in \mathbb{N}_{>0}$ we see that $t \in \mathcal{L}_{\not\infty a}$, but for all $k \in \mathbb{N}_{>0}$ the subtree rooted in 1^k contains an a. Thus, there is no point from which this subtree is a-free, therefore this idea of guessing such a point does not translate to tree automata.

The difficulties to define a BTA to accept $\mathcal{L}_{\not\infty a}$ are inherent to the Büchi-condition. In fact it is possible to show that $\mathcal{L}_{\not\infty a}$ cannot be accepted by any BTA, thus separating the expressiveness of tree automata as in

Theorem 22. *[Rab70, Corollary 8] cited after [GTW02, Theorem 8.6]. BTAs are strictly weaker than MTA in the sense that there exists a language recognizable by a MTA but not recognizable by a BTA.*

This is in contrast to Theorem 8 where we could use the non-determinism of an NBA to emulate a more expressive acceptance condition, e.g. a Muller-condition. But by the same techniques as for the proof of Theorem 8 for the equivalence of the "strong" acceptance conditions, namely LARs, we can ensure

Theorem 23. *[GTW02, Theorem 8.7] MTAs, RTAs and PTAs all recognize the same class of tree languages.*

Regarding the closure properties of this class of tree languages it is again possible to show its closure under union, intersection and negation. This gives

Theorem 24. *[Rab68, Theorem 1.3] [Rab68, Theorem 1.5] The class of languages that can be accepted by MTAs, RTAs and PTAs forms a Boolean-algebra. The transformations can be effectively constructed.*

Proof-Sketch. The concepts follow the proof of Theorem 11: given two PTAs $\mathcal{A}_1, \mathcal{A}_2$ we construct one PTA $\mathcal{A}$ which performs an initial *guess* deciding to execute either $\mathcal{A}_1$ or $\mathcal{A}_2$. Hence, if there is an accepting run for either $\mathcal{A}_1$ or $\mathcal{A}_2$ then there is an accepting run for $\mathcal{A}$ and vice versa. Therefore, $\mathcal{A}$ accepts the union of the languages of $\mathcal{A}_1$ and $\mathcal{A}_2$.

An operator for intersection is achieved by relying on a product construction which simulates both input automata. Using a Muller-condition which formulates the acceptance of the individual components by their original acceptance conditions completes the construction.

The closure for complementation is proven using *alternating* tree automata and therefore postponed until we formally introduced alternating tree automata in the following section. □

3.1.1. Alternating Tree Automata

Up to this point the used automata mirrored the structure of their inputs very closely. Word automata generated words of states and tree automata generated trees of states of the same arity of their input trees. This notion can be relaxed for tree automata to obtain a new class of tree automata, called *Alternating Tree Automata*. Following [MS87] tree automata operate by sending down "copies" through the different paths. In the classical setting at every branching point every successor is awarded with one copy of a state of the automaton. The aforementioned relaxation lies in the idea to allow at a branching point to award some successor multiple states while others may not even receive a single one. This concept is also used to define automata that run over transition systems, e.g. [GTW02, Chapter 9] and [Sch06], where the input tree is the unrollment of the transition system.

Up to this point, transitions for tree automata are defined as tuples of the form $\left(q, \sigma, (q_d)_{d \in D}\right)$. This specifically defines for every direction one single state to move along this direction. We introduce transitions of the form (q, σ, A) where A is a non-empty subset of $A \subseteq D \times Q$ called a *clause*.

Semantically, A describes that a copy of q is sent down the direction d for every $(d, q) \in A$. This notion allows us to refine the tree automaton of Example 6:

Example 8. Recall $\mathcal{A}_{\exists a}$ from Example 6 which determines if a tree contains an a with

$$\Delta = \{(q_a, a, q_a, q_a), (q_a, b, q_a, q_a), (q_s, a, q_a, q_a), (q_s, b, q_s, q_a), (q_s, b, q_a, q_s)\}.$$

The concept of alternating tree automata allows to formulate the "searching" part of the run more concisely[a] by dropping the necessity to send a q_a along the path that is not chosen, thus obtaining

$$(q_s, b, \{(0, q_s)\}), (q_s, b, \{(1, q_s)\}).$$

[a] Occasionally, e.g. in [GTW02, Chapter 9], alternating tree automata are equipped with the possibility to render a subtree accepted regardless of its structure. We explicitly do not include this notion, but it is easy to see that this would allow to equivalently define transitions $(q_a, a, \delta_\top), (q_a, b, \top), (q_s, a, \top)$ where $\top$ renders the subtree accepted.

We formally define similar to [Sch06]

Definition 17. ***Alternating Tree Automata***:
For a finite set of states Q, one distinct intial state $q_0 \in Q$, a finite set of directions D, a finite alphabet Σ and a transition relation Δ we define

$$\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{Acc} \subseteq Q^\omega).$$

The semantics of $\mathcal{A}$ is given by a run r of $\mathcal{A}$ on a D-ary Σ-tree t. Here r is a $D \times Q$-ary $\{0,1\}$-tree with the following neccessities:

1. $r(\epsilon) = 1$,
2. for every $u = (d_1, q_1) \dots (d_n, q_n) \in (D \times Q)^*$ with $r(u) = 1$ there exists $(q_n, t(d_1 \dots d_n), A) \in \Delta$ such that

$$r(u\,(d,q)) = 1 \text{ if and only if } (d,q) \in A.$$

We consider such a run r accepting if the Q-component of all paths that are consistently marked are part of Acc. Formally, we fix for any run r the set of its marked path as

$$M_r = \{\rho \in (D \times Q)^\omega \mid r(v) = 1 \text{ for all finite } v \sqsubseteq \rho\}\,.$$

The set of accepted paths is given by

$$\mathrm{Acc}\,[Q] = \{(d_1, q_1)\,(d_2, q_2) \dots \in (D \times Q)^\omega \mid q_1 q_2 \dots \in \mathrm{Acc}\}\,.$$

(Notably, $\mathrm{Acc}\,[Q]$ denotes those words such that the Q-projection is part of Acc.) A run r is considered accepting if $M_r \subseteq \mathrm{Acc}\,[Q]$. Again, we consider for Acc acceptance conditions which are representable as Büchi-, Rabin-, Muller- or Parity-conditions.

We introduce according acronyms, namely Alternating Büchi Tree Automaton (ABTA), Alternating Rabin Tree Automaton (ARTA), Alternating Muller Tree Automaton (AMTA) and Alternating Parity Tree Automaton (APTA) respectively. By using the same argument of using LARs as for Theorem 8 and Theorem 23 we obtain

Theorem 25. *[MS95, Consequence of Theorem 1.2] ARTA, AMTA and APTA recognize the same class of tree languages.*

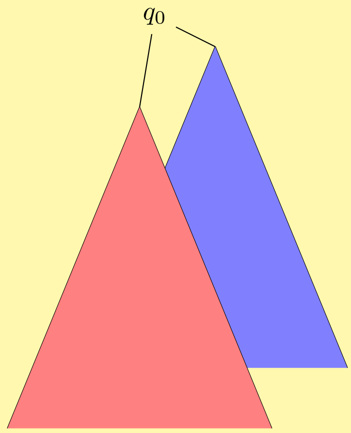
Although *alternation* is an extension of the "classic" tree automata it turns out that it does not increase the expressiveness. Specifically, for every alternating tree automta there is a classic tree automata which accepts the same language of trees.

Theorem 26 (The Simulation Theorem)**.** *[MS95, Theorem 1.2] There is an effective construction which, when given an AMTA, produces an equivalent MTA. Furthermore, given an ABTA, there is a way to effectively construct an equivalent BTA.*

This result and the apparent observation that alternation does not weaken the formalism to accept trees allows to translate the closure properties stated in Theorem 24 to alternating tree automata as well. Thus, we obtain that there are effectively constructable transformations for the union, intersection and complement of tree languages accepted by AMTAs. Nevertheless, we want to present one construction for the closure under intersection because it allows us to relate concepts of PBAs and of alternating tree automata.

Corollary 4. *The class of languages of infinite trees that can be recognized by AMTAs is closed under intersection.*

Figure 7. Beginning from q_0 two run-trees unfold independently for an alternating tree automata. If we consider this as the runs of a union (intersection) positively (almost-surely) accepting PBA both runs are tuned down by the factor $\frac{1}{2}$.



Alternative Proof. Given two AMTAs

$$\mathcal{A}_1 = (Q, q_0, D, \Sigma, \Delta_1, \mathcal{F}_1) \text{ and } \mathcal{A}_2 = (P, p_0, D, \Sigma, \Delta_2, \mathcal{F}_2)$$

we construct an AMTA for their intersection. W.l.o.g. we assume that $Q \cap P = \emptyset$. The idea is to perform two independent runs on the tree in parallel. We illustrate this in Figure 7. Formally, it is accomplished by initially dispatching one transition of each automaton onto the tree. After this initial dispatch both state sets Q and P operate independently on the tree. Hence, we define

$$\mathcal{A}_\cap = \left(Q \cup P \uplus \left\{q^i\right\}, q^i, D, \Sigma, \Delta_1 \cup \Delta_2 \cup \Delta_\cap, \mathcal{F}_1 \cup \mathcal{F}_2\right)$$

with

$$\Delta_\cap = \left\{\left(q^i, \sigma, A \cup B\right) : (q_0, \sigma, A) \in \Delta_1 \text{ and } (p_0, \sigma, B) \in \Delta_2 \text{ for every } \sigma \in \Sigma\right\}.$$

Thus, we can construct from the resulting run r in $\mathcal{A}_\cap$ two individual runs in $\mathcal{A}_1$ and $\mathcal{A}_2$ respectively. For $\mathcal{A}_1$ we obtain a run r_1 from the $D \times Q$-part of the domain of r and the $D \times P$-part of the domain of r induces a run r_2 for $\mathcal{A}_2$. □

While considering the semantics of PBAs we used a notion of considering individual runs and their probabilities. This can be captured very similar to the concept of alternation as follows: we can regard any word $\alpha \in \Sigma^\omega$ as a $\{0\}$-ary Σ-tree t_α. Then one PBA $\mathcal{P}$ induces a run in a tree-sense by unrolling the state sequences of $\mathcal{P}$ around the unary tree t_α. Hereby, we can substitute clauses which have a very absolute semantic (either a path is taken or it is not) by a probability distribution. For a PBA $\mathcal{P} = (Q, \Sigma, \delta, q_0, F)$ we introduce transition functions $G^q_\sigma : \{0\} \times Q \to [0,1]$ such that $G^q_\sigma(0,p) = \delta(q,\sigma,p)$ and fix

$$\mathcal{G} = \{G^q_\sigma : \sigma \in \Sigma, q \in Q\} .$$

We can model the run of $\mathcal{P}$ as a $\{0\} \times Q$-ary $\mathcal{G}$-tree where the chosen G^q_σ one the paths $(0,q_1)\dots(0,q_n)$ are consistent with the current state ($q = q_n$) and input symbol ($\sigma = t(0^n)$). Therefore, every path corresponds to one state sequence of $\mathcal{P}$ and the visited G^q_σ on the path induce the associated probabilities for the peformed steps. In this sense we can use Figure 7 as illustration for the union operator of the proof of Theorem 14. Thereby, the initial separating dispatch is in every direction tuned down by a factor of $\frac{1}{2}$. This intuition is later on re-visited and formally substantiated.

In the meantime we introduce the concept of *graph games* to obtain a transformation that constructs the APTA which accepts the negation of the language of a given APTA.

3.1.2. Graph Games

Tree automata are closely related to graph games (cp. [GTW02, Chapter 9]). Such games are held between two players in an *arena*. An arena is a directed graph where the nodes are partioned into two sets; each set is associated with one player. A game on such an arena begins in one node and unfolds by the choices of the players. The player to whom the current node v belongs may choose the node u the game proceeds to from the set of those nodes connected to v. By associating a winning condition for one player to the game we can call a game won by this player if the played state sequence belongs to this winning condition. In the following we focus on zero-sum games, i.e. if the winning condition of one player is not met we consider the opponent victorious. This leads to the definition

Definition 18. ***Graph Games***:

Arena Let V be a set of nodes which is partioned into V_0 and V_1 and $E \subseteq V \times V$ a set of edges, then we define an arena (V, V_0, V_1, E).

Game For an arena (V, V_0, V_1, E) we define with a winning condition Acc and one distinguished initial vertex $v_0 \in V$ the game $(\mathcal{G}, v_0) = (V, V_0, V_1, E, \mathrm{Acc}, v_0)$.

Play A play for a game $(\mathcal{G}, v_0) = (V, V_0, V_1, E, \mathrm{Acc})$ is any sequence $v_0 v_1 \dots \in V^\omega$ which starts in v_0 and where for every $i > 0$ holds that $(v_i, v_{i-1}) \in E$.

Strategy For $\sigma \in \{0,1\}$ we define a strategy $f_\sigma : V^*V_\sigma \to V$ where for any $w \in V^*$ and $v \in V_\sigma$ $f_\sigma(w \cdot v) \in vE$ (where $vE = \{u \in V \mid (v,u) \in E\}$). A play is considered winning for one player (Eve) if it belongs to Acc. Notably it holds that two strategies f_0, f_1 induce a play $v_0v_1\dots$ by fixing for any $v \in V$ the term

$$\sigma(v) = \begin{cases} 0 & \text{if } v \in V_0 \\ 1 & \text{if } v \in V_1 \end{cases} \text{ and enforcing } v_{i+1} = f_{\sigma(v_i)}(v_0 \dots v_i).$$

If a strategy for player σ can equivalently be represented as a function $f : V_\sigma \to V$ (naturally still satisfying that $f(u) \in uE$ for all $u \in V_\sigma$) we call this strategy *memoryless* or *positional*.

Winning We say a play $v_0v_1\dots$ is *consistent* with a strategy f for player σ if for every i with $v_i \in V_\sigma$ holds that $v_{i+1} = f(v_0 \dots v_i)$. A strategy f for player σ is called a *winning strategy* if *every* play consistent with f is winning for player σ. We denote with *winning regions* $\mathcal{W}_\sigma^\mathcal{G}$ those nodes $v \in V$ such that player σ has a winning strategy in $(\mathcal{G}, v)$.

Acc is represented in finite ways by using Büchi-, Rabin-, Muller- or Parity-conditions. In the following we call the player for whom the acceptance condition Acc is stated Eve (as for MDPs) and her opponent Adam (in the following we treat Adam grammatically as male).

Given any game $\mathcal{G}$ the question arises whether for every node v either Eve or Adam wins by playing the correct strategy (regardless of the strategy of the opponent), i.e. $\mathcal{W}_0^\mathcal{G} \cup \mathcal{W}_1^\mathcal{G} = V$. We call a game with this property *determined*. There is a fundamental result by Martin establishing determinacy for all Borel winning conditions for a more general class of games[1] [Mar75]. Fortunately, this translates to graph games and allows us to state using that Büchi-, Rabin-, Muller- and Parity-conditions are Borel (see Lemma 8 and its Corollary 1)

Theorem 27. *[GTW02, Corollary 2.10] All graph games with Büchi-, Rabin-, Muller- or Parity-condition are determined.*

This means that given for every strategy f of Eve Adam has a strategy g which induces a play not in Acc Adam can find *one* strategy g' which is winning against *all* strategies f' of Eve. For Parity-conditions we even obtain that the corresponding strategies are positional, i.e.

Theorem 28. *[Zie98, Theorem 6] For a graph game with a Parity-condition with finitely many parities both players have positional winning strategies on their corresponding winning regions.*

As mentioned above we use graph games to model the semantics of tree automata. Therefore, we define an acceptance game for an automaton and a tree. In this (infinite)

[1] Gale-Stewart games

graph game EVE constructs a run of the automaton on the tree and ADAM explores this run and tries to find a path which is not accepted. Formally, we restrict ourselves to APTA since they are sufficiently expressive and define (cp. [Löd12; MS87])

Definition 19. ***Tree Acceptance Game***:
For a APTA $\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{par})$ and a D-ary Σ-tree $t : D^* \to \Sigma$ we define the graph game $\mathcal{G}(\mathcal{A}, t) = (V, V_0, V_1, E, \mathrm{par}', (q_0, \epsilon))$ with

Arena $V_0 = Q \times D^*, V_1 = \mathrm{Pot}(D \times Q) \times D^*$ and $V = V_0 \cup V_1$,

Edges EVE chooses the taken transition which is captured by

$$E_0 = \{((q, u), (A, u)) : \text{for all } q \in Q, u \in D^* \text{ and } (q, t(u), A) \in \Delta\}.$$

ADAM on the other hand chooses which element in A is taken, i.e.

$$E_1 = \{((A, u), (q, u \cdot d)) : \text{ for all } A \in \mathrm{Pot}(D \times Q), u \in D^* \text{ and } (q, d) \in A\}.$$

Naturally, we set $E = E_0 \cup E_1$.

Winning Condition the associated winning condition is the projection of the old condition to the state component for nodes in V_0, i.e.

$$\mathrm{par}'((q, u)) = \mathrm{par}(q).$$

Nodes in V_1 should not impact the evaluation of a game directly, therefore we set their parities lower than all parities used in $\mathcal{A}$:

$$\mathrm{par}'((\delta_A, u)) = p_\perp \text{ for one } p_\perp < \min\{\mathrm{par}(q) : q \in Q\}.$$

An illustration of such a game can be found in Figure 8.

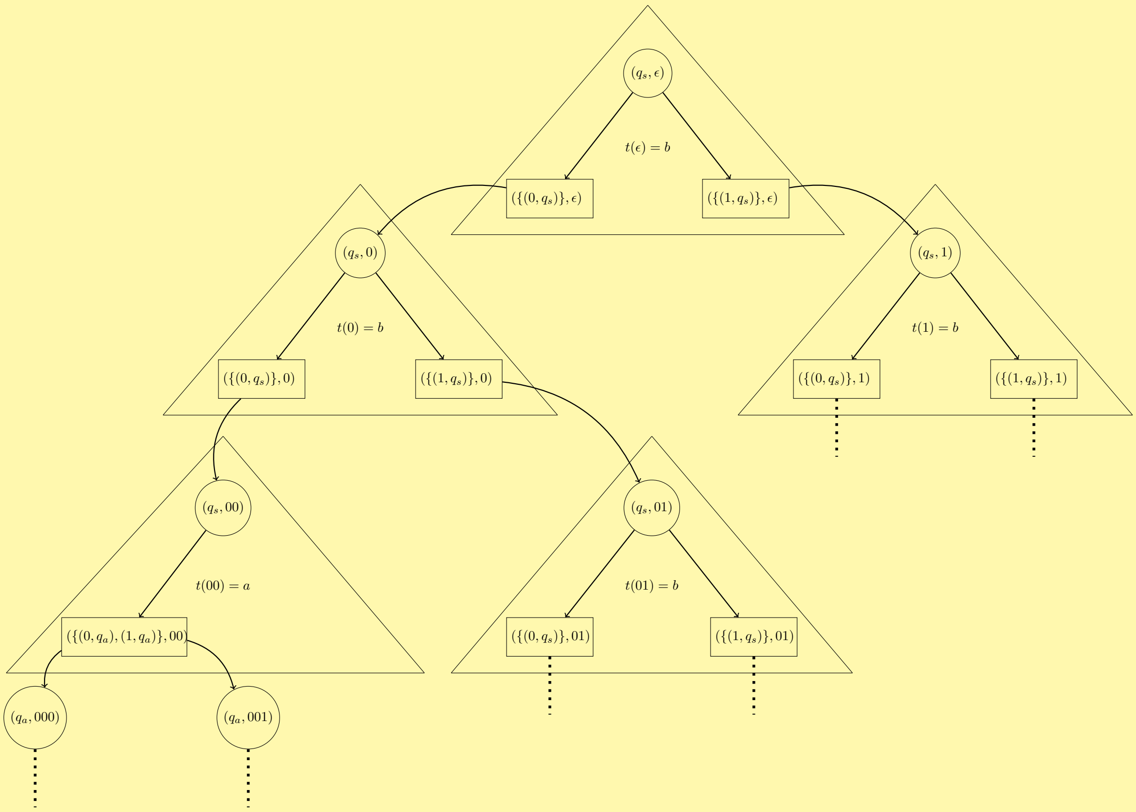
The central insight for this game is the following lemma which connects the acceptance of $\mathcal{A}$ of t with the strategies of EVE in $\mathcal{G}(\mathcal{A}, t)$.

Lemma 14. *[MS87, Lemma 3.1.] The automaton $\mathcal{A}$ accepts t if and only if* EVE *has a winning strategy in the game $\mathcal{G}(\mathcal{A}, t)$.*

In the following we present a proof for this lemma which reveals the central idea behind the construction of a dual automata for a given APTA.

Proof. We can connect strategies f for EVE with runs r of the automaton $\mathcal{A}$ on t directly. Initially, EVE starts in state (q_0, ϵ) and chooses an initial clause A (by her move to (A, ϵ) which by definition of E_0 corresponds to a transition $(q_0, t(\epsilon), A)$. This A can be used to define the first level of a run r of $\mathcal{A}$ on t by setting $r((q, d)) = 1$ if and only if $(q, d) \in A$. ADAM does now choose which element of A to explore (by moving to (q, d)).

Figure 8. Illustration of $\mathcal{G}(\mathcal{A}, t)$ for $\mathcal{A}$ as defined in Example 6 and a $\{0,1\}$-ary $\{a,b\}$-tree t with $t(u) = a$ if and only if $u = 00$. Nodes of player 0 are circles while player 1 plays on rectangles. Every triangle is associated with one word $u \in \{0,1\}^*$ and labeled with $t(u)$. Player 0 may win by moving towards the $t(00)$ subgame, which is easily achievable since player 1 only has trivial moves up to that point.



This can be identified by "moving" to $r(q, d)$. Inductively, Eve's choices correspond to chosen transitions in r and Adam chooses the precise marked paths to explore. Any combination of strategies f and g for Eve and Adam respectively yields a play which corresponds to one path in M_r for one run r (especially r is induced by f and all possible strategies of Adam). Eve wins if this chosen path satisfies the Parity-condition on the Q-component (notably because the definition of the Parity-condition $\mathcal{G}(\mathcal{A}, t)$ ignores nodes of V_1). Considering a winning strategy f for Eve any opposing strategy of Adam is loosing rendering all elements in the corresponding M_r accepted. On the other hand, if there is no winning strategy f for Eve the induced run r and the corresponding M_r always contains at least one not accepted element (which corresponds to the choices Adam may take which lead to Eve's loss). $\square$

Notably, Theorem 27 in combination with Lemma 14 allows to state that $\mathcal{A}$ does not accept t if and only if Adam has a winning strategy in $\mathcal{G}(\mathcal{A}, t)$. Constructing a dual automaton to $\mathcal{A}$ resolves around the idea to encode Adam's strategies in the transitions rather than Eve's. Thus, given the game is in state q and t carries at the current position the letter σ Eve may choose according to any transition of $\mathcal{A}$ for this situation, namely $(q, \sigma, A_1), \ldots, (q, \sigma, A_n)$. Adam's strategy needs to state for every choice of Eve one element in the chosen A_i for $1 \leq i \leq n$ such that he wins the corresponding play. We construct Adam's choices as

$$R_{q,\sigma} = \{\{(d_1, q_1), \ldots, (d_n, q_n) \mid (d_i, q_i) \in A_i \text{ for } 1 \leq i \leq n\}\}.$$

We obtain for a set of transitions Δ the "inverse" ∇ by

$$\nabla = \{(q, \sigma, B) : q \in Q, \sigma \in \Sigma, B \in R_{q,\sigma}\}.$$

We use the transitions in ∇ to effectively describe the game $\mathcal{G}(\mathcal{A}, t)$ with exchanged roles (in the sense of which "type" of moves are executed) of Eve and Adam. Adam pro-actively chooses his answers (which is encoded as transition) and Eve choice of one direction corresponds to an implicit choice of a transition, namely the one to which Adam answers by this move. Considering any transition in ∇ we observe that for every possible choice of Eve in parallel one answer of Adam is explored. This captures the usage of determinacy by exploring all of Eve's option since a winning strategy of Adam produces for all choices of Eve a winning play, hence choosing the set of answers that are initial for winning plays allows to encode Adam's winning strategy. We construct for a given APTA

$$\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{par}),$$

the automaton

$$\mathcal{C} = (Q, q_0, D, \Sigma, \nabla, \mathrm{par} + 1).$$

Assuming Adam does have a winning strategy in $\mathcal{G}(\mathcal{A}, t)$, then Eve[2] has a winning strategy in the game $\mathcal{G}(\mathcal{C}, t)$ since par $+1$ precisely inverts the set of accepted paths. Technically, any run explores all possible plays of Adam in $\mathcal{G}(\mathcal{A}, t)$ while the choices of Eve induce one path in this run. Therefore $\mathcal{C}$ is the dual automaton to $\mathcal{A}$ implying

Proposition 4. *[MS87, Complementation Theorem] The class of languages of trees accepted by APTAs is closed under negation. This holds by Theorem 26 also for PTAs.*

Concludingly, we want to examine the problem of deciding emptiness of tree automata. Again, by the effective transformations between alternating and non-alternating tree automata (Theorem 26) it is sufficient to examine one of those types. We opt for non-alternating tree automata and discuss the reason for this after introducing a game to capture the emptiness of tree automata. Formally, we follow [Zie98, Proof of Theorem 23] when considering for a given PTA

$$\mathcal{M} = (Q, q_0, D, \Sigma, \Delta, \mathrm{par})$$

the following graph game

[2] Beware the role-exchange!

Definition 20. ***Emptiness Game***:
EVE chooses transitions depending on the current state, while ADAM chooses the direction of a transition. Therefore we define the emptiness game by the following notions.

Arena EVE is assigned the states $Q = V_0$ of $\mathcal{M}$, while ADAM operates on the transitions $\Delta = V_1$ of $\mathcal{M}$. The edges correlate to the choices both players make, i.e. $E = E_0 \cup E_1$ with

$$E_0 = \left\{(q, \delta) \in Q \times \Delta \mid \delta = \left(q, \sigma, (p_d)_{d \in D}\right) \in \Delta\right\}$$

and

$$E_1 = \left\{\left(\left(q, \sigma, (p_d)_{d \in D}\right), p\right) \in \Delta \times Q \mid \text{there is } b \in D \text{ such that } p_b = p\right\}.$$

We obtain with $V = V_0 \cup V_1$ an arena (V, V_0, V_1, E).

Winning Condition EVE wins the game if the explored state sequence is accepted in terms of par, hence we define for any $\perp < \min\{\mathrm{par}(q) : q \in Q\}$

$$\mathrm{par}'(\Gamma) = \begin{cases} \mathrm{par}(q) & \text{if } \Gamma = q \in Q, \\ \perp & \text{otherwise.} \end{cases}$$

We define the emptiness game as $\mathcal{G} = (V, V_0, V_1, E, \mathrm{par}', q_0)$.

This game is similar to the one from Definition 19, but instead of choosing a transition for the corresponding letter in a tree EVE is free to choose any transition (and therefore implicitly a letter). The central insight for the emptiness game is formulated in

Lemma 15. EVE *has a winning strategy in* $\mathcal{G}$ *as defined above if and only if the language of* $\mathcal{M}$ *is not empty.*

Proof. Initially, we observe that for any play $q_0\delta_0q_1\delta_1\ldots$ we obtain by definition of par$'$ that $\max\{\mathrm{par}(q) : q \in \mathrm{Inf}(q_0q_1q_2\ldots)\} = \max\{\mathrm{par}'(\Gamma) : \Gamma \in \mathrm{Inf}(q_0\delta_0q_1\delta_1\ldots)\}$.

Consider a winning strategy $f : V^*Q \to \Delta$ for EVE, and an arbitrary riposte $g : V^*\Delta \to Q$ for ADAM. By the definition of E_1 we may equivalently define the strategy of ADAM in terms of $g : V^*\Delta \to D$. In this form, we obtain for f an associated tree $t_f : D^* \to \Sigma$ by considering for any $u_0 \ldots u_n \in D^*$ the play $q_0\delta_0q_1\delta_1\ldots$ which is consistent with f and the riposte g with $g(q_0\delta_0\ldots q_i\delta_i) = u_i$ for $0 \leq i \leq n$ and arbitrary otherwise. Naturally, we set $t(u_0\ldots u_n) = \sigma$ for one σ such that $f(q_0\delta_0\ldots q_n) = \delta_n = \left(q_n, \sigma, (p_d)_{d \in D}\right) \in \Delta$. By the construction of t we see that the choices of f immediately describes a possible run r on t. Since f is winning all paths ADAM can describe are won by EVE (accepted under par$'$) and therefore all paths of r are accepting in terms of par, rendering r an accepting run for t of $\mathcal{M}$.

If we have on the other hand, an accepted tree t of $\mathcal{M}$ with a corresponding run r then Eve simply plays r, i.e. we define f_r with $f_r(q_1\delta_0 \dots q_n) = \delta_n = r(u)$ for the $u \in D^*$ that corresponds to the choices of Adam which led to the play $q_0\delta_0 \dots q_n$. Again, all paths in r are accepting in terms of par rendering all plays consistent with f_r accepted in terms of par′. Hence, f_r is a winning strategy. □

This lemma directly allows us to state

Theorem 29. *[Rab72, Theorem 21] [GTW02, Theorem 7.25] The problem if the language of a given PTA*

$$\mathcal{M} = (Q, q_0, D, \Sigma, \Delta, \mathrm{par})$$

is empty can be decided in time in $\mathcal{O}(|\mathrm{par}(Q)| \cdot |Q \times Q| \cdot \left(\frac{|Q|}{d}\right)^d)$ *with* $d = \lfloor \frac{|\mathrm{par}(Q)|}{2} \rfloor$.

Proof. The stated complexity relates to the complexity of computing winning regions in parity games. Using Lemma 15 directly induces the statement. □

By Theorem 28 we even obtain a positional winning strategy for Eve (if there is any winning strategy)[3].

Note that we designed this emptiness game in terms of PTA rather than APTA. This is due to the fact that the choices of Eve must be consistent with an underlying tree (cp. Example 4). We illustrate this problem in

Example 9. Consider an APTA

$$\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{par})$$

with $Q = \{q_0, q_a, q_b, \bot\}$, $D = \{0, 1\}$, $\Sigma = \{a, b\}$, $\mathrm{par}(q_0) = \mathrm{par}(q_b) = \mathrm{par}(q_a) = 0$ and $\mathrm{par}(\bot) = 1$. The conceptual idea is that q_b (q_a) must move along a path that solely contains of b (a) but any occurence of a (b) yields a move to the non-accepting sink-state $\bot$ which reproduces itself towards all directions (necessarily rendering these paths rejecting). Hence, we add

$$\{(q_\sigma, \sigma, \{(q_{\sigma,d})\}), (q_\sigma, \overline{\sigma}, \{(\bot, 0), (\bot, 1)\}) : \text{for all } \sigma \in \{a, b\}, d \in \{0, 1\}\}$$

to Δ where $\overline{a} = b$ and $\overline{b} = a$. Initially, q_0 dispatches q_a and q_b in the *same* direction (by arbitrary choice) 0, yielding

$$\{(q_0, \sigma, \{(q_a, 0), (q_b, 0)\}) : \text{for } \sigma \in \{a, b\}\}$$

as part of Δ. Naturally, there is *no* possible accepted tree for $\mathcal{M}$ since $t(0)$ is either a or b but each choice leaves one unsatisfied state q_b or q_a respectively. On the

[3] This means that the structure of the obtained tree t indeed is - to a certain degree - simple. Theorem 23 in [Zie98] calls such a tree t *regular* which means that the number of isomorphic subtrees in t is finite. This is similar to ultimatively periodic words for ω-regular languages.

other hand, if we naively translate the concept of the emptiness game (Definition 20) Eve still can win by constructing different trees depending on the fact if q_a or q_b is chosen by Adam in the intial step. For PBAs we addressed this problem with partial observability (see Example 5). But for PBAs Eve only has one choice to make, namely one letter for the current position in the word. Here Eve is entitled to two choices: which letter to choose and which transition to take for this letter and the current state. This requires for a dedicated knowledge management for Eve because it has to be enforced that Eve chooses letters consistent with a tree structure and notably independent from the current state while the choice of the transition is highly dependent on the current state.

3.2. Weighted Descent Tree Automata

The basis of the tree automata we want to present in this section is given in [CHS14]. Namely, in [CHS14] tree automata are equipped with an alternative semantics: a uniform probability measure at each branching point of a run is introduced and the run is considered accepted if the measure of those paths that satisfy an associated condition (e.g. Büchi- or Parity-condition) have a certain measure. As for PBAs this allows for almost-sure or positive acceptance of runs. In the following we examine suggested generalizations of [CHS14] by designing more expressive transitions by adding individual probability measures for them. Furthermore, we incorporate the concept of *alternation* by allowing to send different states on the same path through the tree with individual weight. Formally, we define in analogy to clauses a notion of a weighted clause as probability distribution over the set $D \times Q$, i.e.

Definition 21. ***Generator***:
For a set of states Q and a set of directions D we call a probability function on $Q \times D$, i.e. $G : Q \times D \to [0, 1]$ with

$$\sum_{q \in Q, d \in D} G(q, d) = 1,$$

a generator over Q and D.

These generators are used to substitute clauses in alternating tree automata and we obtain the following

Definition 22. ***Weighted Descent Tree Automata***:
We define a Weighted Descent Tree Automaton as tuple $\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{Acc})$ where Q is a finite set of states, q_0 the initial state, D a finite set of directions, Σ is a finite Alphabet and $\mathrm{Acc} \subseteq Q^\omega$ is a ω-regular target of infinite words of elements

in Q. Again, we consider Acc to be expressed as Büchi-, Parity-, Rabin- or Muller-condition. Δ is defined as a finite set of transitions (q, σ, G) with $q \in Q, \sigma \in \Sigma$ and G is a generator over Q and D. Additionally, we define the set of all used generators

$$\mathcal{G}(\mathcal{A}) = \{G : \text{exist } q \in Q \text{ and } \sigma \in \Sigma \text{ such that } (q, \sigma, G) \in \Delta\}.$$

The semantics of a WDTA $\mathcal{A}$ is given as a run $r : (Q \times D)^* \to \mathcal{G}(\mathcal{A})$ on a tree $t : D^* \to \Sigma$ with similar requirements as a run for an alternating tree automata, namely

1. $(q_0, t(\epsilon), r(\epsilon)) \in \Delta$,
2. and for every $r((q_1, d_1) \dots (q_n, d_n)) = G$ there is $(q_n, t(d_1 \dots d_n), G) \in \Delta$.

Analogously to the semantics of PBAs or a strategy in a MDP a run induces a probability measure on $\mathcal{B}(Q \times D)$ by fixing the measure of cylinders by

$$\mu_r(\mathrm{cyl}((q_1, d_1) \dots (q_n, d_n))) = r(\epsilon)(q_1, d_1) \cdot \prod_{i=1}^{n-1} r((q_1, d_1) \dots (q_i, d_i))(q_{i+1}, d_{i+1})$$

for every $(q_1, d_1) \dots (q_n, d_n) \in (Q \times D)^*$ and uniquely extend this measure to $\mathcal{B}(Q \times D)$ (cp. Theorem 1). Naturally, the acceptance of r is defined by the measure on those paths accepting the condition in the state component, i.e. $\mathrm{Acc}\,[Q]$. In order to ensure the well-formedness of this defintion it is necessary to show the measurability of $\mathrm{Acc}\,[Q]$. We obtain this by the same argument as for Lemma 8 or respectively Corollary 1 by considering $\mathcal{B}(Q \times D)$.

Lemma 16. *For a Büchi-, Parity-, Rabin- or Muller-condition represented as F,* par*, R, $\mathcal{F}$ respectively, the sets* $\mathrm{Acc}\,[Q]\,(F), \mathrm{Acc}\,[Q]\,(\mathrm{par}), \mathrm{Acc}\,[Q]\,(R), \mathrm{Acc}\,[Q]\,(\mathcal{F})$ *are measureable in $\mathcal{B}(Q \times D)$.*

Proof. Given a set of states Q with an associated Büchi-condition F and a set of directions D we prove the measureability of $\mathrm{Acc}\,[Q]\,(F)$ by setting $Q' = Q \times D$ and $F' = \{(q, d) \in Q' \mid q \in F\}$. With Lemma 8 we obtain that $\mathrm{Acc}(F') \in \mathcal{B}(Q') = \mathcal{B}(Q \times D)$. It suffices to show $\mathrm{Acc}(F') = \mathrm{Acc}\,[Q]\,(F)$. For every non-empty $A = \{(p_1, b_1) \dots (p_k, b_k)\} \subseteq Q'$ we have $A \cap F' \neq \emptyset$ if and only if there is one i such that $(p_i, b_i) \in F'$ which holds by choice of F' if and only if $p_i \in F$ and therefore $A\,[Q] = \{p_1, \dots, p_k\} \cap F \neq \emptyset$. Then we have for every $\alpha = (q_1, d_1)\,(q_2, d_2) \dots \in \mathrm{Acc}(F')$ if and only if $A = \mathrm{Inf}(\alpha) \cap F' \neq \emptyset$ if and only if $A\,[Q] \cap F \neq \emptyset$ if and only if $q_1 q_2 \dots \in \mathrm{Acc}(F)$ if and only if $\alpha \in \mathrm{Acc}\,[Q]\,(F)$. The measureability for Muller-conditions (and therefore also for Parity- or Rabin-conditions) follows analogously to the proof of Corollary 1. $\square$

In the following, we focus on Parity-conditions for WDTAs and motivate this by the observation that Parity-conditions are sufficient to express all other conditions (as suggested in [CHS14, page 24:9, Proposition 6]).

Lemma 17. *For every WDTA $\mathcal{A}$ with a Büchi-, Rabin- or Muller-condition we can construct an equivalent WDTA $\mathcal{C}$ with a Parity-condition. This construction respects choicelessness, uni-directionality and uniform distributions.*

Proof. Firstly, we observe that Büchi- and Rabin-conditions can be expressed as Muller-conditions. Secondly, we construct for a WDTA

$$\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathcal{F} = \{F_1, \ldots, F_n\})$$

an equivalent WDTA $\mathcal{C}$ with a Parity-condition making use of an LAR. Using an LAR increases the state space but preserves the structure, especially the weighting, of a run and thus yields the same measure of the set of accepted paths. We construct

$$\mathcal{C} = (\mathrm{LAR}(Q), \ell_0, D, \Sigma, \Delta', \mathrm{par})$$

where ℓ_0 is one arbitrary element of $\mathrm{LAR}(Q)_{[q_0]}$ and par is defined as before for LARs[4] (see proof of Theorem 8). For every transition $(q, \sigma, G) \in \Delta$ we define

$$G'_\ell(\ell', d) = \begin{cases} G_i(p, d) & \text{if } \ell' = \mathrm{update}(\ell, p), \\ 0 & \text{otherwise,} \end{cases}$$

and add

$$\{(\ell, \sigma, G'_\ell) : \text{for every } \ell \in \mathrm{LAR}(Q)\}$$

to Δ'. We fix a tree t and observe that every run r of $\mathcal{A}$ can be embedded in a run r' of $\mathcal{C}$ by an inductive construction: for $(q_0, t(\epsilon), r(\epsilon)) \in \Delta$ exists an associated $(\ell_0, t(\epsilon), G)$ and we set $r'(\epsilon) = G$. Notably, we can pick any element $(\ell, d) \in \mathrm{LAR}(Q) \times D$ with $G(\ell, d) > 0$ and project it to $(p, d) \in Q \times D$ such that $\mathrm{update}(\ell_0, p) = \ell$ and iterate this argument. This embeds r in r'. Moreover, we can observe that this allows to obtain for any run r' of $\mathcal{C}$ on t a run r of $\mathcal{A}$ on t by iteratively using the corresponding original transitions from which transitions in $\mathcal{C}$ are derived. Hence, we can relate by the inductive construction paths in r and r' in both directions such that

$$\begin{array}{cccccc} q_0 & (q_1, d_1) & (q_2, d_2) & (q_3, d_3) & \ldots \\ \ell_0 & (\ell_1, d_1) & (\ell_2, d_2) & (\ell_3, d_3) & \ldots \end{array}$$

with $\ell_i = \mathrm{update}(\ell_{i-1}, q_i)$ for all $i > 0$ and state that all other paths in the domain of r' have a measure of 0 by definition of the generators in $\mathcal{C}$. Specifically, we obtain

$$\mu_r(\mathrm{cyl}((q_1, d_1) \ldots (q_n, d_n))) = \mu_{r'}(\mathrm{cyl}((\ell_1, d_1) \ldots (\ell_n, d_n)))$$

for the bijected elements. This allows to conclude with Lemma 7

$$\mu_r(\mathrm{Acc}(\mathcal{F})) = \mu_{r'}(\mathrm{Acc}(\mathrm{par})),$$

which yields the claimed acceptance equivalence of $\mathcal{A}$ and $\mathcal{C}$. Also, it is easy to observe that this construction respects choicelessness, uni-directionality and uniform distribution of $\mathcal{A}$. □

[4]Recall that parities grow proportional with the size of the hit-set and are even for hit-sets that are part of $\mathcal{F}$ and odd otherwise.

We distinguish certain structural properties on WDTAs allowing us to categorize WDTAs, namely

Definition 23. ***Structural Properties***:
A WDTA $\mathcal{A}$ is called

choiceless if there is at most one transition for every pair $q \in Q, \sigma \in \Sigma$ in Δ,

uni-directional if for every clause there is at most one state send to every direction, i.e. for every $G \in \mathcal{G}_{\mathcal{A}}$ there is at most one $q \in Q$ such that $G(q,d) > 0$ for all $d \in D$. Intuitively, this means that the automaton explores every path with at most one state,

uniformly distributed if $\mathcal{A}$ is uni-directional and for a direction $d \in D$ every clause agrees on the weight that is sent down that direction. Thus, for all $d \in D$ holds that for all $G_1, G_2 \in \mathcal{G}_{\mathcal{A}}$ we have $G_1(q,d) = G_2(p,d)$ for the unique q,p for which $G_1(q,d) > 0$, respectively $G_2(p,d) > 0$ or $G_1(q,d) = G_2(p,d) = 0$ for all q,p. Thus, we can fix a probability distribution $B : D \to [0,1]$ such that for every $G \in \mathcal{G}_{\mathcal{A}}$ there is one $q \in Q$ with $B(d) = G(q,d)$. We call this distribution B the *blueprint* of $\mathcal{A}$.

This allows us to categorize the mainly examined tree automata in [CHS14] as uniformly distributed WDTAs over a fixed set of directions $D = \{0,1\}$ with a blueprint B such that

$$B(0) = B(1) = \frac{1}{2}.$$

Given these definitions, we explore the theory of WDTAs.

3.2.1. Closure Properties

In the following, we examine the closure properties of WDTAs. We provide constructions to obtain closure for union and intersection for WDTAs, regarding complementation we leave the closure property open. However, we obtain for choiceless WDTAs that there is no closure under complementation regarding almost-sure acceptance of Büchi-conditions as a simple corollary from the result to embed PBAs into WDTAs. Mainly, the presented results are obtained by re-iterating arguments for PBAs or tree automata. We examine these properties selectively, predominantly focusing on almost-surely accepting models because they prove most relevant for the examined use-cases later on.

Unsuprisingly, the non-determinism of the model induces closure under union for certain classes in a straightforward manner (as suggested by [CHS14, Proposition 14]).

Proposition 5 (Union - uni-directional, unrestricted)**.** *Uni-directional and unrestricted WDTAs are closed under union for either positive and almost-sure acceptance. Moreover, uniformly distributed WDTAs with a common blueprint are closed under union.*

Proof. For two automata $\mathcal{A}_1 = (Q_1, q_1^0, D, \Sigma, \Delta_1, \mathrm{par}_1)$ and $\mathcal{A}_2 = (Q_2, q_2^0, D, \Sigma, \Delta_2, \mathrm{par}_2)$ we can w.l.o.g. assume that $Q_1 \cap Q_2 = \emptyset$. The union WDTA guesses in the very first transition which of the automata is checked for the tree:

$$\mathcal{C} = \left(Q_1 \cup Q_2 \uplus \left\{q^i\right\}, q^i, D, \Sigma, \Delta' = \Delta_1' \cup \Delta_2' \cup \Delta^i, \mathrm{par}_\cup\right)$$

where q^i is a new state and the transitions separate in

$$\Delta_j' = \{(q, \sigma, G\left[Q_{3-j} \times D \mapsto 0\right]) : (q, \sigma, G) \in \Delta_j\} \text{ for } j = 1, 2$$

and

$$\Delta^i = \left\{\left(q^i, \sigma, G\left[Q_j \times D \mapsto 0\right]\right) : \text{for } j = 1, 2 \text{ and every } (q_0, \sigma, G) \in \Delta_{3-j}\right\}.$$

Each transition in Δ_1' or Δ_2' mirrors one original transition in either $\mathcal{A}_1$ or $\mathcal{A}_2$ respectively, but its generator is extended to the necessary complete domain of $Q \times D$ for formal reasons. Setting all transitions for states of the other automaton to zero ensures that it precisely simulates the original transition. On the other hand we have Δ^i as the union of all initial transitions from Δ_1 and Δ_2. Again the generators are defined for the complete domain $(Q_1 \cup Q_2) \times D$ but effectively each generator mirrors a generator of a transition from either Δ_1 or Δ_2. Finally, we define

$$\mathrm{par}_\cup(q) = \begin{cases} \mathrm{par}_i & \text{if } q \in Q_i \text{ for } i = 1, 2, \\ \min\left(\mathrm{par}_1(Q) \cup \mathrm{par}_2(P)\right). & \text{if } q = q^i. \end{cases}$$

For any run r of $\mathcal{C}$ the element $r(\epsilon)$ corresponds with a transition in Δ_j for j being either 1 or 2. By the definition of the transitions in Δ' we can state that $\mu_r(\mathrm{cyl}(u)) = 0$ for any $u \notin (Q_j \times D)^*$. More importantly, for any run r of $\mathcal{C}$ on t we find a run r' of $\mathcal{A}$ on t (and vice verca) such that $\mu_r(\mathrm{cyl}(u)) = \mu_{r'}(\mathrm{cyl}(u))$ for any $u \in (Q_i \times D)^*$ and therefore $\mu_r(A) = \mu_{r'}(A)$ for all $A \in \mathcal{B}(Q_j \times D)$ and $\mu_r(B) = 0$ for all $B \in \mathcal{B}((Q_1 \cup Q_2) \times D) \setminus \mathcal{B}(Q_j \times D)$. Hence, all runs in $\mathcal{C}$ can be bijected to all runs of either $\mathcal{A}_1$ or $\mathcal{A}_2$. By definition of $\mathrm{par}_\cup$ we have $\mathrm{Acc}(\mathrm{par}_\cup) \cap \mathcal{B}(Q_j \times D) = \mathrm{Acc}(\mathrm{par}_j)$ for $j = 1, 2$ and therefore $\mathcal{C}$ accepts the trees which are accepted by either $\mathcal{A}_1$ or $\mathcal{A}_2$. This construction respects uni-directionality and even a common blueprint of $\mathcal{A}_1$ and $\mathcal{A}_2$ since transition are simply mirrored. $\square$

We can use the aforementioned equivalence of uniformly distributed WDTAs with the model introduced in [CHS14] to carry over the follwing results:

Proposition 6. *[CHS14, Proposition 14, Proposition 15] Uniformly distributed WDTAs with common blueprint and almost-sure acceptance are closed under intersection but not under complement.*

In the following, we obtain closure under intersection for WDTAs with almost-sure acceptance and for uniformly distributed WDTAs with common blueprint and positive acceptance measure. For almost-sure accepting WDTAs we use the same construction as for the union of positively accepting PBAs or the intersection of alternating tree automata respectively. Namely, the intersection operator separates in the first transition the runs of both WDTA but tunes down both runs with a probability of $\frac{1}{2}$ each. Nevertheless, both of these runs individually need an acceptance measure of 1 to render the new run almost-surely accepted.

Proposition 7. *The class of tree languages that can be recognized by WDTAs with almost-sure acceptance is closed under intersection.*

Proof. We introduce the intersection automaton as follows

Definition 24.
For

$$\mathcal{A}_1 = \left(Q_1, q_0^1, D, \Sigma, \Delta_1, \mathrm{par}_1\right) \text{ and } \mathcal{A}_2 = \left(Q_2, q_0^2, D, \Sigma, \Delta_2, \mathrm{par}_2\right)$$

(again assuming w.l.o.g. that $Q_1 \cap Q_2 = \emptyset$) we define

$$\mathcal{C} = \left(Q_1 \cup Q_2 \uplus \{q_\cap\}, q_\cap, D, \Sigma, \Delta_1' \cup \Delta_2' \cup \Delta_\cap, \mathrm{par}_\cap\right)$$

with

$$\Delta_i' = \{(q, \sigma, G\left[Q_{3-i} \cup \{q_\cap\} \times D \mapsto 0\right]) : (q, \sigma, G) \in \Delta_i\} \text{ for } i = 1, 2,$$

and

$$\Delta_\cap = \left\{\left(q_\cap, \sigma, G_{G_2}^{G_1}\right) : \text{for all } \left(q_0^i, \sigma, G_i\right) \in \Delta_i, i = 1, 2\right\}$$

where

$$G_{G_2}^{G_1}(q, d) = \begin{cases} \frac{1}{2} \cdot G_i(q, d) & \text{if } q \in Q_i \text{ for } i = 1, 2, \\ 0 & \text{otherwise.} \end{cases}$$

Regarding the acceptance condition we choose to use every individual acceptance condition and render $q_\cap$ irrelevant by setting

$$\mathrm{par}_\cap(q) = \begin{cases} \mathrm{par}_i(q) & \text{if } q \in Q_i \\ \min\left\{\mathrm{par}_1(q_0^1), \mathrm{par}_2(q_0^2)\right\} & \text{otherwise.} \end{cases}$$

Analogously to the proof of Corollary 4, we can identify two independent runs on the tree. The main observation is that no weighting is exchanged between the Q_1 and Q_2 part of any run. Let r be a run of $\mathcal{C}$ on a tree t, then for all $u \in ((Q_1 \cup Q_2) \times D)^* \setminus ((Q_1 \times D)^* \cup (Q_2 \times D)^*)$ we have $\mu_r(\mathrm{cyl}(u)) = 0$. Additionally, we can find two runs r_1 and r_2 for $\mathcal{A}_1$ and $\mathcal{A}_2$ respectively on t such that for every $u \in (Q_1 \times D)^+$ and $w \in (Q_2 \times D)^+$ holds $r_1(u) = r(u)$ and $r_2(w) = r(w)$ since Δ_i' mirrors Δ_i for $i = 1, 2$ respectively (note that also for every two such runs r_1 and r_2 a run r in $\mathcal{C}$ exists). By construction of $\Delta_\cap$ we can additionally state that $\mu_r(\mathrm{cyl}(u)) = \frac{1}{2} \cdot \mu_{r_1}(\mathrm{cyl}(u))$ and $\mu_r(\mathrm{cyl}(u)) = \frac{1}{2} \cdot \mu_{r_2}(\mathrm{cyl}(w))$ respectively. Using the fact that $\mathrm{Acc}(\mathrm{par}_\cap)$ coincides with

$\mathrm{Acc}(\mathrm{par}_1)$ and $\mathrm{Acc}(\mathrm{par}_2)$ on $(Q_1 \times D)^\omega$ and $(Q_2 \times D)^\omega$ allows us to state that

$$\mu_r(\mathrm{Acc}(\mathrm{par}_\cap)) = \frac{1}{2} \cdot \mu_{r_1}(\mathrm{Acc}(\mathrm{par}_1)) + \frac{1}{2} \cdot \mu_{r_2}(\mathrm{Acc}(\mathrm{par}_2)).$$

Hence, $\mu_r(\mathrm{Acc}(\mathrm{par}_\cap)) = 1$ if and only if $\mu_{r_1}(\mathrm{Acc}(\mathrm{par}_1)) = 1$ and $\mu_{r_2}(\mathrm{Acc}(\mathrm{par}_2)) = 1$ which ensures that $\mathcal{C}$ indeed accepts the intersection of $\mathcal{A}_1$ and $\mathcal{A}_2$. Note, that this construction respects choicelessness of the original automata. □

Additionally, we can observe in the presented construction that $\mu_r(\mathrm{Acc}(\mathrm{par}_\cap)) > 0$ if and only if $\mu_{r_1}(\mathrm{Acc}(\mathrm{par}_1)) > 0$ or $\mu_{r_2}(\mathrm{Acc}(\mathrm{par}_2)) > 0$. Therefore, we can use $\mathcal{C}$ as alternative construction to ensure the closure of WDTAs with positive acceptance.

Corollary 5. *The class of languages that can be recognized by choiceless WDTAs with almost-sure (positive) acceptance is closed under intersection (union).*

Actually, this results hints at the duality between choiceless WDTAs with positive acceptance and almost-sure acceptance which is expressd in

Proposition 8. *For every choiceless WDTAs-skeleton (which are the structural components but not an acceptance condition) $\mathcal{S} = (Q, q_0, D, \Sigma, \Delta)$ the WDTA $\mathcal{A} = (\mathcal{S}, \mathrm{par})$ with positive (almost-sure) acceptance is dual to the WDTA $\mathcal{C} = (\mathcal{S}, \mathrm{par} + 1)$ with almost-sure (positive) acceptance.*

Proof. We already argued in the proof of Theorem 11 that increasing all parities by one exchanges accepting and non-accepting paths. Therefore,

$$(Q \times D)^\omega = \mathrm{Acc}\,[Q]\,(\mathrm{par}) \cup \mathrm{Acc}\,[Q]\,(\mathrm{par}+1) \text{ with } \mathrm{Acc}\,[Q]\,(\mathrm{par}) \cap \mathrm{Acc}\,[Q]\,(\mathrm{par}+1) = \emptyset.$$

Hence, for the unique run r_t of $\mathcal{S}$ on every t we obtain that $\mu_{r_t}(\mathrm{Acc}\,[Q]\,(\mathrm{par})) = 1 - \mu_{r_t}(\mathrm{Acc}\,[Q]\,(\mathrm{par}+1)$ and therefore we get that $\mathcal{A}$ accepts t almost-surely (positively) if and only if $\mathcal{C}$ does not accept t positively (almost-surely). □

3.3. Modelling in Weighted Descent Tree Automata

We want to further motivate the introduction of WDTAs by showing their strength to model the behavior of other probabilistic models.

3.3.1. Word Automata

Initially, we consider PBAs. By reducing the directions to a singelton set, a tree degenerates to a single path. W.l.o.g. we consider $D = \{0\}$ whenever D is said to be a singleton direction set. To ease the following formulations we consider the set of words $\alpha \in \Sigma^\omega$ equivalent to the set of all functions $t : \{0\}^* \to \Sigma$. Formally, we biject these sets by

$$\alpha_0\alpha_1 \cdots \mapsto t_{\alpha_1\alpha_2\ldots} \text{ with } t_{\alpha_1\alpha_2\ldots}(u) = \alpha_{|u|} \text{ for all } u \in D^*$$

and interchangably use α for both words and unary trees.

Lemma 18. *For every PBA $\mathcal{P}$ exists a choiceless Büchi-WDTA $\mathcal{A}$ with*

$$\mathcal{P} = (Q, \Sigma, \delta, q_0, F) \text{ and } \mathcal{A} = (Q, q_0, \{0\}, \Sigma, \Delta, F)$$

such that

$$(Q^\omega, \mathcal{B}(Q), \mu_\alpha) \text{ and } (Q^\omega, \mathcal{B}(Q \times \{0\}), \mu_r)$$

are isomorphic for the unique run r of $\mathcal{A}$ on α. The isomorphism bijects the acceptance conditions of $\mathcal{P}$ and $\mathcal{A}$.

Moreover, for every choiceless Büchi-WDTA $\mathcal{A}$ exists a PBA $\mathcal{P}$ with

$$\mathcal{A} = (Q, q_0, \{0\}, \Sigma, \Delta, F) \text{ and } \mathcal{P} = (Q, \Sigma, \delta, q_0, F)$$

such that

$$(Q^\omega, \mathcal{B}(Q \times \{0\}), \mu_r) \text{ and } (Q^\omega, \mathcal{B}(Q), \mu_\alpha)$$

are isomorphic for the unique run r of $\mathcal{A}$ on α. The isomorphism bijects the acceptance conditions of $\mathcal{P}$ and $\mathcal{A}$.

Proof. For a PBA $\mathcal{P} = (Q, \Sigma, \delta, q_0, F)$ we define WDTA $\mathcal{A} = (Q, q_0, \{0\}, \Sigma, \Delta, F)$ with

$$\Delta = \left\{ \left(q, \sigma, G_q^\sigma\right) : \text{for every } q \in Q, \sigma \in \Sigma \right\} \text{ with } G_q^\sigma(p, 0) = \delta(q, \sigma, p).$$

Fix one $\alpha \in \Sigma^\omega$ and the unique run r of $\mathcal{A}$ on α. Defining $f : Q \to \{0\}$ with $f(q) = 0$ for all $q \in Q$ allows to use Lemma 6 to obain for $(Q^\omega, \mathcal{B}(Q), \mu_\alpha)$ an isomorphic probability space $\left(\text{lift}_f(Q^\omega), \mathcal{B}(Q \times \{0\})_{|\text{lift}_f(Q^\omega)}, \mu'\right)$ with $\mu' = \mu \circ \text{lift}_f^{-1}$. We observe that for every $u = q_1 \ldots q_n$ holds

$$\begin{aligned}
\mu_\alpha(\text{cyl}(u)) &= \delta(q_0, \alpha_0, q_1) \cdot \delta(q_1, \alpha_1, q_2) \ldots \delta(q_{n-1}, \alpha_{n-1}, q_n) \\
&= G_{q_0}^{\alpha_0}(q_1, 0) \cdot \prod_{i=1}^{n-1} G_{q_i}^{\alpha_i}(q_{i+1}, 0) \\
&= \mu_r(\text{lift}_f(\text{cyl}(u))) = (\mu_r \circ \text{lift}_f)(\text{cyl}(u)).
\end{aligned}$$

This implies that

$$((Q \times \{0\}), \mathcal{B}(Q \times \{0\}), \mu_r) \text{ and } \left(\text{lift}_f(Q^\omega), \mathcal{B}(Q \times \{0\})_{|\text{lift}_f(Q^\omega)}, \mu'\right)$$

coincide. This gives the announced isomorphism by the function lift_f. Moreover, it holds that $\text{lift}_f(\text{Acc}(F)) = \text{Acc}[Q](F)$.

For the opposite direction, we fix a choiceless Büchi-WDTA

$$\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, F).$$

By the choicelessness of $\mathcal{A}$ there is exactly one $(q, \sigma, G) \in \Delta$ for every pair $q \in Q$ and $\sigma \in \Sigma$. We denote with G_q^σ this unique G. We construct a PBA

$$\mathcal{P} = (Q, \Sigma, \delta, q_0, F)$$

with $\delta(q, \sigma, p) = G_q^\sigma(p, 0)$. The reasoning from above immediately translates. $\square$

As a consequence of this construction, we obtain

Theorem 30. *The class of languages that can be recognized by positively (almost-surely) accepting PBAs and positively (almost-surely) accepting Büchi-WDTAs coincide.*

Proof. Lemma 18 immediately induces the claim. □

The construction in Lemma 18 can be expanded to all possible acceptance conditions.

Corollary 6. *The statement of Lemma 18 also holds if the Büchi-conditions for the probabilistic word automaton and choiceless WDTA is replaced with a Rabin-, Parity- or Muller-condition.*

Proof. It suffices to observe that

$$\mathrm{lift}_f(\mathrm{Acc}(\mathcal{F})) = \mathrm{Acc}\,[Q]\,(\mathcal{F})$$

holds for a Muller-condition $\mathcal{F}$, since Rabin- and Parity-conditions can be expressed as Muller-conditions. □

By the combination of this result and the sufficiency of Parity-conditions to express all other acceptance conditions for WDTAs (Lemma 17) provides the basis to infer the equivalence of "strong" acceptance conditions for PBAs (Theorem 16) as a corollary. Additionally, considering the equivalence of choiceless Büchi-WDTAs over words with PBAs allows to use Proposition 2 to deduce

Proposition 9. *The class of languages that can be recognized by choiceless Büchi-WDTAs with almost-sure acceptance is not closed under complementation.*

Moreover, we can use uni-directional WDTAs over words to emulate any ω-regular word automaton. Fixing a singleton direction set D simply allows for one possible probability distribution for the associated blueprint B of a uniformly distributed WDTAs, i.e. $B(0) = 1$. Moreover, this implies that acceptance is dependent on precisely one path in the run. This path is accepted if the state components satisfy an ω-regular condition. These observations naturally lead to the equivalence of these automata models, i.e.

Theorem 31. *The class of languages of unary trees recognizable with uni-directional WDTAs with either positive or almost-sure acceptance coincides with the ω-regular languages.*

Proof. For every uni-directional WDTAs over a singleton direction set there is excatly one possible probability distribution such that there is only one state with weight for this direction, i.e. the one where a state is weighted with probability 1. Therefore uni-directionality on WDTAs with a single direction set implies an uniform distribution with blueprint B such that $B(0) = 1$. We can encode every viable generator G by the one state q such that $G(q,0)=1$. This allows us to define for every uniformly distributed WDTA

$$\mathcal{A}=(Q,q_0,\{0\},\Sigma,\Delta,\mathrm{par})$$

an equivalent NPA

$$\mathcal{P}=(Q,q_0,\Sigma,\Delta',\mathrm{par})$$

with

$$\Delta'=\{(q,\sigma,p):\text{there is }(q,\sigma,G)\in\Delta\text{ such that }G(p,0)=1\}.$$

Any run r of $\mathcal{A}$ on a tree $t:\{0\}^*\to\Sigma$ induces a Dirac measure μ_r on $\mathcal{B}(Q\times D)$ for one path $\alpha_r\in(Q\times\{0\})^\omega$. Therefore, $\mu_r(\mathrm{Acc}\,[Q]\,(\mathrm{par}))=1$ if and only if $\alpha_r\in\mathrm{Acc}\,[Q]\,(\mathrm{par})$ which happens if and only if $\alpha_r\in\mathrm{Acc}(\mathrm{par})$. Note here, that this Dirac measure renders positive and almost-sure acceptance equivalent. Also, by the construction of Δ' this state sequence is a viable run of $\mathcal{P}$.

On the other hand, consider one run $\beta=\beta_1\beta_2\cdots\in Q^\omega$ of $\mathcal{P}$ on a word α. By construction of Δ' we have a run r_β of $\mathcal{A}$ on α which induces a Dirac measure on the path $(\beta_1,0),(\beta_2,0)\ldots$. Hence, r_β is almost-surely or positively accepting if and only if $\beta\in\mathrm{Acc}(\mathrm{par})$ which happens if and only if α is accepted by $\mathcal{P}$. Hence, $\mathcal{P}$ and $\mathcal{A}$ accept the same language.

For the converse argument, we construct for a given NPA

$$\mathcal{P}=(Q,\Sigma,q_0,\Delta,\mathrm{par})$$

the following WDTA

$$\mathcal{A}=(Q,q_0,\{0\},\Sigma,\Delta',\mathrm{par})$$

with

$$\Delta'=\{(q,\sigma,G_p):(q,\sigma,p)\in\Delta\}\text{ and }G_p(p',0)=\begin{cases}1 & \text{if }p=p',\\ 0 & \text{otherwise.}\end{cases}$$

Every run of $\mathcal{A}$ on α induces a Dirac measure $\mu_{(\beta_1,0)(\beta_2,0)\ldots}$ on $\mathcal{B}(Q\times\{0\})$. By construction of Δ', this $\beta_1\beta_2\cdots\in Q^\omega$ is a viable run of $\mathcal{P}$ on α. Moreover, $\beta_1\beta_2\cdots\in\mathrm{Acc}(\mathrm{par})$ if and only if $(\beta_1,0)\,(\beta_2,0)\cdots\in\mathrm{Acc}\,[Q]\,(\mathrm{par})$. The claim follows. □

Hence, the determinisation of ω-regular automata (Theorem 8) translates for uni-directional WDTA with singleton direction sets.

Corollary 7. *For every uni-directional WDTA over a singleton direction set exists an equivalent choiceless uni-directional WDTA.*

Proof. For a given uni-directional WDTA $\mathcal{A}$ we use Theorem 31 to obtain an equivalent NPA $\mathcal{P}$. Theorem 8 allows to construct an equivalent DPA $\mathcal{P}_D$ for $\mathcal{P}$. Applying the construction from Theorem 31 again results in an equivalent uni-directional WDTA. By examining the construction of the proof of Theorem 31, we see that the resulting WDTA is indeed choiceless. □

Modelling probabilistic word automata entails immediately undecideability results: Theorem 17 allows to deduce the undecideability of the emptiness problem even for choiceless WDTAs with positive acceptance of an associated Büchi-condition (and therefore also for Rabin-, Parity- and Muller-conditions). By Corollary 2 we find undecidability results for the emptiness problem for almost-sure accepting choiceless WDTAs for Rabin-, Parity- and Muller-conditions.

Corollary 8. *The emptiness problem (even for choiceless) WDTAs is undecideable for positive acceptance of Büchi-, Rabin-, Parity- and Muller-conditions and almost-sure acceptance of Rabin-, Parity- and Muller-conditions.*

For PBAs we know that more expressive acceptance conditions allows to move from positive acceptance to almost-sure acceptance. Unfortunately, this does not translate to choiceless WDTAs. In order to show this we derive a few helpful lemmas for reasoning about runs for WDTAs.

Lemma 19. *For any run r of a WDTA $\mathcal{A}$ with a Parity-condition* par *on a tree t we can consider subruns r_u for $u \in (Q \times D)^*$ with $r_u(w) = r(u \cdot w)$ for $w \in (Q \times D)^*$ and state for any $\{v_1, \ldots, v_n\} \subseteq (Q \times D)^*$ such that* $\mathrm{cyl}(v_i)$ *for $1 \leq i \leq n$ partitions $(Q \times D)^\omega$*

$$\mu_r(\mathrm{Acc}\,[Q]\,(\mathrm{par})) = \sum_{1 \leq i \leq n} \mu_r(\mathrm{cyl}(v_i)) \cdot \mu_{r_{v_i}}(\mathrm{Acc}\,[Q]\,(\mathrm{par})).$$

Proof. Since we partition $(Q \times D)^\omega$ into $\mathrm{cyl}(v_i)$ for $1 \leq i \leq n$ we can also partition $\mathrm{Acc}\,[Q]\,(\mathrm{par})$ into n sets $\mathrm{Acc}_i = \mathrm{Acc}\,[Q]\,(\mathrm{par}) \cap \mathrm{cyl}(v_i)$ for $1 \leq i \leq n$. Therefore, we state using the observation that the observation Acc_i corresponds to the independent observations to end up in $\mathrm{cyl}(v_i)$ and satisfy the Parity-condition

$$\mu_r(\mathrm{Acc}\,[Q]\,(\mathrm{par})) = \sum_{1 \leq i \leq n} \mu_r(\mathrm{Acc}_i) = \sum_{1 \leq i \leq n} \mu_r(\mathrm{cyl}(v_i)) \cdot \mu_{r_{v_i}}(\mathrm{Acc}\,[Q]\,(\mathrm{par})).$$

□

This allows to deduce the following observation which is also used in the proof to Lemma 13:

Lemma 20. *A run r of a WDTA $\mathcal{A}$ with a Parity-condition* par *on a tree t is almost-surely accepted if and only if for all $u \in (Q \times D)^+$ with $\mu_r(\mathrm{cyl}(u)) > 0$ holds*

$$\mu_{r_u}(\mathrm{Acc}\,[Q]\,(\mathrm{par})) = 1.$$

Proof. First consider that $\mu_{r_u}(\mathrm{Acc}\,[Q]\,(\mathrm{par})) = 1$ for all $u \in (Q \times D)^+$, then choose an arbitrary set $\{u_1, \ldots, u_n\} \subseteq (Q \times D)^+$ such that $\mathrm{cyl}(u_i)$ for $1 \leq i \leq n$ partitions $(Q \times D)^\omega$. Therefore, we see that $\mu_r(\mathrm{cyl}(\cdot))$ describes a probability measure on $\{u_1, \ldots, u_n\}$. This implies that

$$\mu_r(\mathrm{Acc}\,[Q]\,(\mathrm{par})) = \sum_{1 \leq i \leq n} \mu_r(\mathrm{cyl}(u_i)) \cdot \underbrace{\mu_{r_{u_i}}(\mathrm{Acc}\,[Q]\,(\mathrm{par}))}_{=1} = \sum_{1 \leq i \leq n} \mu_r(\mathrm{cyl}(u_i)) = 1.$$

On the other hand, consider that there is one $u = (q_1, d_1) \ldots (q_n, d_n) \in (Q \times D)^+$ with $\mu_r(\mathrm{cyl}(u)) > 0$ and $\mu_{r_u}(\mathrm{Acc}\,[Q]\,(\mathrm{par})) < 1$. We construct a set

$$R = \bigcup_{0 \leq i < n} \left\{ (q_1, d_1) \ldots (q_i, d_i) \cdot (p, d) : \text{for all } p \in Q, d \in D \text{ with } \begin{array}{ll} \text{either} & p \neq q_{i+1} \\ \text{or} & d \neq d_{i+1} \end{array} \right\}.$$

Notably, $\mathrm{cyl}(a)$ for $a \in R \cup \{u\})$ partitions $(Q \times D)^\omega$ and again $\mu_r(\mathrm{cyl}(\cdot))$ is a probability distribution on $R \cup \{u\}$. Therefore, we obtain

$$\begin{aligned} \mu_r(\mathrm{Acc}\,[Q]\,(\mathrm{par})) = &\sum_{w \in R} \mu_r(\mathrm{cyl}(w)) \cdot \mu_{r_w}(\mathrm{Acc}\,[Q]\,(\mathrm{par})) \\ &+ \underbrace{\mu_r(\mathrm{cyl}(u))}_{>0} \underbrace{\mu_{r_w}(\mathrm{Acc}\,[Q]\,(\mathrm{par}))}_{<1} < 1. \end{aligned}$$

□

Using these insights allows us to deduce

Proposition 10. *There exists a language that is accepted by a choiceless WDTA with a positive acceptance of a Büchi-condition but which is not accepted by any choiceless WDTA with an almost-sure accepted Parity-condition.*

Proof. We define $\mathcal{L}_{\exists a}$ as the language of all trees with directions $\{0, 1\}$ over the alphabet $\{a, b\}$ that do contain an a. First, we show that this language can be accepted by a choiceless Büchi WDTA with positive acceptance. Therefore, we construct a WDTA with two states q_a, q_b where q_b is the initial state. The final state is q_a and we distribute uniformely over both branches q_b until an a occurs where the automaton distributes uniformely the state q_a regardless of the read letter, thus q_a is a positive sink-state. Any tree that is accepted does necessarily contain an a since it is the only chance to produce an acceptance subtree and conversely, if a tree does contain an a there is a finite path of length n towards the subtree rooted at this occurence of a, hence in the run of the automaton on this tree there is an accepted subtree which is reached with probability $(\frac{1}{2})^n > 0$ rendering the run accepting for this tree.

On the other hand we show that there is no choiceless WDTA with an almost-sure Parity-condition par that precisely accepts $\mathcal{L}_{\exists a}$. In the following we consider three trees t_0, t_1, t_b where every letter in t_i is b except for $t_i(i) = a$ for $i = 0, 1$ and t_b only contains *b.* Assume there is a choiceless WDTA $\mathcal{A}$ with an almost-sure acceptance of a Parity-condition to accept $\mathcal{L}_{\exists a}$. We fix the unique runs of $\mathcal{A}$ on t_0, t_1, t_b as r_0, r_1, r_b respectively. Since $\mathcal{A}$ is choiceless and $t_0(\epsilon) = t_1(\epsilon) = t_b(\epsilon) = b$ we get $r^0(\epsilon) = r^1(\epsilon) = r^b(\epsilon)$. We can fix T_0 and T_1 with $T_i = \{(q, i) : \text{for } q \in Q\}$ for $i = 1, 2$. Observably, $\text{cyl}(T_0) \cup \text{cyl}(T_1)$ parititons $(Q \times D)^\omega$. Since r^i is almost-surely accepting we use Lemma 20 to deduce that $\mu_{r^i_u}(\text{Acc}\,[Q]\,(\text{par})) = 1$ for all $u \in T_{1-i}$ $(i = 1, 2)$. Therefore, we get with Lemma 19

$$\begin{aligned}
\mu_{r^b}(\text{Acc}\,[Q]\,(\text{par})) &= \sum_{v \in T_0} \mu_{r^b}(\text{cyl}(v)) \cdot \mu_{r^b_v}(\text{Acc}\,[Q]\,(\text{par})) \\
&\quad + \sum_{v \in T_0} \mu_{r^b}(\text{cyl}(v)) \cdot \mu_{r^b_v}(\text{Acc}\,[Q]\,(\text{par})) \\
&= \sum_{\substack{v \in T_0 \\ \text{s.t. } \mu_{r^b_u}(\text{cyl}(v) > 0)}} \mu_{r^b}(\text{cyl}(v)) \cdot \overbrace{\mu_{r^b_v}(\text{Acc}\,[Q]\,(\text{par}))}^{=\mu_{r^1_v}(\dots)=1} \\
&\quad + \sum_{\substack{v \in T_1 \\ \text{s.t. } \mu_{r^b_u}(\text{cyl}(v) > 0)}} \mu_{r^b}(\text{cyl}(v)) \cdot \underbrace{\mu_{r^b_v}(\text{Acc}\,[Q]\,(\text{par}))}_{=\mu_{r^0_v}(\dots)=1} \\
&= 1.
\end{aligned}$$

Thus, r^b is an accepting run although $t_b \notin \mathcal{L}_{\exists a}$. □

This directly entails

Corollary 9. *The class of languages recognizable with WDTAs almost-sure acceptance strictly contains the class of languages recognizable with choiceless WDTAs with almost-sure acceptance.*

Proof. It is trivial to observe that unrestricted WDTAs includes the class of choiceless WDTAs. For the reverse observation it suffices to show that an unrestricted WDTA is capable to recognize $\mathcal{L}_{\exists a}$. This is rather straightforward by using two states q_0, q_a where q_0 is the initial state and changes into the accepting q_a after an a is read and distributes this q_a uniformly everywhere. On the other hand q_0 is used to search for the occurence of a by using Dirac distributions in two transitions which sent q_0 either in direction 0 or 1 and the non-determinism to decide which transition to take. □

For PBAs we already established with Lemma 11 that the local probabilities matter for the accepted language. This translates to WDTAs by the following example:

Example 10. [CHS14, Example 7] Let $\mathcal{L}$ be the set of all binary trees over $\{a, b\}$ such that under a uniform probability distribution of successors the set of those branches which contain infinitely many a has measure 1. We can accept $\mathcal{L}$ with a deterministic uniformly distributed WDTA with Büchi-condition and almost-sure acceptance

$$\mathcal{A} = (\{q_a, q_b\}, q_s, \{0,1\}, \{a,b\}, \Delta, \{q_a\})$$

with

$$\Delta = \{(q_\sigma, \sigma, G_\sigma) : \text{for every } \sigma \in \Sigma\} \text{ and } G_\sigma(q,d) = \begin{cases} \frac{1}{2} & \text{if } q = q_\sigma \text{ and both } d, \\ 0 & \text{otherwise.} \end{cases}$$

The argument that this automaton accepts $\mathcal{L}$ is straightforward since only those paths that contain infinitely often an a do have infinitely often a weighted occurence of q_a; thus, the run of $\mathcal{A}$ mirrors the input tree t and the language $\mathcal{L}$ is defined analogously to the semantics of WDTAs. Note that $\mathcal{A}$ respects a blueprint $B(0) = B(1) = \frac{1}{2}$ and can therefore be considered uniformly distributed.

We additionally introduce

Proposition 11. *[CHS14, Proposition 11] For two reals $0 < p < q < 1$ we define $\mathcal{A}_p$ and $\mathcal{A}_q$ analogously to $\mathcal{A}$ from Example 10 but with blueprint $B(0) = p, B(1) = 1 - p$ and $B(0) = q, B(1) = 1 - q$ respectively. Then, there is a tree t such that for the unique runs r_p and r_q of $\mathcal{A}_p$ and $\mathcal{A}_q$ respectively on t holds that $\mu_{r_p}(\mathrm{Acc}(F)) = 0$ while $\mu_{r_q}(\mathrm{Acc}(F)) = 1$.*

3.3.2. Probabilistic Weighted Automata

Recall, that PBAs are considered to be NBAs where the non-determinism is resolved using a probability distribution. This is conceptually slightly different to the concept of WDTAs which still incorporate a non-deterministic choice of transitions to construct a run. This run in turn uses a weighting along its paths. This concept is modelled using probability distributions but is not necessarily understood as a probabilistic choice. We want to discuss a concept introduced in [CHS14, Chapter 4] where the non-determinism of choosing a transition is resolved by probability distributions. This is conceptually closer to the idea of PBAs. Nevertheless, we still use a weighting within the runs itself, but in a more restricted fashion, namely by relying on uniformly distributed weighting only. This decouples the probability of a movement through the individual runs and the probability of the construction of one particular run itself. This allows for an easier mathematical analysis.

We begin by defining the class of Probabilistic Weighted Automaton (PWA) analogously to [CHS14, Definition 4.1.1]

Definition 25. ***Probabilistic Weighted Automata***:
Fix a finite alphabet Σ, a finite set of directions D and a finite set of states Q. We fix a probability distribution $B : D \to [0,1]$ which weights the movement through

any run. The possible transitions are defined as a non-empty set

$$\Delta = \{\tau_1, \dots, \tau_n\}$$

where $\tau_i : D \to Q$ for $1 \leq i \leq n$. The transitions are "chosen" by a function $\delta : Q \times \Sigma \times \Delta \to [0,1]$ such that $\delta(q, \sigma, \cdot)$ is a probability distribution over Δ. Concludingly, with a Büchi-condition $F \subseteq Q$ we obtain PWA

$$\mathcal{A} = (Q, q_0, D, \Sigma, B, \Delta, \delta, F)\,.$$

One run of such an automaton $\mathcal{A}$ on a tree $t : D \to \Sigma$ differs from runs for WDTAs since the run does not need to encode the weighting of paths through the run. Nevertheless, forgetting about the distribution δ for now, we can use the formalism of WDTAs to define the semantics of one run by defining for every pair $q \in Q$ and $\sigma \in \Sigma$ a set of transitions of the form (q, σ, G_τ) for every τ with $\delta(q, \sigma, \tau) > 0$ where

$$G_\tau(p, b) = \begin{cases} B(b) & \text{if } \tau(b) = p, \\ 0 & \text{otherwise.} \end{cases}$$

Although sufficient, this renders runs as objects of unneccessary complexity and we therefore rely on a more concise (but equivalent) formalism: We define one run $r : D^* \to Q$ such that $r(\epsilon) = q_0$. And, for every $w \in D^*$ such that $q = r(w)$ and $\sigma = t(w)$ there is one $\tau \in \Delta$ with $\delta(q, \sigma, \tau) > 0$ and $r(w \cdot d) = \tau(d)$ for all $d \in D$. As seen before, the distribution B induces a measure μ_B on the Borel-algebra $\mathcal{B}(D)$. We introduce the following renaming:

$$\text{Paths} = D^\omega$$

and obtain a probability space $(\text{Paths}, \mathcal{B}(D), \mu_B)$. For a run r we can collect the set of accepting paths regarding the Büchi-condition F as $\text{Acc}_F(r) = \{\alpha \in D^\omega \mid r(\alpha) \in \text{Acc}(F)\}$. We consider such a run r almost-surely accepting if $\mu_B(\text{Acc}_F(r)) = 1$ and positively accepting if $\mu_B(\text{Acc}_F(r)) > 0$.

Additionally, to examine the probabilities induced by δ we introduce a σ-algebra which is generated by partial runs, i.e. finite beginnings of runs. Therefore, we define a finite tree as a prefix-closed subset of finite words over directions.

Definition 26. ***Prefix-Tree***:
For a finite set of directions D we call a finite $T \subseteq D^*$ a prefix-tree if for every $w \in T$ and every prefix $v \sqsubseteq w$ holds $v \in T$. Additionally, we call a prefix-tree T *proper* if for all $w \in T$ and $d \in D$ such that $w \cdot d \in T$ holds $w \cdot d' \in T$ for all $d' \in D$. We define the *inner* nodes of a proper T as those nodes that do have an extension in T as $\text{inner}(T) = \{w \in T \mid w \cdot d \in T \text{ for all } d \in D\}$.

For a PWA $\mathcal{A}$, a tree $t : D \to \Sigma$ and a proper prefix-tree T we can consider a partial run $r : T \to Q$ of $\mathcal{A}$ on t by enforcing $r(\epsilon) = q_0$ and for every $u \in \text{inner}(T)$ holds that there is a τ with $\delta(r(u), t(u), \tau) > 0$ such that $r(u \cdot d) = \tau(d)$ for all $d \in D$. For one partial run $r : T \to Q$ it is straightforward to define the set of all runs $r' : D^* \to Q$ of $\mathcal{A}$ on t such that these runs agree on T with r, i.e. for all $w \in T$ holds $r(w) = r'(w)$. We call this set compRuns(r). Furthermore, we can gather the set of all possible runs of $\mathcal{A}$ on t in a set $\text{Runs}^t_{\mathcal{A}}$. Analogously, to the concept of cylinders over a set we can capture a σ-algebra $\mathcal{R}^t_{\mathcal{A}}$ over runs by using the sets compRuns(r) for every partial run r on t as generating sets (cp. [CHS14, Chapter 4]). For any partial run $r : T \to Q$ and any $u \in \text{inner}(T)$ we define τ_u as the unique τ such that $r(u \cdot d) = \tau(d)$ for all $d \in D$ and obtain a probability for the partial run and inherently a probability for all possible extensions compRuns(r), namely

$$\mu_t(\text{compRuns}(r)) = \prod_{w \in \text{inner}(T)} \delta(r(u), t(u), \tau_u).$$

This allows us, again using Theorem 1, to define a measureable space by using the unique extension of μ_t on $\mathcal{R}^t_{\mathcal{A}}$ (by abuse of notation also referred to as μ_t)

$$\left(\text{Runs}^t_{\mathcal{A}}, \mathcal{R}^t_{\mathcal{A}}, \mu_t\right).$$

In [CHS14] different semantics for this automaton model are discussed. For example

1. In terms of μ_t we can formulate the acceptance condition as

   $$\mu_t(\left\{r \in \text{Runs}^t_{\mathcal{A}} \mid r(\alpha) \in \text{Acc}(F) \text{ for every } \alpha \in D^\omega\right\}),$$

   for example by requiring a positive or almost-sure measure. Note, that this acceptance condition ignores the weighting of the individual runs.

2. On the other hand, we can define the acceptance condition in terms of of the μ_t measure of the set of positively accepting runs or almost-surely accepting runs. Here any combinations of measures for μ_t and μ_B are possible, but we restrict our arguments on the almost-sure μ_t measure of almost-sure μ_B accepting runs, i.e. to the condition

   $$\mu_t(\left\{r \in \text{Runs}^t_{\mathcal{A}} \mid \mu_B(\text{Acc}_F(r)) = 1\right\}) = 1.$$

In order to analyse this automata model, we define the function

$$f^t_{\mathcal{A}} : \text{Runs}^t_{\mathcal{A}} \times \text{Paths} \to [0,1] \text{ with } f_{\mathcal{A}}(r,p) = \begin{cases} 1 & \text{if } r(p) \in \text{Acc}_r(F), \\ 0 & \text{otherwise.} \end{cases}$$

Thus, $f^t_{\mathcal{A}}(r, \cdot)$ is a characteristic function of the set

$$\{\alpha \in D^\omega \mid r(\alpha) \in \text{Acc}_F(r)\}.$$

In the following we show the integrability of the function $f^t_{\mathcal{A}}$ in the product space of $(\text{Paths}, \mathcal{B}(D), \mu_B)$ and $\left(\text{Runs}^t_{\mathcal{A}}, \mathcal{R}_t, \mu_t\right)$. From this we deduce the well-foundness of our definition, i.e. the measureability of the set

$$\left\{r \in \text{Runs}^t_{\mathcal{A}} \mid \mu_B(\text{Acc}_F(r)) = 1\right\}.$$

Firstly, to ease the following argument we restrict our attention to partial runs that are "balanced"(as suggested by [CHS14, Remark 35]), i.e. all paths have the same length. This restriction is purely argumentativ which is ensured by

Lemma 21. *The σ-algebra that is generated by all balanced partial runs and the σ-algebra generated by all partial runs which are compatible with an underlying tree t are identical.*

Proof. Initially, we can observe that every balanced partial run that is compatible with t is a partial run itself and therefore, the σ-algebra generated by all partial runs compatible with t contains the σ-algebra that is generated by all balanced partial runs compatible with t.

On the other hand, we consider any partial run $r : T \to Q$ that is compatible with t. We fix $n = \max\{|u| : u \in T\}$. Furthermore, we define $T' = D^n \setminus T$ as the set of "missing" elements to balance T. Since there are only finitely many elements in Q and T', there are only finitely many functions $r' : T' \to Q$. We gather these in a set $R = \{r_1, \dots, r_k\}$. This implies that there are only finitely many balanced partial runs $s_1, \dots, s_k$ with $s_i : D^n \to Q$ that agree with r on T, namely

$$s_i(u) = \begin{cases} r(u) & \text{if } u \in T, \\ r_i(u) & \text{if } u \in T'. \end{cases}$$

Moreover, the set of those balanced partial runs $z_1, \dots, z_m$ that are compatible with t are a subset of $s_1, \dots, s_k$ and we obtain

$$\bigcup_{1 \leq i \leq m} \mathrm{compRuns}(z_i) = \mathrm{compRuns}(r).$$

This ensures the membership of compRuns(r) for every partial run r that is compatible with t in the σ-algebra generated by the set of balanced partial runs that are compatible with t. This implies the claimed equality by the closure properties of the generated σ-algebras and their minimality. □

This allows us to formulate

Lemma 22. *[CHS14, Lemma 36] $f^t_{\mathcal{A}}$ is a integrable function in the product space $\left(\mathrm{Runs}^t_{\mathcal{A}}, \mathcal{R}_{\mathcal{A}}, \mu_t\right) \otimes (\mathrm{Paths}, \mathcal{B}(D), \mu_B)$.*

Proof. For simplicity we refer to $f^t_{\mathcal{A}}$ in the following as f. f is by definition a characteristic function of the set

$$\left\{(r, p) \in \mathrm{Runs}^t_{\mathcal{A}} \times \mathrm{Paths} \mid r(p) \in \mathrm{Acc}_r(F)\right\}.$$

Therefore, it suffices to show that this set (denoted by $f^{-1}(1)$) is measureable in the product algebra $\mathcal{R}_{\mathcal{A}} \otimes \mathcal{B}(D)$. By Theorem 5 $\mathcal{R}_{\mathcal{A}} \otimes \mathcal{B}(D)$ is generated by the sets compRuns$(r) \times$ cyl(p) for all partial runs r and $p \in D^*$. Lemma 21 allows to consider only balanced runs.

In the following, we re-iterate concepts of the proof for Lemma 8 in this more complex setting to obtain the measureability of $f^{-1}(1)$. For any $p \in D^*$ fix the set of all $R_p \subseteq \text{Runs}^t_{\mathcal{A}} \times \text{Paths}$ such that $(r, \rho) \in R_p$ if and only if $p \sqsubseteq \rho$ and for all $p \sqsubseteq u \sqsubseteq \rho$ holds $r(u) \notin F$. The complement of $\cup_{w \in D^*} R_w$ describes $f^{-1}(1)$, since every element (r, p) in the complement of $\cup_{w \in D^*} R_w$ is not part of any R_w. Obviously, it is not part of those R_w for which $w \not\sqsubseteq p$ and for those $w \sqsubseteq p$ all w do have a prolongation v with $w \sqsubseteq v \sqsubseteq p$ with $r(v) \in F$. If on the other hand $(r, p) \in f^{-1}(1)$ then p is a path in r which satisfies the Büchi-condition and therefore for every prefix $w \sqsubseteq p$ there is a point v with $w \sqsubseteq v \sqsubseteq p$ with $r(v) \in F$ which makes (r, p) not part of any R_w.

It remains to show that R_w are measureable for every $w \in D^*$. Fix one such w and for every $n > |w|$ gather for all balanced proper partial runs r with depth n and words $p \in D^n$ with $w \sqsubseteq p$ and $r(p) \notin F$. The countable union of all such compRuns(r) and cyl(p) is collected in C_n and we claim that $R_u = \cap_{|u|<n} C_n$ by the following argument: for any $(r, p) \in R_u$ every prolongation of u that stays on p does not visit F and hence for every these prolongations v we know that (r, p) is part of $C_{|v|}$ and hence $(r, p) \in \cap_{|u|<n} C_n$. On the other hand assume $(r, p) \in \cap_{|u|<n} C_n$, thus for every viable prolongation of length k for u C_k witnesses the absence of an occurence of F.

This renders f^{-1} measureable and as indicator function for a measureable set f is $\mathcal{A} \otimes \mathcal{B}(D)$-measurable and by Theorem 4 we obtain the claim. □

Given the function $f^t_{\mathcal{A}}$ and its measureability in $\left(\text{Runs}^t_{\mathcal{A}}, \mathcal{R}_{\mathcal{A}}, \mu_t\right) \otimes (\text{Paths}, \mathcal{B}(D), \mu_B)$ allows to deduce by Theorem 6 the measureability of

$$g : \text{Runs}^t_{\mathcal{A}} \to [0, 1] \text{ with } g(r) = \int_{\text{Paths}} f^t_{\mathcal{A}}(r, \cdot)$$

in $\mathcal{R}_{\mathcal{A}}$. Moreover, we obtain the integrability of g by Theorem 4 and therefore deduce the measureability of $\left\{r \in \text{Runs}^t_{\mathcal{A}} \mid \mu_B(\text{Acc}(r)) = 1\right\}$ as $g^{-1}(1)$. This lays the basis for

Proposition 12. *[CHS14, Proposition 42] For a PWA $\mathcal{A}$ and a tree t holds*

$$\mathcal{A} \textit{ accepts } t \textit{ iff } \int f_{\mathcal{A}} d\mu_t \otimes \mu_B = 1.$$

Proof. We derive the following equivalences:

$$\begin{aligned}
\mathcal{A} \text{ accepts } t \text{ iff } & \mu_t(g^{-1}(1)) = 1 && \text{semantics of } \mathcal{A} \\
\text{iff } & \int_{\text{Runs}^t_{\mathcal{A}}} g d\mu_t = 1 && \text{Lemma 2} \\
\text{iff } & \int_{\text{Runs}^t_{\mathcal{A}}} \int_{\text{Paths}} f_{\mathcal{A}} d\mu_B d\mu_t && \text{Definition of } g \\
\text{iff } & \int_{\text{Runs}^t_{\mathcal{A}} \times \text{Paths}} f_{\mathcal{A}} d\mu_B \otimes \mu_t. && \text{Theorem 6}
\end{aligned}$$

□

We proceed by defining an equivalent WDTA for a given PWA, yielding

Theorem 32 (PWA Inclusion)**.** *For any PWA $\mathcal{A}$ exists a choiceless almost-surely accepting Büchi-WDTA $\mathcal{C}$ such that the languages accepted by $\mathcal{A}$ and $\mathcal{C}$ are equivalent.*

Proof. We provide for a given PWA $\mathcal{A}$ the definition of the equivalent WDTA $\mathcal{C}$ as

Definition 27.
For any PWA $\mathcal{A} = (Q, q_0, D, \Sigma, B, \Delta, \delta, F)$ we define a WDTA

$$\mathcal{C} = (Q, q_0, D, \Sigma, \Delta', F)$$

such that there exists one generator G^q_σ for every pair of $\sigma \in \Sigma$ and $q \in Q$ and

$$\Delta' = \{(q, \sigma, G^q_\sigma) : \text{ for all } q \in Q, \sigma \in \Sigma\} \text{ and } G^q_\sigma(p, d) = B(d) \cdot \sum_{\substack{\tau \in \Delta \\ \tau(d) = p}} \delta(\tau).$$

We examine the unique run r of $\mathcal{C}$ on any tree $t : D \to \Sigma$ and the measure μ_r it defines on $\mathcal{B}(Q \times D)$. We claim that $\mu_r(\mathrm{Acc}\,[Q]\,(F)) = 1$ if and only if $\int_{\mathrm{Runs}^t_{\mathcal{A}} \times \mathrm{Paths}} f^t_{\mathcal{A}} d\mu_t \otimes d\mu_B = 1$ which induces the equivalences of the languages of $\mathcal{A}$ and $\mathcal{C}$ by Proposition 12.

Therefore, we initially define

$$h : \mathrm{Paths} \to [0, 1] \text{ with } h(p) = \int_{\mathrm{Runs}^t_{\mathcal{A}}} f^t_{\mathcal{A}}(\cdot, p).$$

Again, by Theorem 6, we obtain

$$\int_{\mathrm{Runs}^t_{\mathcal{A}} \times \mathrm{Paths}} f^t_{\mathcal{A}} d\mu_t \otimes d\mu_B = 1 \text{ if and only if } \int_{\mathrm{Paths}} h d\mu_B = 1.$$

We argue that

$$\mu_r(\mathrm{cyl}((q_1, d_1) \ldots (q_n, d_n))) \text{ coincides with } \int_{p \in \mathrm{cyl}(d_1 \ldots d_n)} \mu_t(R^{d_1 \ldots d_n}_{q_1 \ldots q_n}) d\mu_B$$

where $R^{d_1 \ldots d_n}_{q_1 \ldots q_n} = \left\{ r \in \mathrm{Runs}^t_{\mathcal{A}} \mid r(d_i) = q_i \text{ for } 1 \leq i \leq n \right\}$ by

$$\begin{aligned}
\mu_r(\mathrm{cyl}((q_1, d_1) \ldots (q_n, d_n))) &= \prod_{0 \leq i \leq n-1} \left(B(d_{i+1}) \cdot \sum_{\substack{\tau \in \Delta \\ \tau(d_{i+1}) = q_{i+1}}} \delta(\tau) \right) \\
&= \prod_{0 \leq i \leq n-1} B(d_{i+1}) \cdot \prod_{0 \leq i \leq n-1} \sum_{\substack{\tau \in \Delta \\ \tau(d_{i+1}) = q_{i+1}}} \delta(\tau) \\
&= \mu_B(\mathrm{cyl}(d_1 \ldots d_n)) \cdot \mu_t(R^{d_1 \ldots d_n}_{q_1 \ldots q_n})
\end{aligned}$$

because the term

$$\prod_{0 \leq i \leq n-1} \sum_{\substack{\tau \in \Delta \\ \tau(d_{i+1}) = q_{i+1}}} \delta(\tau)$$

describes the probability that δ chooses while moving along $d_1 \ldots d_n$ those transitions that ensure a state sequence of $q_1 \ldots q_n$. Observing now that

$$\mu_r(\mathrm{Acc}\,[Q]\,(F)) = \int_{p \in \mathrm{Paths}} \mu_t(\{r \in \mathrm{Runs}^t_{\mathcal{A}} \mid r(p) \in \mathrm{Acc}(Q)\}) d\mu_B = \int_{\mathrm{Paths}} h\mu_B$$

implies the claim and concludes the proof. □

The concept of PWAs allows to define an interesting tree language.

Example 11. [CHS14, Proposition 43] For a given PBA $\mathcal{P} = (Q, q_0, \Sigma, \delta, F)$ and a blueprint $B : D \to [0, 1]$ we define an associated language $\mathcal{L}^B_{\mathcal{P}}$. This language contains all trees t such that the set of branches in t which are almost-surely accepted by $\mathcal{P}$ holds a μ_B-measure of 1. The idea is to define transitions τ_q for every two $q \in Q$ with $\tau_q(d) = q$ for every $d \in D$. And for a state p and a letter σ the transition τ_q is chosen with probability $\delta(p, \sigma, q)$. Then, every run contains one possible execution of $\mathcal{P}$ on every path of the tree. When translated into the equivalent WDTA we obtain transitions of the form

$$(q, \sigma, G) \text{ with } G(p, d) = \delta(q, \sigma, p) \cdot B(d).$$

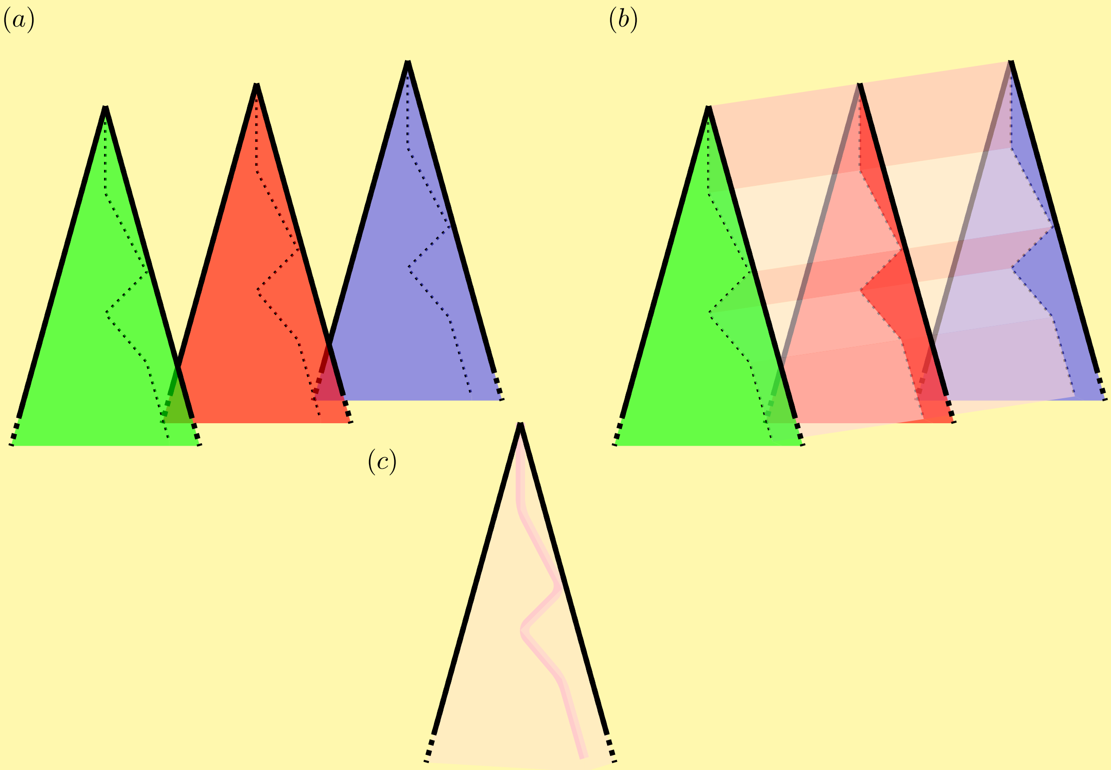
Therefore, along one particular path the run-tree of $\mathcal{P}$ unfolds where every movement is scaled with $B(d)$ (see Figure 9 for an illustration).

3.3.3. Partially Observable Markov Decision Processes

In the following we want to examine the observation that any run of a choiceless WDTA effectively is an unrollment of a transition system which is induced by a tree t. This behavior appears similarly to the concept of MDPs where the strategy of the actor unrolls the transition system. Indeed, we establish a close connection between POMDPs and WDTAs. In the beginning, we show that choiceless WDTAs can be used to model POMDPs where trees are interpreted as strategies. Furthermore, we use the mechanism of POMDPs to establish decideability results for algorithmic questions about WDTAs, e.g. deciding emptiness, or respectively deduce undecideability results. This allows us to reveal intriguing connections between WDTAs and POMDPs. For example the fact that alternation in tree automata precisely models partial observability in MDPs.

In the following, we use a similar observation as for the "equivalence" of unary trees to words. Namely, we consider strategies of the form $f : D^* \to \Sigma$ as D-ary Σ-trees. Therefore, we use the terms strategies and trees interchangably in the following.

Figure 9. Illustration of a PWA which simulates a PBA on all paths. In (a) we picture some of the parallelly executed runs with one highlighted path. On this path every run executes one possible run of the PBA. Therefore, all runs are interconnected on this path by executing different state sequences (see (b)). When translated into the equivalent WDTA along this path the complete run-tree of the associated PBA is executed (highlighted by the colored path in (c)).



Lemma 23. *Given a POMDP $\mathcal{M}$ with states S, actions A and equivalence relation $\sim$. Fixing $O = \{[s]_\sim : s \in S\} = [S]_\sim$, the* observations *of $\mathcal{M}$, allows to define a choiceless WDTA-skeleton $\mathcal{A}$ running over O-ary A-trees such that any strategy $s : O^* \to A$ induces*

$$(S^\omega, \mathcal{B}(S), \mu_s) \ \text{ and } \ \left((S \times O)^\omega, \mathcal{B}(S \times O), \mu_r\right)$$

for $\mathcal{M}$ and $\mathcal{A}$ respectively if r is the unique run of $\mathcal{A}$ on s. Moreover, $(S^\omega, \mathcal{B}(S), \mu_s)$ can be isomorphically embedded into $\left((S \times O)^\omega, \mathcal{B}(S \times O), \mu_r\right)$.

Proof. Given the POMDP

$$\mathcal{M} = \left(S, A, (\tau_a)_{a \in A}, s_0, \sim\right).$$

Define O as above. We define

$$\mathcal{A} = (S, s_0, O, A, \Delta)$$

where

$$\Delta = \{(s, a, G_s^a) : s \in S, a \in A\} \text{ with } G_s^a(s', o) = \begin{cases} \tau_a(s, s') & \text{if } o = [s']_\sim \,, \\ 0 & \text{otherwise.} \end{cases}$$

Thus, from the current state those states with observation $o \in O$ are sent in direction o. We define obs : $S \to O$ with $\text{obs}(s) = [s]_\sim$. Applying Lemma 6 on obs and the probability space $(S^\omega, \mathcal{B}(S), \mu_s)$ gives a probability space $((S \times O)^\omega, \mathcal{B}(S \times O), \mu')$ with $\mu' = \mu_s \circ \text{lift}_{\text{obs}}^{-1}$. Moreover, we know that lift_{obs} is an isomorphism from $(S^\omega, \mathcal{B}(S), \mu_s)$ to $\left(\text{lift}_{\text{obs}}(S^\omega), \mathcal{B}(S \times O)_{|\text{lift}_{\text{obs}}(S^\omega)}, \mu'\right)$. We argue that μ' coincides with μ_r for the unique run r of $\mathcal{A}$ on s. Examining the definitions of generators $\mathcal{G}(\mathcal{A})$, we observe

$$\mu_r(\text{cyl}((s_1, o_1) \dots (s_n, o_n))) = \begin{cases} \prod_{0 \leq i \leq n-1} \tau_{t(o_1 \dots o_i)}(s_i, s_{i+1}) & \text{if } [s_i]_\sim = o_i \text{ for } 1 \leq i \leq n, \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

Hence, μ' coincides with μ_r by unique extension of μ' and μ_r. The claim follows from Lemma 6. □

This lemma allows to deduce the following

Theorem 33. *For any POMDP $\mathcal{M}$ with states S, observations O, actions A with an associated Parity-condition* par *exist a choiceless WDTA $\mathcal{A}$ such that for any object $s : O^* \to A$ and the unique run r of $\mathcal{A}$ on s holds that $\mu_s(\text{Acc}(\text{par})) = \mu_r(\text{Acc}\,[Q]\,(\text{par}))$. If $\sim$ is the equality relation $\mathcal{A}$ is uni-directional.*

Proof. The construction follows Lemma 23. Additionally, we equip the resulting skeleton $\mathcal{A}$ with the acceptance condition par. The claim is implied by the observation that $\text{lift}_{\text{obs}}(\text{Acc}(\text{par})) = \text{Acc}\,[Q]\,(\text{par}) \cap \text{lift}_{\text{obs}}(S^\omega)$ and $\mu_r(\text{Acc}\,[Q]\,(\text{par})) = \mu_r(\text{Acc}\,[Q]\,(\text{par}) \cap \text{lift}_{\text{obs}}(S^\omega))$ since $\mu_r(S^\omega \setminus \text{lift}_{\text{obs}}(S^\omega)) = 0$.

Notably, if $\sim$ is the equality relation we can identify $[s]_\sim$ with s for all $s \in S$ and obs becomes the identity function. By construction of $\mathcal{A}$ all generators in G only weights tuple (s, s). □

On the other hand, we want to relate choiceless WDTAs to POMDPs. Especially, we obtain arguments to solve algorithmic questions by the corresponding questions for POMDPs. Example 5 introduced the idea to consider PBAs as POMDPs with one equivalence class (for the consistency of generated word). We adapt this idea to obtain a comparable equivalence. Namely, we construct a POMDP for a given choiceless WDTA such that any run r of WDTA is the unrollment of the POMDP if the tree is used as strategy. The choicelessness of the WDTA ensures to obtain an interchangability of the strategies and trees. This is the case because the choices of a strategy immediately determine the transition probabilities while general WDTAs allow for a *second* decision once the letter is chosen, namely which transition to take.

This equivalence allows to formulate the emptiness problem of a choiceless WDTA as computation of a strategy for POMDPs. The choices of Eve (i.e. the strategy) are which letter is used at which position of the tree. Hence, the player effectively constructs with his choices the tree t by providing for every situation the corresponding letter. We define analogously to [CHS14, Section 3.5 Emptiness Problem]

Definition 28. ***Run Building Game***:
For a choiceless WDTA-sceleton $\mathcal{A} = (Q, q_0, D, \Sigma, \Delta)$ we define the set of actions for the player as Σ. The states of the game correspond with the positions a run can stay in and a dedicated initial position, i.e. $S = (Q \times D) \cup \{q_0\}$. In order to obtain a valid tree from a strategy we tinker with the observability of the states, namely we restrict the player to only observe the directions and not the states. Hence, we define $\sim \subseteq (Q \times D)^2$ such that

$$(q, d) \sim (p, b) \text{ if and only if } d = b.$$

The actions of the player lead to the movement probabilities induced by the generators associated with the current state and the player's choice. Firstly, there is one unique generator G^q_σ such that $(q, \sigma, G^q_\sigma) \in \Delta$. Secondly, we define τ_σ for every $\sigma \in \Sigma$ such that

$$\tau_\sigma((q, d), (p, b)) = G^q_\sigma(p, b)$$

and, additionally,

$$\tau_\sigma(q_0, (q, d)) = G^{q_0}_\sigma(q, d).$$

Concludingly, we obtain an associated POMDP

$$\mathcal{M} = \left(Q \times D \cup \{q_0\}, \Sigma, (\tau_\sigma)_{\sigma \in \Sigma}, q_0\right).$$

In order to prove the viability of this game definition we show that strategies and trees are both interchangeable objects. Moreover, a strategy directly defines a measure on $\mathcal{B}(Q \times D)$ in $\mathcal{M}$ and a tree indirectly induces a measure on $\mathcal{B}(Q \times D)$ by the corresponding unique run r. The central observation is that these measures coincide.

Lemma 24. *For any object $t : D^* \to \Sigma$ the measures μ_r on $\mathcal{B}(Q \times D)$ for the unique run r on t of $\mathcal{A}$ and the measure μ_t on $\mathcal{B}(Q \times D)$ for the strategy t in $\mathcal{M}$ coincide.*

Proof. Firstly, we observe that by construction of τ_σ every play can *never* return to q_0. Additionally, by definition of strategies for POMDPs, the initial state is encoded implicitly (since its unique we can omit its explicit occurence in the history of a game). This allows us to restrict any observation to those strategies that are defined on histories in the $Q \times D$-part of $\mathcal{M}$. Furthermore, we recall that it suffices to show the equivalence of both measures on the generating objects, i.e. on all $\mathrm{cyl}(u)$ for all $u \in (Q \times D)^*$, since these values uniquely determine the extended measures on $\mathcal{B}(Q \times D)$. We fix one object $t : D^* \to \Sigma$, the unique run r of $\mathcal{A}$ on t and $u = (q_1, d_1) \dots (q_n, d_n) \in (Q \times D)^*$. For t, r and u we observe

$$\begin{aligned}
\mu_t(\mathrm{cyl}(u)) &= \tau_{t(\epsilon)}(q_0, (q_1, d_1)) \cdot \prod_{1 \leq i < n} \tau_{t(d_1 \dots d_i)}((q_i, d_1), (q_{i+1}, d_{i+1})) && \text{semantics of } \mathcal{M} \\
&= G^{q_0}_{t(\epsilon)}(q_1, d_1) \cdot \prod_{1 \leq i < n} G^{q_i}_{t(d_1 \dots d_i)}(q_{i+1}, d_{i+1}) && \text{definition of } (\tau_\sigma)_{\sigma \in \Sigma} \\
&= \mu_r(\mathrm{cyl}(u)) && \text{semantics of } \mathcal{S}.
\end{aligned}$$

This yields the claimed equality of the generating objects and therefore, by unique extension the equivalence of μ_t and μ_r. □

This directly entails the following

Theorem 34. *For every choiceless WDTA $\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{Acc})$ exists a POMDP $\mathcal{M} = \left(Q \times D \cup \{q_0\}, q_0, (\tau_\sigma)_{\sigma \in \Sigma}, \mathrm{Acc}[Q], \sim\right)$ such that any strategy s for $\mathcal{M}$ is isomorphic to a tree $t : D^* \to \Sigma$ and for the unique run r of $\mathcal{A}$ on t holds that μ_r and μ_s coincide on $\mathcal{B}(Q \times D)$.*

Regarding the choicelessness requirement on the modelled WDTA we expand in the following the discussion in Example 9. We use the partial observability of the emptiness game to obtain a consistency of the tree Eve constructs. Allowing Eve to additionally choose a transition requires a more complex management of information. The letter of the tree must be consistent for all plays that have a common D-projection. For the transition on the other hand it is necessary to expose the current state (and potentially the complete history of the play) to capture the full expressibility of WDTAs. POMDPs are not equipped to handle these multiple information requirements. This renders choiceless WDTAs algorithmically easier to approach while maintaining a sufficient expressiveness.

The construction of a POMDP for a choiceless WDTAs (Theorem 34) with interchangable notions of trees and stratgies allows to use the computation of strategies for POMDPs to solve the emptiness problem for choiceless WDTAs. Especially, for almost-surely accepting choiceless Büchi-WDTAs we may use Theorem 20 to do so.

Corollary 10 (Emptiness Almost-Sure Büchi WDTA)**.** *The emptiness problem for a choiceless WDTA with almost-sure acceptance measure of a Büchi condition is decideable in time exponential in $|Q| \cdot |D|$.*

On the other hand, we obtain for uni-directional WDTAs a decision procedure for the emptiness problem even for Parity-conditions. The central idea is to drop the partial observability from the representation since there is only one weighted state at each position. Thus, the restriction on observability is not necessary to ensure the interchangeability of a strategy and a tree. Hence, we obtain

Corollary 11 (Emptiness Almost-Sure Uni-Directional WDTA)**.** *[CHS14, Corollary 47] The emptiness problem for an uni-directional WDTA with almost-sure or positive acceptance measure of a Parity-condition is decidable. If a tree exists it can be computed in time polynomial in $|Q| \cdot |D|$.*

Proof. We translate a given uni-directional WDTA $\mathcal{A} = (Q, q_0, D, \Sigma, \Delta, \mathrm{par})$ to a MDP

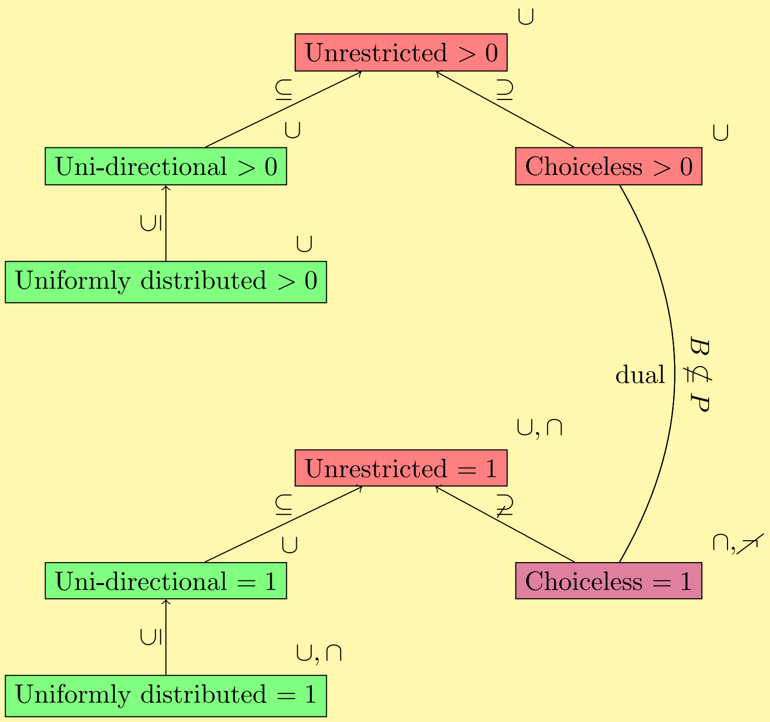
$$\mathcal{M} = \left(Q \times D \cup \{q_0\}, q_0, \mathcal{G}_{\mathcal{A}}, (\tau)_{G \in \mathcal{G}_{\mathcal{A}}}\right) \text{ with } \tau_G((q,d), (p,b)) = G((p,b)).$$

By an inductive argument we can exploit the uni-directionality to obtain that for a strategy s we can state that for any $d_1 \ldots d_n \in D^*$ there is at most one sequence $q_1 \ldots q_n \in Q^*$ such that $q_0 (q_1, d_1) \ldots (q_n, d_n)$ is a possible play under s in $\mathcal{M}$. Fix these sequences in D^* as P. Moreover, s induces a tree t by fixing for every $d_1 \ldots d_n \in P$ and the unique $q_1 \ldots q_n$ such that $q_0 (q_1, d_1) \ldots (q_n, d_n)$ describes a possible play in $\mathcal{M}$ under s and setting $t(d_1 \ldots d_n)$ to one $\sigma \in \Sigma$ such that $(q_n, \sigma, s((q_1, d_1) \ldots (q_n, d_n))) \in \Delta$. All other elements in t are set arbitrarily. We observe that the unrollment of $\mathcal{M}$ under s coincides with the run on t that takes those transitions s chooses its generators from. Additionally, we can obtain for any run r of $\mathcal{A}$ on a tree t a corresponding strategy s with $s(u) = r(u)$ for all $u \in (Q \times D)^*$. Concludingly, using Theorem 21 on this $\mathcal{M}$ and the Parity-condition par (applied to the state component of the states in $\mathcal{M}$) yields the postulated algorithm. $\square$

The constructions used for Theorem 34 and Corollary 11 allows to pinpoint the direct connection of complete observability to uni-directionality. Specifically, the resulting MDP that corresponds to the run of a choiceless WDTA requires partial observability if and only if the WDTA is *not* uni-directional.

Concludingly, we summarize our results on WDTAs in Figure 10.

Figure 10. A "map" of the presented results for WDTAs. Arrows marked with inclusion symbols are used to illustrate increasing expressiveness from the beginning to the end of the arrow (the equality symbol is striked out if it is a proven strict generalisation). Annotated $\cap$, $\cup$ and $\neg$ stand for closure under intersection, union and complement respectively. Naturally, a striked out version of one operater shows the proven lack of closure under it. Additionally, we use $B \not\subseteq P$ to state that a model restricted to a Büchi-condition is not included in a model with access to Parity-conditions. Colors encode decidability results for the emptiness problem: red illustrates undecidability, green decidability and purple decidability for Büchi- but undecidability for Parity-conditions.



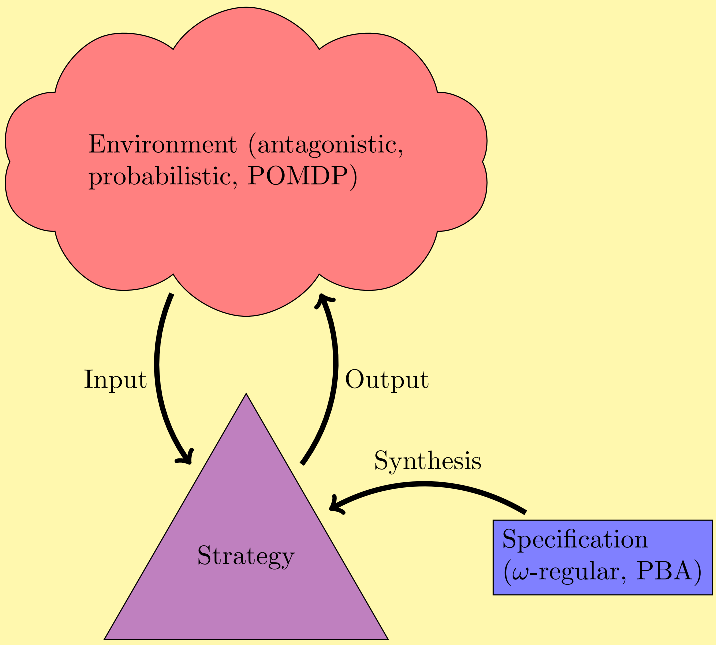
4. Synthesis

The synthesis problem is formulated in [Chu62] as the question if for a given logical formula, we call this specification, we can derive an algorithm which behavior is "good" in the sense of the formula. We formalize this question in the following

Definition 29. ***Synthesis Problem***:
Given a logical specification $\phi(\cdot, \cdot)$ over inputs and outputs from I^ω and J^ω respectively. The synthesis problem requires to compute for any ϕ an algorithm $S : I^+ \to J$ such that for every $\alpha_1\alpha_2 \dots \in I^\omega$ and every $S(\alpha) = S(\alpha_1)S(\alpha_1\alpha_2)\dots$ satisfies $\phi(I, S(I))$ or prove that such an S cannot exist (cp. the illustration in Figure 11).

Naturally, the complexity of this problem is tightly related to the expressiveness of

Figure 11. Illustration of the synthesis question. The aim is to provide an algorithm which "synthesises" for any specification a strategy or proves that there cannot exist a strategy that satisfies the specification.



ϕ. The comprehensive demand of this question, namely to generate for all inputs a "good" output, allows for a game-theoretic formulation. As suggested in [BL69] we can consider the environment as antagonistic and formulate the synthesis question in terms of strategies, i.e. one player generates inputs while another player generates outputs. We call these players Input and Output respectively. The game unfolds by Input sequentially choosing inputs $i_1 i_2 \dots$ and Output reacting by $j_1 j_2 \dots$. Thus, a play

$$\begin{matrix} i_1 & i_2 & i_3 & \dots \\ j_1 & j_2 & j_3 & \dots \end{matrix}$$

forms and Output wins if and only if $\phi(i_1 i_2 \dots, j_1 j_2 \dots)$ evaluates to true. Strategies of Output are usually described as functions $f : I^+ \to J$ such that Output plays at position $n > 0$ the output-symbol $f(i_1 \dots i_n) = j_n$. In this sense we are interested in winning strategies of Output. Following, we present known synthesis results for those ϕ which allow to capture the associated relation as a ω-regular language. Subsequently, we pivot towards specifications given as almost-surely accepting PBAs. Afterwards, we consider environments for which we assume probabilistic behavior. These environments allow to state the synthesis problem for almost-sure satisfaction of ϕ. At last, we model probabilistic environments more explicit as POMDPs. We try to compute strategies for these that almost-surely satisfy given specifications.

4.1. Antagonisitic Environments

Initially, we consider an ω-regular class of specifications ϕ for inputs I and outputs J. Hence, we assume the existence of an ω-regular language $\mathcal{L}_\phi \subseteq (J \times I)^\omega$ with

$$(j_0, i_0)\,(j_1, i_1) \dots \in \mathcal{L}_\phi \text{ if and only if } \phi(i_0 i_1 \dots, j_1 j_2 \dots) \text{ is true.} \tag{4.1}$$

Notably, the output symbol j_0 is irrelevant but included to ease the technicalities of the following argument. With the well-researched theory of tree automata the associated synthesis problem for ω-regular specifications can be solved transparently. In fact, it is one of the initial motivations to study tree automata [Rab72]. The idea is to use trees to model the interaction between input and output symbols. The directions of a tree model inputs while the symbols in the tree represent outputs. Similar to the arguments of the proof of Theorem 33, a tree models a strategy. We design a PTA such that a tree is accepted if and only if all paths are part of $\mathcal{L}_\phi$. We model this in analogy to [Rab72, Lemma 15] with

Definition 30. ***Synthesis PTA***:
For a given DPA

$$\mathcal{P} = (Q, q_0, J \times I, \delta, \mathrm{par})$$

we define a PTA

$$\mathcal{A} = (Q, q_0, I, J, \Delta, \mathrm{par})$$

where Δ contains transtitions $(q, j, (p_i)_{i \in I})$ with $p_i = \delta(q, (j, i))$.

Notably, $\mathcal{A}$ executes $\mathcal{P}$ on every path and therefore, if we choose $\mathcal{P}$ as the DPA which precisely accepts $\mathcal{L}_\phi$, we deduce that $\mathcal{A}$ accepts only those trees that satisfy ϕ on all paths. This implies that $\mathcal{A}$ accepts those strategies t which satisfy ϕ. Hence, constructing (see above) and deciding the emptiness (see Theorem 29) for $\mathcal{A}$ yields the desired synthesis algorithm for ω-regular specifications ϕ and gives

Theorem 35. *[Rab72, Theorem 21, Theorem 22] The synthesis problem for ω-regular specifications ϕ is decidable.*

If a strategy exists and ϕ is captured by a DPA

$$\mathcal{P} = (Q, q_0, J \times I, \delta, \text{par})$$

then we can compute such a strategy in time in $\mathcal{O}(|\text{par}(Q)| \cdot |Q \times Q| \cdot \left(\frac{|Q|}{d}\right)^d)$ with $d = \lfloor \frac{|\text{par}(Q)|}{2} \rfloor$.

This approach suits the game-theoretic interpretation of the synthesis problem well. Consider the associated emptiness-game to the defined PTA $\mathcal{A}$. Since $\mathcal{A}$ shows deterministic behavior, we express Eve's strategy by choosing a letter rather than a transition. The transition is uniquely identified by the letter due to $\mathcal{A}$'s determinism. Also, Adam's role is always captured by choosing a direction the game moves to. Therefore, Eve's and Adam's correspond to the behavior of Output and Input respectively while the DPA that is associated with ϕ is executed on every play of the input-output game.

In the following, we consider specifications ϕ which are not ω-regular but definable by almost-surely accepting PBAs. In analogy to the formulation (4.1) above, we assume for ϕ the existence of an almost-surely accepting PBA $\mathcal{P}$ such that

$$(j_0, i_0)\,(j_1, i_1)\ldots \text{ is accepted by } \mathcal{P} \text{ if and only if } \phi(i_0 i_1 \ldots, j_1 j_2 \ldots) \text{ is true.} \tag{4.2}$$

Specifically, we emphasize that probabilistic behavior is not considered anywhere else but the specification. Notably, we consider deterministic strategies for Output.

We model our approach in close connection to the game-theoretic interpretation of Theorem 35. This entails introducing a new class of graph games, called Partial Observation Stochastic Games (POSGs) (cp. [CD14; CLS18]). We draw motivation from the behavior of POMDPs and introduce a proper generalisation. POMDPs essentially are graph-games between Eve and a second player, called Random. The game unfolds by Eve specifying with her action a set of possible successors and Random choosing one of these successors. However, although Eve may base her choice on the history of the play Random is restricted to randomized positional strategies only (encoded in the transition probabilities). Therefore, Random is reffered to as a half-player and POMDPs as $1\frac{1}{2}$-player games [CD14].

We introduce POSG as $2\frac{1}{2}$-player games by introducing a “full”-player Adam as enemy of Eve. The game unfolds in very similar fashion to POMDPs but at every position Eve *and* Adam simultaneously and independently choose an action and Random reacts by a randomized positional strategy. We formalize this in

Definition 31. ***Partial Observation Stochastic Game***:
A POSG-arena G is defined by a set of states S (with an inital state $s_0 \in S$), actions for Eve and Adam as E and A respectively, transition probabilities $(\tau_{e,a})_{e\in E, a\in A}$ and equivalence classes $\sim_E$ and $\sim_A$ restricting the observations of Eve and Adam respectively. We obtain

$$G = \Big(S, s_0, E, A, (\tau_{e,a})_{e\in E, a\in A}, \sim_E, \sim_A\Big)$$

with

$$\tau_{e,a} : S \times S \to [0,1] \text{ s. t. } \tau_{e,a}(s,\cdot) \in \mathcal{D}(S) \text{ for all } a \in A, e \in E.$$

A strategy for Eve (Adam) is defined as $f : [S]^*_{\sim_E} \to E$ ($g : [S]^*_{\sim_A} \to A$). For any such pair of strategies f and g we obtain a probability space

$$(S^\omega, \mathcal{B}(S), \mu_{f,g})\,.$$

A POSG is defined by an arena G and an associated language $\mathrm{Acc} \subseteq S^\omega$ forming $\mathcal{G} = (G, \mathrm{Acc})$. Notably, $\mathcal{G}$ is a zero-sum game, i.e. Eve tries to obtain a play in Acc while Adam tries to force plays in $S^\omega \setminus \mathrm{Acc}$.

We restrict Acc to be defined as Parity-, Rabin-, Muller- or Büchi-condition which directly ensures its measurability. In similar sense as for graph games we consider a strategy f for Eve almost-surely (postively) winning if for all strategies g of Adam we have $\mu_{f,g}(\mathrm{Acc}) = 1$ ($\mu_{f,g}(\mathrm{Acc}) > 0$). The natural initial observation that Adam's role can be reduced to oblivion by only granting him a single action allows to capture the notions of POMDPs. This entails strong restrictions regarding algorithmic approaches for computing strategies for Eve.

Corollary 12. *[CLS18; CD14] For a given POSG $\mathcal{G} = (G, \mathrm{Acc})$ the following problems are undecidable:*

- *Exists a positively winning strategy for* Eve *if* Acc *is defined as Büchi-, Parity-, Rabin- or Muller-condition?*
- *Exists an almost-surely winning strategy for* Eve *if* Acc *is defined as Parity-, Rabin- or Muller-condition?*

Proof. These are immediate consequences of the negligibility (not in the sense of measurability theory but in the game-theoretic sense) of Adam's actions and Theorem 17 (or the respective Corollary 3). □

In order to reduce the complexity of our arguments we only consider POSGs where Eve and Adam are *equally informed*, i.e. $\sim_E = \sim_A$. Similar to POMDPs, almost-surely winning strategies for Eve in POSG with Büchi-conditions are "simple" enough to allow for computation of winning strategies.

Theorem 36. *[CD14, Theorem 6][CLS18, Theorem 5.3][1] It is possible to decide if there is an almost-surely winning strategy for* Eve *in a POSG* $\mathcal{G} = (G, F)$ *where* F *is a Büchi-condition and* $\sim_E = \sim_A$*.*

This decision procedure takes time doubly exponential in $|S|$*.*

This allows us to answer the synthesis question imposed by an almost-surely accepting PBA $\mathcal{P}$ as specification positively for antagonistic environments. The idea is to define a POSG in which both players only observe an input-output game while the stoachastic process of $\mathcal{P}$ unfolds in the "background". Any play is eventually evaluated in terms of acceptance of $\mathcal{P}$. Formally, for a fixed almost-surely accepting PBA

$$\mathcal{P} = (Q, q_0, I \times J, \delta, F')$$

we define

Definition 32. ***POSG for*** $\mathcal{P}$:
Set

$$\mathcal{G}_{\mathcal{P}} = \Big(I \times Q \times J \uplus \{q_0\}, J, I, (\tau_{j,i})_{j \in J, i \in I}, \sim_E, \sim_A, F\Big)$$

with

$$\sim = \sim_E = \sim_A = \left\{((i,q,j), (i',q',j')) \in (I \times Q \times J)^2 \mid i = i', j = j'\right\}$$

Moreover, we fix $F = I \times F' \times J$ and

$$\tau_{j',i'}((i,q,j), (o,p,u)) = \begin{cases} \delta(q, (i',j'), p) & \text{if } o = i', u = j', \\ 0 & \text{otherwise,} \end{cases}$$

where q_0 is treated as (i, q_0, j) for any $i \in I, j \in J$.

Here Eve corresponds with Output while Adam plays the role of Input. Central to this argument is that regarding the observations of Eve and Adam $\mathcal{G}_{\mathcal{P}}$ presents as input-output game. Setting $S = I \times Q \times J$ we observe by definition of $\sim$ that we can partition S into $\{[(i,*,o)]_\sim : i \in I, o \in J\}$ where $[(i,*,o)]_\sim$ describes the equivalence class $\{(i,q,o) : q \in Q\}$. By definition of all $\tau_{j,i}$ we obtain $\mathrm{support}(\tau_{j,i}(s,\cdot)) \subseteq [(i,*,j)]_\sim$. Hence, every strategy for Eve (Adam) presents as function $f : (I \times J)^* \to J$ ($g : (I \times J)^* \to I$). The only stochastic process in this game is associated with the state component which is observed neither by Eve nor by Adam. This allows us to deduce that any pair of strategies f and g for Eve and Adam respectively induce one unique word $\alpha_f^g \in (I \times J)^\omega$. Moreover, we can consider a subset of $\mathcal{B}(I \times Q \times J)$ which is

[1] Since it was unclear if the publication of this paper preceeded the completion of this thesis, we included the relevant arguments in Appendix A to ensure their availability. Especially, since we rely on the contained arguments for the claimed complexity bound.

induced by $\alpha_f^g = (i_1, j_1)\,(i_2, j_2)\ldots$. Namely, we define two sets

$$F = \bigcap_{k>0} \bigcup_{\substack{p_1\ldots p_k \in Q^k \\ u_1\ldots u_k \in J^k}} \mathrm{cyl}((i_1, p_1, u_1)\ldots(i_k, p_k, u_k))$$
$$G = \bigcap_{k>0} \bigcup_{\substack{p_1\ldots p_k \in Q^k \\ u_1\ldots u_k \in I^k}} \mathrm{cyl}((u_1, p_1, j_1)\ldots(u_k, p_k, j_k)).$$

By definition F and G are measurable sets in $\mathcal{B}(I \times Q \times J)$ such that the J-component respectively the I-component agrees with α_f^g. Moreover, by definition of $\tau_{i,j}$ for all $i \in I, j \in J$ we deduce that

$$\mu_{f,g}(C) = 0 \text{ for all } C \notin \mathcal{B}(I \times Q \times J)_{|F\cap G}. \tag{4.3}$$

The central idea of this proof is formulated in

Lemma 25. $\left(Q^\omega, \mathcal{B}(Q), \mu_{\alpha_f^g}\right)$ *can be isomorphically embedded into*

$$\left((I \times Q \times J)^\omega, \mathcal{B}(I \times Q \times J), \mu_{f,g}\right).$$

Proof. We propose the following bijection

$$\mathrm{io} : Q^\omega \to \mathcal{B}(I \times Q \times J)_{|F\cap G} \text{ with } \mathrm{io}(q_1 q_2 \ldots) = (i_1, q_1, j_1)\,(i_2, q_2, j_2)\ldots.$$

We observe that $\mathcal{B}(I \times Q \times J)_{|F\cap G}$ can be generated by all cylindric sets

$$\mathrm{cyl}((i_1, p_1, j_1)\ldots(i_n, p_n, j_n)) \cap F \cap G \text{ for } p_1 \ldots p_n \in Q^*.$$

Notably, it holds

$$\mathrm{io}(\mathrm{cyl}(p_1 \ldots p_n)) = \mathrm{cyl}((i_1, p_1, j_1)\ldots(i_n, p_n, j_n)) \cap F \cap G.$$

Hence, we biject the generating $\mathcal{B}(Q)$ and $\mathcal{B}(I \times Q \times J)_{|F\cap G}$ sets and obtain similar to the proof of Lemma 6 the measurability of io as well as io^{-1}.

We show that $\mu_{f,g}$ coincides with $\mu_{\alpha_f^g} \circ \mathrm{io}^{-1}$ on $\mathcal{B}(I \times Q \times J)_{|F\cap G}$ which induces the required embedding by Equation 4.3. For a fixed sequence $p_1 \ldots p_n \in Q^*$ we have

$$\begin{aligned}
\mu_{\alpha_f^g}(\mathrm{cyl}(p_1 \ldots p_n)) &= \delta(q_0, (i_1, j_1), p_1) \prod_{1\le k<n} \delta(p_k, (i_{k+1}, j_{k+1}), p_{k+1}) \\
&= \tau_{j_1,i_1}(q_0, (i_1, p_1, j_1)) \prod_{1\le k<n} \tau_{j_k,i_k}((i_k, p_k, j_k), (i_{k+1}, p_{k+1}, j_{k+1})) \\
&= \mu_{f,g}(\underbrace{\mathrm{cyl}((i_1, p_1, j_1)\ldots(i_n, p_n, j_n)) \cap F \cap G}_{=\mathrm{io}(\mathrm{cyl}(p_1\ldots p_n))}).
\end{aligned}$$

Hence, by unique extension of $\mu_{\alpha_f^g}$ and $\mu_{f,g}$ the claim follows. □

This leads us into characterising almost-surely winning strategies for Eve in $\mathcal{G}_{\mathcal{P}}$ in

Lemma 26. *A strategy f in $\mathcal{G}_{\mathcal{P}}$ is almost-surely winning for* Eve *if and only if f synthesizes an algorithm for the associated specification $\phi_{\mathcal{P}}$.*

Proof. Initially, we observe that for a fixed strategy f every input $\beta = \beta_0\beta_1 \cdots \in I^\omega$ describes one strategy of Adam by setting $g(u) = \beta_{|u|}$. On the other hand, we obtain for every possible strategy g one sequence of inputs, namely the I-projection of α_f^g. Hence, the strategy space of Adam coincides with all possible input-sequences.

Moreover, using Equation 4.3 and Lemma 25 we state

$$\begin{aligned}
\mu_{\alpha_f^g}(\mathrm{Acc}(F')) &= \mu_{\alpha_f^g}(\mathrm{io}(\mathrm{Acc}(F'))) \\
&= \mu_{f,g}(\mathrm{Acc}(F) \cap F \cap G) && \text{Lemma 25} \\
&= \mu_{f,g}(\mathrm{Acc}(F)) && \text{Equation (4.3).}
\end{aligned}$$

Therefore, given an almost-surely winning strategy f for Eve we get for every g that $\mu_{f,g}(\mathrm{Acc}(F)) = 1$. Hence, $\mu_{\alpha_f^g}(\mathrm{Acc}(F')) = 1$ and therefore, α_f^g is accepted by $\mathcal{P}$. This means that every possible input-sequence β is met with an output-sequence γ such that $\phi_{\mathcal{P}}(\beta, \gamma)$ evaluates to true.

If on the other hand f is not almost-surely winning for Eve there is a riposte g of Adam such that $\mu_{f,g}(\mathrm{Acc}(F)) < 1$. Hence, $\mu_{\alpha_f^g}(\mathrm{Acc}(F')) < 1$ and therefore the non-acceptance of α_f^g by $\mathcal{P}$. This implies that Adam can generate an input-sequence β such that Eve generates the output-sequence γ but $\phi_{\mathcal{P}}(\beta, \gamma)$ evaluates to false. □

By this characterisation of winning strategies of Eve in $\mathcal{G}_{\mathcal{P}}$ we immediately may conclude

Theorem 37. *For any almost-surely accepting PBA*

$$\mathcal{P} = (Q, q_0, I \times J, \delta, F')$$

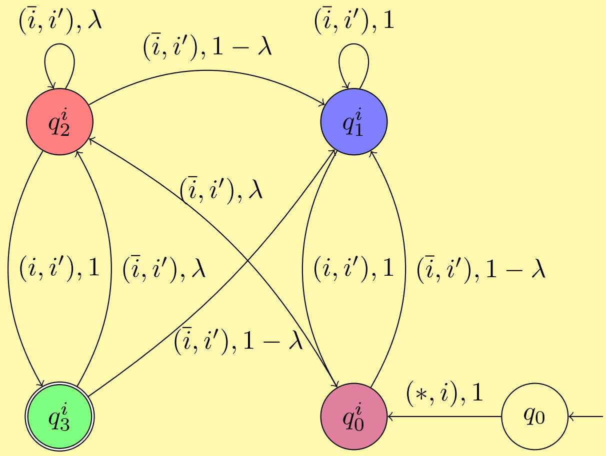
the synthesis problem for the specification ϕ imposed by $\mathcal{P}$ (as in (4.2)) is decidable. If a strategy exists it can be computed in time doubly exponential in $|I| \cdot |Q| \cdot |J| + 1$.

Proof. Using Theorem 36 on $\mathcal{G}_{\mathcal{P}}$ gives the desired decision procedure. □

This result actually is intriguing in the sense that we consider a probabilistic specification but obtain a *deterministic* algorithm which surely shows desired behavior. We consider the following

Example 12. Reconsider the PBA examined in Example 3. We alter this PBA slightly to obtain a specification which is defined in terms of an PBA by replacing a-elements by those occurences where the ouput differs from the last input and b-elements by occurences where the output mirrors the input. The initial output is dismissed (as suggested by (4.2)). The resulting almost-surely accepting PBA is illustrated in Figure 12. And the derived input-output game is shown in Figure 13.

Figure 12. Specification-PBA for Example 12 (based on the PBA depicted in Figure 6). Consider $I = J = \{0, 1\}$ where the superscript i represents a storage for every state (realised by using two states, e.g. q_0^0 and q_0^1) which holds the last read input symbol. The output symbols either mirrors the stored symbol (i) or inverts it ($\bar{i}$), while the input i' is always stored within the state. Initially, the output is irrelevant for the transition (indicated by $*$).



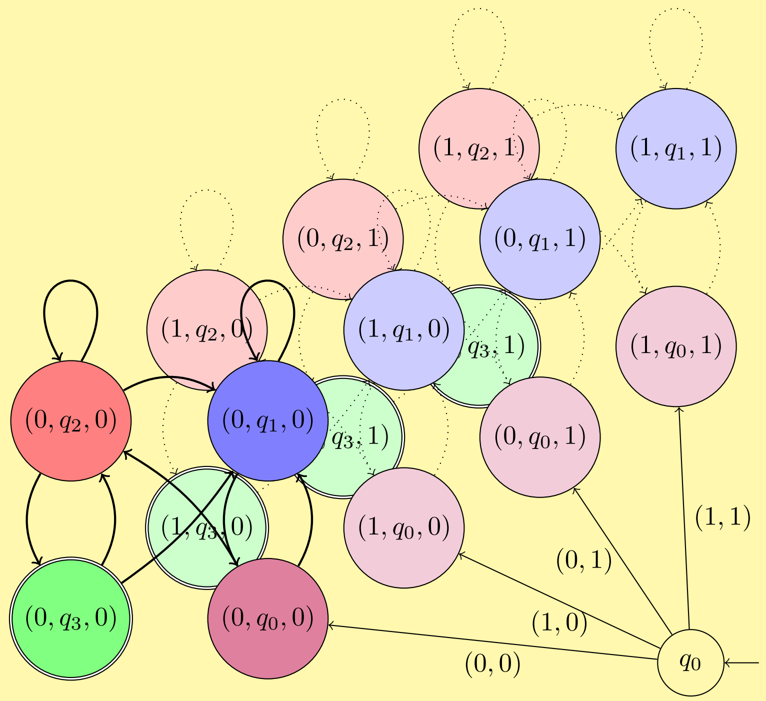
4.2. Stochastic Environments

Another approach is introduced in [Sch06] by considering environments which do not act antagonistically but probabilistically, i.e. we assume that inputs are generated by a probabilistic process. We explore the usage of alternating tree automata to address this problem. Specifically, their close connection to games is essential to the following considerations. We formulate the concept of ϵ-environments. For an $\epsilon > 0$ such an environment generates an input i with a certain probability which respects at any time ϵ as lowerbound. The form of the strategy that is meant to be synthesised is given as a J-labeled I-tansition-system $\mathcal{T}$ of the form

$$\mathcal{T} = (S, s_0, \tau, \ell)\,.$$

Here S is a set of states, s_0 an inital state and $\tau : S \times I \to S$ describes a deterministic transition function and $\ell : S \to J$ a labelling of S. Semantically, $\mathcal{T}$ starts in s_0 and reacts to an input i by moving to $z = \tau(s, i)$ and outputting $\ell(z)$. Naturally, such a transition

Figure 13. Game graph for a POSG that resolves around the PBA defined in Figure 12. Adam and Eve only observe the layer they play in but never the Q-component. On the other hand, regarding the evaluation every state with common Q-component is treated equally, hence the underlaying PBA only observes the colors. The local storage of the states of the underlying PBA are dismissed since the stored input value already uniquely identifies the "correct" state. Thus, technically we have e.g. q_0^0 and q_0^0 but the choice of Adam e only allows movements to q_0^e, hence we dismissed unreachable states. Similarly, we removed the unreachabe (i, q_0, j) for the *original* q_0 state (not for the q_0^0, q_0^1 states).



system $\mathcal{T}$ models a strategy $t : I^+ \to J$ by constructing for every $u = u_1 \dots u_n \in I^+$ the unique sequence $s_0 s_1 \dots s_n \in S^+$ such that $s_{i+1} = \tau(s_i, u_{i+1})$ for $0 \le i < n$ and returning $\ell(s_n)$. As mentioned in the introduction of Section 3.1.1 alternating tree automata can be used to run on the unfolding of such transition systems. Assuming a ω-regular specification ϕ we may obtain a DPA $\mathcal{P} = (Q, q_0, (J \times I), \delta, \mathrm{par})$ accepting $\mathcal{L}_\phi$. We can attach $\mathcal{P}$ onto a given transition system $\mathcal{T}$ in a straightforward manner to obtain a transition system with an associated Parity-condition (but dropping the labelling)

$$\mathcal{G}_{\mathcal{P}}^{\mathcal{T}} = (S \times Q, (s_0, q_0), \tau', \mathrm{par}')$$

where $\tau'((s,q), i) = (\tau(s,i), \delta(q, (\ell(s), i)))$ and $\mathrm{par}'((s,q)) = \mathrm{par}(q)$. For this transition system and an ϵ-environment $\mathcal{E}$ we can identify structural properties in $\mathcal{G}_{\mathcal{P}}^{\mathcal{T}}$ to determine whether $\mathcal{T}$ satisfies ϕ almost-surely respectively positively. This structural property is connected with Strongly-Connected Component (SCC). These are subsets of nodes G such that there is a path between every two nodes $u, v \in G$ (cp. [Tar72]). Additionally, a leaf-SCC is an SCC S such that there is no other SCC reachable from any $v \in S$ (cp. [CY95, Bottom-SCC]). Intuitively, it is unlikely to not eventually end up in a leaf-SCCs since there is infinitely often the possibility to take a non-zero probability into smaller SCCs. Hence we state

Lemma 27. *[Sch06, Lemma 1] An J-labeled I-transition-system $\mathcal{T}$ almost-surely (positively) satisfies a specification ϕ if and only if the highest priority in all (some) reachable leaf-SCC of $\mathcal{G}_{\mathcal{P}}^{\mathcal{T}}$ is even, where $\mathcal{P}$ is a DPA accepting $\mathcal{L}_\phi$.*

This structual property can be checked by a APTA. This again yields an algorithm for the synthesis problem of almost-sure (respectively positive) satisfaction of an ω-regular ϕ by checking emptiness of the obtained APTA. Therefore, we obtain

Theorem 38. *[Sch06, Theorem 1] Given a DPA $\mathcal{P}$ there is a APTA $\mathcal{A}_{\mathcal{P}}$ ($\mathcal{O}_{\mathcal{P}}$) such that $\mathcal{A}_{\mathcal{P}}$ ($\mathcal{O}_{\mathcal{P}}$) accepts a J-labeled I-transition-system $\mathcal{T}$ if and only if[2] $\mathcal{T}$ satisfies almost-surely (positively) $\mathcal{P}$.*

If $\mathcal{P}$ has n states and c parities then $\mathcal{A}_{\mathcal{P}}$ and $\mathcal{O}_{\mathcal{P}}$ have at most $n \cdot \lceil 2 + \frac{c}{2} \rceil + 1$ states.

Proof-Sketch. The main argument for this proof is that the search for the leaf-SCCs can be understood as a two-player-game. One player plays for acceptance (called Acceptor) while the other tries to spoil the correctness of the transition system (called Spoiler). We describe the case of $\mathcal{A}_{\mathcal{P}}$ but the case $\mathcal{O}_{\mathcal{P}}$ can be constructed analogously by exchanging the player role in the first phase of the game. The game for $\mathcal{T}$ in the $\mathcal{A}_{\mathcal{P}}$ operates in three phases:

1. Spoiler chooses one leaf-SCC S,
2. Acceptor chooses one priority p of a node in S,
3. Spoiler tries to find a higher priority than p in S.

The unrollment of $\mathcal{T}$ yields an I-ary J-tree. The states of $\mathcal{A}_{\mathcal{P}}$ encode the current state of $\mathcal{P}$ and if the game is in the first, second or third phase. If the game is in the third phase the chosen priority is also stored in the state. While operating in the first phase $\mathcal{A}_{\mathcal{P}}$ dispatches in all directions first-phase states, representing possible choices of Spoiler and also in one direction a second-phase state; that is a second phase state which is only dispatched into one direction, representing the choice of Acceptor. This second-phase state may transform at any point to a third-phase state and store the (even) priority of the current state of $\mathcal{P}$ in the state of $\mathcal{A}_{\mathcal{P}}$. This third-phase state again dispatches in all directions and as soon as one state with a higher parity than

[2] The original theorem only states the if-direction but the corresponding proof deals with the only if-direction as well.

the stored one appears, moves to a sink-error-state. The first-phase states are given a parity of 0, the second-phase states of 1 and the third-phase states of 2. This allows that for all choices of Spoiler (which constantly reproduce themselves) only those that proceed to the second phase actually matter or for the third-phase only the occurence of a higher parity. Since the parity of the second-phase states is odd it is enforced that Acceptor eventually chooses one parity to store. Note the similarity to the proof of Proposition 4 in formulating the desired winning strategy of one player as actual choices which may use the non-determinism of the automaton while the Spoiler explores all possible moves. □

This allows us to solve the associated synthesis problem for ω-regular specifications and ϵ-environments. By solving emptiness for a APTAs.

On the other hand, we consider in the following environments that are represented by MDPs or POMDPs and try to synthesis for a given specification a strategy to almost-surely or positively satisfy this specification. This approach is conceptually different to ϵ-environments since concrete probabilities are given and also the interaction of the actor with the environment is explicitly modelled. We formalize the synthesis question with (cp. [CDH10; CJH03; CY95])

Definition 33.
Given a POMDP
$$\mathcal{M} = \left(S, s_0, A, (\tau_a)_{a\in A}, \sim\right)$$
and a specification $\phi \subseteq S^\omega$ such that $\phi \in \mathcal{B}(S)$. The qualitative synthesis problem (M, ϕ) demands the computation of a strategie $s : [S]^*_\sim \to A$ such that $\mu_s(\phi) = 1$.

Theorem 20 and Theorem 21 already formulated positive results for this question. Again, we focus on ω-regular ϕ (in this case ϕ is itself a language and is not translated to one). Note that the formulation in Theorem 20 and Theorem 21 use acceptance conditions directly onto the states of the POMDP and MDP respectively. Initially, we observe that these formulations are interchangable: In analogy to the constructions in [CY95; CY90] we fix a POMDP
$$\mathcal{M} = \left(S, s_0, A, (\tau_a)_{a\in A}, \sim\right)$$
and a ω-regular $\phi \subseteq S^\omega$. Let $\mathcal{P} = (Q, S, q_0, \delta, \mathrm{par})$ be a DPA which accepts ϕ. We construct
$$\mathcal{M} \otimes \mathcal{P} = \left(S \times Q, (s_0, q_0), A, (\tau'_a)_{a\in A}, \sim', \mathrm{par}'\right)$$
with
$$\tau'_a((s,q), (z,p)) = \begin{cases} \tau_a(s,z) & \text{if } p = \delta(q,s), \\ 0 & \text{otherwise,} \end{cases}$$
$$\mathrm{par}'((s,q)) = \mathrm{par}(q) \text{ and } (s,q) \sim' (z,p) \text{ iff } s \sim z.$$

We observe that the definition of $\sim'$ allows to identify $[(s,q)]_{\sim'}$ with $[s]_{\sim}$ and therefore, the strategy spaces for $\mathcal{M}$ and $\mathcal{M} \otimes \mathcal{P}$ coincide. Central to the following argument is the following

Lemma 28. *For any strategy $s : [S]^*_{\sim} \to A$ the resulting probability space $(S^\omega, \mathcal{B}(S), \mu_s)$ can be isomorphically embedded into $((S \times Q)^\omega, \mathcal{B}(S \times Q), \mu'_s)$.*

Proof. We define the following function $f : S^\omega \to (S \times Q)^\omega$ with

$$f(s_1 s_2 \dots) = (s_1, q_1)(s_2, q_2) \dots \text{ such that } \delta(q_{i-1}, s_{i-1}) = q_i \text{ for all } i > 0.$$

Moreover, we consider the two isomorphic probability spaces

$$(S^\omega, \mathcal{B}(S), \mu_s)$$

and

$$\Big(f(S^\omega), \mathcal{B}(S \times Q)_{|_{f(S^\omega)}}, \mu_s \circ f^{-1}\Big)$$

by the isomorphism f. By definition of τ'_a for all $a \in A$ and inductive reasoning, we observe

$$\mu'_s(\mathrm{cyl}((s_1,q_1)\dots(s_n,q_n))) = \begin{cases} \prod_{0 \le i < n} \tau_{s(s_1 \dots s_i)}(s_i, s_{i+1}) & \substack{\text{for the unique parital run} \\ q_0 \dots q_n \text{ of } \mathcal{P} \text{ on } s_0 \dots s_n}, \\ 0 & \text{otherwise.} \end{cases}$$

Hence, it is easy to see that on the generating sets of $\mathcal{B}(S \times Q)_{|_{f(S^\omega)}}$ the measures μ'_s and $\mu_s \circ f$ coincide. Therefore, the claim follows by unique extension of μ'_s and $\mu_s \circ f^{-1}$. $\square$

This leads to

Theorem 39. *The problems to compute a strategy for a POMDP with states S which almost-surely (positively) satisfies an ω-regular strategy $\phi \subseteq S^\omega$ and to compute a strategy for a POMDP which almost-surely (positively) satisfies an associated Parity-condition are effectively reducable to each other.*

Proof. It is easy to obtain from an associated Parity-condition an ω-regular language by simply considering Acc(par) for which the measures coincide.

For the converse, we consider $\mathcal{M} \otimes \mathcal{P}$ and use Lemma 28. Moreover, we observe that $f(\phi) = \mathrm{Acc}(\mathrm{par}) \cap f(S^\omega)$ which entails $\mu_s(\phi) = \mu'_s(\mathrm{Acc}(\mathrm{par}))$. $\square$

If we consider the construction $\mathcal{M} \otimes \mathcal{P}$ for a MDP $\mathcal{M}$ we observe by the uniqueness of the associated state sequences that the argumentation of Lemma 28 still applies if we consider $\mathcal{M} \otimes \mathcal{P}$ to be a MDP as well. This allows to state

Corollary 13. *The problems to compute a strategy for a MDP with states S which almost-surely (positively) satisfies an ω-regular strategy $\phi \subseteq S^\omega$ and to compute a strategy for a MDP which almost-surely (positively) satisfies an associated Parity-condition are effectively reducable to each other.*

Moreover, if we consider ϕ to be recognizable by a DBA we state by the same argument

Corollary 14. *The problems to compute a strategy for a POMDP with states S which almost-surely (positively) satisfies a DBA-recognizable $\phi \subseteq S^\omega$ and to compute a strategy for a POMDP which almost-surely (positively) satisfies an associated Büchi-condition are effectively reducable to each other.*

We note here that by the results of Theorem 20 and Theorem 21 it is possible to synthesis strategies for positive and almost-sure satisfaction of ω-regular ϕ in MDPs and almost-sure satisfaction for DBA-recognizable ϕ in POMDPs. The conceptual idea of Lemma 27 is essential for the algorithm of Theorem 21 by identifying strategies which move into SCCs which are winning for Eve [CJH04]. On the other hand, we observe that for POMDPs and almost-sure satisfaction we are restricted to DBA-recognizable specifications by the undecidability results of Corollary 3. However, we consider ϕ which are defined in terms of almost-surely accepting PBAs. By Proposition 1 we know that almost-surely accepting PBAs subsumes the expressiveness of DBAs. Hence, solving the synthesis problem for ϕ which can be expressed as languages of almost-surely accepting PBAs is a generalisation of Corollary 14 by the stronger expressiveness of almost-surely accepting PBAs over DBAs. Formally, we examine the following problem:

Definition 34. ***PBA-Synthesis Question***:
Given a POMDP $\mathcal{M}$ and a PBA $\mathcal{P}$ with

$$\mathcal{M} = \left(S, s_0, A, (\tau_a)_{a \in A}, \sim\right) \text{ and } \mathcal{P} = (Q, q_0, S, \delta, F)\,.$$

Does a strategy $s : [S]^*_\sim \to A$ exist such that almost-all executions of $\mathcal{A}$ under s are almost-surely accepted by $\mathcal{P}$?

Considering a certain restricted class of MDPs, namely those where for every state ever following state is defined by a probability distribution B (regardless of the action chosen by the player), then this result can be derived as a corollary from the construction examined in Example 11. This construction is the main inspiration of the following argument. Mainly, we unroll the run-trees of PBA along state sequences of an MDP. Recalling that in Theorem 35 the inputs are identifies with the directions along a tree, this idea presents very similar. We strengthen our result to apply for POMDPs by merging directions of equivalent observation.

For the rest of the section, we fix one POMDP $\mathcal{M}$ and a PBA $\mathcal{P}$ with

$$\mathcal{M} = \left(S, s_0, A, (\tau_a)_{a \in A}, \sim\right) \text{ and } \mathcal{P} = (Q, q_0, S, \delta, F)\,.$$

Additionally, we consider one strategy $s : [S]^*_\sim \to A$ which induces a probability space for $\mathcal{M}$

$$(S^\omega, \mathcal{B}(S), \mu_s)\,.$$

Furthermore, for any given $\alpha \in S^\omega$ the stochastic process of $\mathcal{P}$ also yields a probability space

$$(Q^\omega, \mathcal{B}(Q), \mu_\alpha)\,.$$

We approach the mathematical analysis of this problem by defining the following Markov-kernel

$$K : S^\omega \times \mathcal{B}(Q) \to [0,1] \text{ with } K(\alpha, A) = \mu_\alpha(A)$$

for the measurable spaces $(S^\omega, \mathcal{B}(S))$ and $(Q^\omega, \mathcal{B}(Q))$. Initially, we observe

Lemma 29. *K as defined above is a Markov-kernel.*

Proof. With $K(\alpha, \cdot) = \mu_\alpha$ it is trivial to state that $K(\alpha, \cdot)$ defines a probability measure for $(Q^\omega, \mathcal{B}(Q))$.

By Lemma 5 it suffices to check that $K(\cdot, B)$ is $\mathcal{B}(S)$-measurable only for cylindric sets B, i.e. $B = \text{cyl}(u)$ for some $u \in S^*$.

We observe that by definition of the transition probabilities for a PBA we obtain a determinism for the run-tree of $\mathcal{P}$. This means that for any $u \in S^n$ and $\alpha, \beta \in \text{cyl}(u)$ holds

$$\mu_\alpha(\text{cyl}(v)) = \mu_\beta(\text{cyl}(v)) \text{ for all } v \in \cup_{1 \le i \le n} Q^i.$$

This justifies the use of the notion μ_u for the probabilities of all cylindric sets $A = \text{cyl}(v)$ for all $v \in Q^*$ with $|v| \le |u|$. By Theorem 3 the measurability of $K(\cdot, \text{cyl}(u))$ is implied if for all $a \in \mathbb{R}$

$$\{\alpha \in S^\omega \mid K(\alpha, \text{cyl}(u)) \le a\} \in \mathcal{B}(S).$$

This follows from the representation as

$$\{\alpha \in S^\omega \mid K(\alpha, \text{cyl}(u)) \le a\} = \bigcup \left\{\text{cyl}(v) : v \in S^{|u|} \text{ with } \mu_v(\text{cyl}(u)) \le a\right\}$$

since $S^{|v|}$ is finite. Concludingly, we obtain that K is indeed a Markov-kernel. □

Observing that

$$K(\cdot, \text{Acc}(F))^{-1}(1) = \{\alpha \in S^\omega \mid \mu_\alpha(\text{Acc}(F)) = 1\}$$

is the set of executions which are almost-surely accepted by $\mathcal{P}$, allows us to state for K and Lemma 2

$$\int_{S^\omega} K(\cdot, \text{Acc}(F)) d\mu_s = \int_{\alpha \in S^\omega} \mu_\alpha(\text{Acc}(F)) d\mu_s = 1 \text{ iff } \mu_s(K(\cdot, \text{Acc}(F))^{-1}(1)) = 1.$$

Hence, a strategy s is an answer to the synthesis question of Definition 34 if and only if $\int_{S^\omega} K(\cdot, \text{Acc}(F)) d\mu_s = 1$. Using Theorem 7 allows us to refine this notion further. Considering the probability space $(S^\omega \times Q^\omega, \mathcal{B}(S) \otimes \mathcal{B}(Q), \mu_s \otimes K)$, we obtain that

$$\int_{S^\omega} K(\cdot, \text{Acc}(F)) d\mu_s = (\mu_s \otimes K)(S^\omega, \text{Acc}(F)).$$

Concludingly giving

Lemma 30. *The strategy s is an answer to an instance of the synthesis question of Definition 34 if and only if*

$$(\mu_s \otimes K)(S^\omega, \mathrm{Acc}(F)) = 1.$$

We proceed by defining a choiceless WDTA which captures this product space:

Definition 35. *Synthesis WDTA*:
For a POMDP $\mathcal{M}$ and a PBA $\mathcal{P}$ with

$$\mathcal{M} = \left(S, A, (\tau_a)_{a \in A}, s_0, \sim\right) \text{ with } \mathcal{O} = \{[s]_\sim : s \in S\} \text{ and } \mathcal{P} = (Q, S, \delta, q_0, F)$$

we construct the choiceless WDTA

$$\mathcal{A}_{\mathcal{M}}^{\mathcal{P}} = (S \times Q, (s_0, q_0), A, \mathcal{O}, \Delta, S \times F).$$

The transitions in Δ are of the form

$$\left((s,q), a, G^a_{(s,q)}\right) \text{ with } G^a_{(s,q)}((z,p), o) = \begin{cases} \tau_a(s,z) \cdot \delta(q,z,p) & \text{if } [z]_\sim = o, \\ 0 & \text{otherwise.} \end{cases}$$

Similar to the construction in Section 3.3.3, all trees for this WDTA $\mathcal{A}_{\mathcal{M}}^{\mathcal{P}}$ are valid strategies for the POMDP $\mathcal{M}$ and vice versa. For one tree s we fix the unique run r of $\mathcal{A}_{\mathcal{M}}^{\mathcal{P}}$ on s. We formulate the following essential

Lemma 31. *The probability space $(S^\omega \times Q^\omega, \mathcal{B}(S) \otimes \mathcal{B}(Q), \mu_s \otimes K)$ can be isomorphically embedded into $(((S \times Q) \times O)^\omega, \mathcal{B}((S \times Q) \times O), \mu_r)$.*

Proof. Initially, we consider an isomorphism b between $(S^\omega \times Q^\omega, \mathcal{B}(S) \otimes \mathcal{B}(Q), \mu_s \otimes K)$ and $((S \times Q)^\omega, \mathcal{B}(S \times Q), \mu_s \otimes K \circ b^{-1})$. Therefore, we define

$$b : S^\omega \times Q^\omega \to (S \times Q)^\omega \text{ with } b(\alpha_1\alpha_2 \ldots, \beta_1\beta_2 \ldots) = (\alpha_1, \beta_1)(\alpha_2, \beta_2) \ldots.$$

Lemma 3 states that $\mathcal{B}(S) \otimes \mathcal{B}(Q)$ is generated by balanced cylindric sets. Considering two words $u = a_1 \ldots a_n \in A^*$ and $v = b_1 \ldots b_n \in B^*$, allows to observe that

$$b(\mathrm{cyl}(u) \times \mathrm{cyl}(v)) = \mathrm{cyl}((a_1, b_1) \ldots (a_n, b_n)).$$

Therefore, we get the required measurability for b and b^{-1} by construction of measurable sets in the Borel-algebras. Hence, the claimed isomorphism follows by definition of the measure $\mu_s \otimes K \circ b^{-1}$.

Using Lemma 6 with the function obs : $S \times Q \to O$ such that $\mathrm{obs}(s,q) = [s]_\sim$ yields an isomorphic embedding of

$$(S^\omega \times Q^\omega, \mathcal{B}(S) \otimes \mathcal{B}(Q), \mu_s \otimes K)$$

into

$$\left(((S\times Q)\times O)^{\omega},\mathcal{B}((S\times Q)\times O),\mu_s\otimes K\circ b^{-1}\circ\mathrm{lift}_{\mathrm{obs}}^{-1}\right).$$

Hence, it suffices to show that $\mu_s\otimes K\circ b^{-1}\circ\mathrm{lift}_{\mathrm{obs}}^{-1}$ coincides with μ_r on all cylindric sets $\mathrm{cyl}(((s_1,q_1),o_1)\dots((s_n,q_n),o_n))$. Inductive reasoning on all generators in $\mathcal{G}(\mathcal{A}_{\mathcal{M}}^{\mathcal{P}})$ shows that weight only moves along paths such that the O-component agrees with the observation of the S-component. Therefore, we state if $o_i=[s_i]_{\sim}$ for all $1\le i\le n$

$$\begin{aligned}
&\mu_r(\,\mathrm{cyl}(((s_1,q_1),o_1)\dots((s_n,q_n),o_n)))\\
&=G_{(s_0,q_0)}^{s(\epsilon)}(((s_1,q_1),o_1))\cdot\prod_{i=1}^{n-1}G_{(s_i,q_i)}^{s(o_1\dots o_i)}((s_{i+1},q_{i+1}),o_{i+1})\\
&=\delta(q_0,s_1,q_1)\cdot\tau_{s(\epsilon)}(s_0,s_1)\prod_{i=1}^{n-1}\delta(q_i,s_{i+1},q_{i+1})\cdot\tau_{s(o_1\dots o_n)}(s_i,s_{i+1})\\
&=\mu_{s_1\dots s_n}(\mathrm{cyl}(q_0\dots q_n))\cdot\mu_s(\mathrm{cyl}(s_1\dots s_n))\\
&=\int_{\mathrm{cyl}(s_1\dots s_n)}K(\cdot,\mathrm{cyl}(q_0\dots q_n))d\mu_s.
\end{aligned}$$

By the observation that

$$\mathrm{cyl}(((s_1,q_1),o_1)\dots((s_n,q_n),o_n))=\mathrm{lift}_{\mathrm{obs}}\circ b(\mathrm{cyl}(s_1\dots s_n),\mathrm{cyl}(q_1\dots q_n))$$

and the uniqueness of $\mu_s\otimes K$ and the extension of μ_r, the claim follows. □

Concludingly, we state

Theorem 40 (Qualitative PBA Synthesis)**.** *The synthesis question of an environment $\mathcal{M}$ and a specification provided as almost-surely accepting PBA $\mathcal{A}$ is decidable.*

Proof. We claim that constructing $\mathcal{A}_{\mathcal{M}}^{\mathcal{P}}$ and deciding its emptiness (Corollary 10) yields the required decision procedure. Therefore, we observe

$$(\mathrm{lift}_{\mathrm{obs}}\circ b)\,(S^{\omega},\mathrm{Acc}(F))=(\mathrm{Acc}\,[S\times Q]\,(S\times F))\cap(\mathrm{lift}_{\mathrm{obs}}\circ b)(S^{\omega},Q^{\omega})$$

which implies for the unique run r on a tree t that

$$\mu_r(\mathrm{Acc}\,[S\times Q]\,(S\times F))=\mu_s\otimes K(S^{\omega},\mathrm{Acc}(F)).$$

Hence, with Lemma 30 we have that $\mathcal{A}_{\mathcal{M}}^{\mathcal{P}}$ almost-surely accepts t if and only if t solves the synthesis question imposed by $\mathcal{M}$ and $\mathcal{P}$. □

Although inspired by the construction of Theorem 32, the usage of WDTAs is not essential for the argument above. Equivalently Theorem 39 can be extended towards ϕ which are defineable by almost-surely accepting PBAs by the natural product construction: we consider a POMDP $\mathcal{M}$ and a PBA $\mathcal{P}$ with

$$\mathcal{M}=\left(S,s_0,A,(\tau_a)_{a\in A},\sim\right)\ \text{and}\ \mathcal{P}=(Q,q_0,S,\delta,F)$$

and construct a product POMDP

$$\mathcal{M} \otimes \mathcal{P} = \left(S \times Q, (s_0, q_0), A, (\tau'_a)_{a \in A}, \sim', F' = S \times F\right)$$

with

$$\tau'_a((s,q),(z,p)) = \tau_a(s,z) \cdot \delta(q,s,p) \text{ and } (s,q) \sim' (z,p) \text{ iff } s \sim z.$$

This argument removes the use of $\mathrm{lift}_{\mathrm{obs}}$ in the proof of Lemma 31 and renders for any strategy s which are transferable by the choice of $\sim'$ the probability spaces for $\mathcal{M}$ with measure $\mu_s \otimes K$ and $\mathcal{M} \otimes \mathcal{P}$ with measure μ_s isomorphic. The correctness argument is analogous to the argument which involve WDTAs above. In this setup MDPs are also transformed to POMDPs since the construction needs to hide the current state of $\mathcal{P}$. Specifically, PBAs have in general possibly multiple following states in their run-tree. Nevertheless, this yields a POMDP with states $S' = Q \times S$ and allows to compute the corresponding s in time exponential of $|S'|$. Effectively, we get

Corollary 15. *If a strategy for the qualitative synthesis problem for a POMDP with states S and a specification in form of an almost-surely accepting PBA with states Q exists it can be computed in time exponential in $|Q \times S|$.*

This is a notable improvement over the argument resolving around WDTAs since the associated time complexity there is exponential in $|S| \times |Q| \times |[S]_\sim|$ (cp. Corollary 10).

5. Conclusion

5.1. Evaluation

WDTAs present with a complex theory and strong capabilities to model other probabilistic systems. Namely, even the restricted class of choiceless WDTAs subsumes various stochastic models such as POMDPs or PBAs. Hence reducing the complexity of WDTAs by restriction to choicelessness seems a valuable approach. Neverthelss, we argue that the theory of choiceless WDTAs is not itself subsumed by the theory of POMDPs although we established strong connections between these models. Notably, there is a subtle difference in semantics of the considered objects. WDTAs provide a tool to argue about languages of trees and therefore induce questions about collections of elements where strategies for POMDPs are more considered in an existential setting, i.e. exists a strategy that satisfy certain conditions. The view on collections of trees or strategies and the associated notions of e.g. closure properties raises questions about strategies that are well-behaved for more than one structure (or even for the same structure with different objectives).

Additionally, WDTAs have proven a useful tool to translate the rich theory of word and tree automata and their connections to the probabilistic setting. For example we see that the weighted unrollment of PBAs run-trees along the paths in a WDTA mirrors the parallel executions of word automata along tree automata. Interestingly, we unvail in this way the chimeric nature of PBAs regarding non-determinism. The usage of probabilities to resolve non-determinism allows to use "deterministic" run-trees (which is important for the parallel execution on paths) but still allow to model decisions of the automaton. Notably, this allows to skip the determinisation step that is usually required for the specifying automaton (cp. ω-regular languages). This determinisation is inherently costly. However, it is highly non-trivial to formulate specifications in almost-surely accepting PBAs. Therefore, we note that the obtained synthesis results are ought to be considered intermediate. In contrast to ω-regular languages which capture linear time logics [BK08] and the monadic second-order theory of linearly ordered sets [Büc90], languages that can be accepted by almost-surely accepting PBAs do not have an immediate (more accessible) formulation.

On the other hand, we consider solving the synthesis question for an antagonistic domain against a probabilistic specification an intriguing result. It allows to contain the probabilistic aspects of the problem within the specification. This is especially interesting since the probabilistic behavior of the specification conceptually distinguishes it from e.g. ω-regular languages. Additionally, observing probabilistic specifications for POMDPs, we are able to push the probabilistic behavior into the structure of the POMDP cleansing the specification from it. This embeds nicely into the de-randomisation reductions of [Cha+15] where is also noted that strategies for EVE can be cleansed from randomisation.

5.2. Future Work

Naturally, the complex theory of WDTAs cannot be thoroughly explored in all its facets in the limited scope of this thesis. For various restricted and the unrestricted classes there are open questions about closure properties and expressiveness.

Beside this, we want to motivate research around the non-deterministic choices of WDTAs. Considering a given tree t, we can adapt Definition 19 to design an acceptance game as MDP with states $Q \times D^*$ where EVE decides which transitions (compatible with t) to take. Unfortunately, it is not immediately clear how to obtain an emptiness game from this construction as it is explained in Example 9. Giving EVE control of the tree and the run entails that she must construct a consistent tree but depending on the current state might choose different transitions. This induces requirements on the observations of EVE which are not fitting the concept of partial observability as discussed in this thesis. Therefore, we propose to consider co-operative POSGs where e.g. EVE constructs the tree while ADAM complementarily builds an associated tree. The game is won for both players if almost-surely or positively the acceptance condition of the associated WDTA is satisfied. This approach maybe allows to use the different observations for both players to obtain a consistent tree and a run for this tree which utilizes the complete expressiveness of WDTAs. This is especially interesting in the context of the synthesis problem for logics that reason about the unrollment of transition systems, e.g. CTL. In [KV99] alternating tree automata are used to formulate synthesis results for CTL which even incorporate unobservable information. Additionally, this approach might reveal how the Simulation Theorem (Theorem 26) for alternating tree automata translates into the weighted case.

It is also possible to use non-deterministic choices in other probabilistic automata, e.g. PBAs. This yields an automaton which upon reading a letter chooses non-deterministically a probability distribution. This distribution weights the move into its next state. This model subsumes ω-regular languages by using the non-determinism to choose from Dirac distributions (for both positive and almost-sure semantics) and includes the properties of PBAs with positive or almost-sure semantics respectively. Initially, since non-determinism is sufficient to use Büchi-conditions for all ω-regular languages this raises the questions if this holds in the probabilistic case as well. Analogously to [GTW02, Theorem 1.10] we obtain an equivalent NBA for a given NPA by introducing for every winning parity a copy of the automaton which lacks transitions to higher odd parities and marking in this copy the states with the associated winning partiy as Büchi-states. Hence, the NBA guesses the moment from which on no higher odd parity is visited anymore and moves into the "correct" copy. An immediate translation of this construction to the probabilistic case does not work for the almost-sure case: the probability of loosing runs in a probabilistic word automaton with Parity-condition and almost-sure acceptance might diminish constantly for a winning run. Nevertheless, a movement into one of the copies for a winning parity occurs after finitely many steps. Therefore, there might be no situation after finitely many steps for such a movement which does not allow for a measurable set of "failing" runs. In [CHS14] the *value* of a tree t in an automaton $\mathcal{A}$ is defined as the supremum of the measure of all accepting paths over all possible runs of $\mathcal{A}$ for t. This notion can be used here as well and our construction suggests that the value of α is 1 in the Büchi-automaton if and only if it is accepted by the original probabilistic automaton with Parity-condition.

Hence, we expect interesting results on the topic of WDTAs from further investigation of co-operative POSG which might be used as emptiness games for WDTAs. Also, the further study of WDTAs might reveal new connections to POSGs as well. Additionally, we propose to consider automata with non-determistic as well as probabilistic choices.

Moreover, considering the presented synthesis results it is a natural next step to develop a logic to express the specification which can be translated to almost-surely accepting PBAs. This is an important step for carrying the formulated synthesis results into the real world. Initial work on this topic is done in [Wei12] where PMAs are connected with weighted monadic second order logic. Also, the complexity for the synthesis problem associated with a specification given as almost-surely accepting PBA against an antagonistic environment can be further explored. We rely on results which consider Adam more informed than Eve. In this case, for both Adam and Eve a subset-construction is used to determine winning strategies. In our specific setting we hope to save one exponential with an algorithm that makes use of the fact that Eve and Adam are equally informed.

5.3. Closing Remarks

Concludingly, the results of this thesis are twofold. Mainly, we introduced WDTAs as a class of automata on infinite trees that incorporate the concept of weighted paths and alternation. Its close connection to POMDPs and PBAs unveiled positive and negative results for the decidability of emptiness problems. More importantly, WDTAs have proven an interesting model by their expressiveness and complex nature.

Secondly, WDTAs are an essential source of ideas to formulate and prove synthesis results for antagonistic and probabilistic environments against specifications given by PBAs. Most notably, we obtained decidability results for the following problems:

1. Given a specification as almost-surely accepting PBA. Compute a deterministic algorithm that *always* satisfies the given specification or prove that such an algorithm cannot exist.

2. Given a specification as almost-surely accepting PBA and a POMDP. Compute a determinisitc strategy such that almost-all executions under this strategy satisfy the posed specification or prove that such a strategy cannot exist.

Bibliography

[ABK11] Shaull Almagor, Udi Boker, and Orna Kupferman. "What's Decidable about Weighted Automata?" In: *Automated Technology for Verification and Analysis, 9th International Symposium, ATVA 2011, Taipei, Taiwan, October 11-14, 2011. Proceedings.* 2011, pp. 482–491. DOI: 10.1007/978-3-642-24372-1_37. URL: https://doi.org/10.1007/978-3-642-24372-1%5C_37.

[AK11] Shaull Almagor and Orna Kupferman. "Max and Sum Semantics for Alternating Weighted Automata". In: *Automated Technology for Verification and Analysis, 9th International Symposium, ATVA 2011, Taipei, Taiwan, October 11-14, 2011. Proceedings.* 2011, pp. 13–27. DOI: 10.1007/978-3-642-24372-1_2. URL: https://doi.org/10.1007/978-3-642-24372-1%5C_2.

[AK16] Shaull Almagor and Orna Kupferman. "High-Quality Synthesis Against Stochastic Environments". In: *25th EACSL Annual Conference on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France.* 2016, 28:1–28:17. DOI: 10.4230/LIPIcs.CSL.2016.28. URL: https://doi.org/10.4230/LIPIcs.CSL.2016.28.

[Bai+04] Christel Baier et al. "Controller Synthesis for Probabilistic Systems". In: *Exploring New Frontiers of Theoretical Informatics, IFIP 18th World Computer Congress, TC1 3rd International Conference on Theoretical Computer Science (TCS2004), 22-27 August 2004, Toulouse, France.* 2004, pp. 493–506. DOI: 10.1007/1-4020-8141-3_38. URL: https://doi.org/10.1007/1-4020-8141-3_38.

[Bau92] Heinz Bauer. *Maß- und Integrationstheorie.* De Gruyter, 1992. ISBN: 978-3-11-013625-8.

[BBG08] Christel Baier, Nathalie Bertrand, and Marcus Größer. "On Decision Problems for Probabilistic Büchi Automata". In: *Foundations of Software Science and Computational Structures, 11th International Conference, FOSSACS 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29 - April 6, 2008. Proceedings.* 2008, pp. 287–301. DOI: 10.1007/978-3-540-78499-9_21. URL: https://doi.org/10.1007/978-3-540-78499-9_21.

[BBG09] Christel Baier, Nathalie Bertrand, and Marcus Größer. "The Effect of Tossing Coins in Omega-Automata". In: *CONCUR 2009 - Concurrency Theory, 20th International Conference, CONCUR 2009, Bologna, Italy, September 1-4, 2009. Proceedings.* 2009, pp. 15–29. DOI: 10.1007/978-3-642-04081-8_2. URL: https://doi.org/10.1007/978-3-642-04081-8%5C_2.

[BD08] Dietmar Berwanger and Laurent Doyen. "On the Power of Imperfect Information". In: *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2008, December 9-11, 2008, Bangalore, India*. 2008, pp. 73–82. DOI: 10.4230/LIPIcs.FSTTCS.2008.1742. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2008.1742.

[BG05] Christel Baier and Marcus Größer. "Recognizing omega-regular Languages with Probabilistic Automata". In: *20th IEEE Symposium on Logic in Computer Science (LICS 2005), 26-29 June 2005, Chicago, IL, USA, Proceedings*. 2005, pp. 137–146. DOI: 10.1109/LICS.2005.41. URL: https://doi.org/10.1109/LICS.2005.41.

[BGK17] Mikołaj Bojańczyk, Hugo Gimbert, and Edon Kelmendi. "Emptiness of Zero Automata Is Decidable". In: *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*. 2017, 106:1–106:13. DOI: 10.4230/LIPIcs.ICALP.2017.106. URL: https://doi.org/10.4230/LIPIcs.ICALP.2017.106.

[BK08] Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT Press, 2008. ISBN: 978-0-262-02649-9.

[BL69] J. Richard Buchi and Lawrence H. Landweber. "Solving Sequential Conditions by Finite-State Strategies". In: *Transactions of the American Mathematical Society* 138 (1969), pp. 295–311. ISSN: 00029947. URL: http://www.jstor.org/stable/1994916.

[BNN91] Danièle Beauquier, Maurice Nivat, and Damian Niwinski. "About the Effect of the Number of Successful Paths in an Infinite Tree on the Recognizability by a Finite Automaton with Büchi Conditions". In: *Fundamentals of Computation Theory, 8th International Symposium, FCT '91, Gosen, Germany, September 9-13, 1991, Proceedings*. 1991, pp. 136–145. DOI: 10.1007/3-540-54458-5_58. URL: https://doi.org/10.1007/3-540-54458-5%5C_58.

[Büc90] J. Richard Büchi. "On a Decision Method in Restricted Second Order Arithmetic". In: *The Collected Works of J. Richard Büchi*. Ed. by Saunders Mac Lane and Dirk Siefkes. New York, NY: Springer New York, 1990, pp. 425–435. ISBN: 978-1-4613-8928-6. DOI: 10.1007/978-1-4613-8928-6_23. URL: https://doi.org/10.1007/978-1-4613-8928-6_23.

[CD14] Krishnendu Chatterjee and Laurent Doyen. "Partial-Observation Stochastic Games: How to Win when Belief Fails". In: *ACM Trans. Comput. Log.* 15.2 (2014), 16:1–16:44. DOI: 10.1145/2579821. URL: http://doi.acm.org/10.1145/2579821.

[CDH10] Krishnendu Chatterjee, Laurent Doyen, and Thomas A. Henzinger. "Qualitative Analysis of Partially-Observable Markov Decision Processes". In: *Mathematical Foundations of Computer Science 2010, 35th International Symposium, MFCS 2010, Brno, Czech Republic, August 23-27, 2010. Pro-*

ceedings. 2010, pp. 258–269. DOI: 10.1007/978-3-642-15155-2_24. URL: https://doi.org/10.1007/978-3-642-15155-2_24.

[Cha+15] Krishnendu Chatterjee et al. “Randomness for free”. In: *Inf. Comput.* 245 (2015), pp. 3–16. DOI: 10.1016/j.ic.2015.06.003. URL: https://doi.org/10.1016/j.ic.2015.06.003.

[CHS11] Arnaud Carayol, Axel Haddad, and Olivier Serre. “Qualitative Tree Languages”. In: *Proceedings of the 26th Annual IEEE Symposium on Logic in Computer Science, LICS 2011, June 21-24, 2011, Toronto, Ontario, Canada.* 2011, pp. 13–22. DOI: 10.1109/LICS.2011.28. URL: https://doi.org/10.1109/LICS.2011.28.

[CHS14] Arnaud Carayol, Axel Haddad, and Olivier Serre. “Randomization in Automata on Infinite Trees”. In: *ACM Trans. Comput. Log.* 15.3 (2014), 24:1–24:33. DOI: 10.1145/2629336. URL: http://doi.acm.org/10.1145/2629336.

[Chu62] Alonzo Church. “Logic, Arithmetic, Automata”. In: *Proc. International Mathematical Congress.* 1962, pp. 23–35.

[CJH03] Krishnendu Chatterjee, Marcin Jurdzinski, and Thomas A. Henzinger. “Simple Stochastic Parity Games”. In: *Computer Science Logic, 17th International Workshop, CSL 2003, 12th Annual Conference of the EACSL, and 8th Kurt Gödel Colloquium, KGC 2003, Vienna, Austria, August 25-30, 2003, Proceedings.* 2003, pp. 100–113. DOI: 10.1007/978-3-540-45220-1_11. URL: https://doi.org/10.1007/978-3-540-45220-1_11.

[CJH04] Krishnendu Chatterjee, Marcin Jurdzinski, and Thomas A. Henzinger. “Quantitative stochastic parity games”. In: *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2004, New Orleans, Louisiana, USA, January 11-14, 2004.* 2004, pp. 121–130. URL: http://dl.acm.org/citation.cfm?id=982792.982808.

[CKL94] Anthony R. Cassandra, Leslie Pack Kaelbling, and Michael L. Littman. “Acting Optimally in Partially Observable Stochastic Domains”. In: *Proceedings of the 12th National Conference on Artificial Intelligence, Seattle, WA, USA, July 31 - August 4, 1994, Volume 2.* 1994, pp. 1023–1028. URL: http://www.aaai.org/Library/AAAI/1994/aaai94-157.php.

[CLS18] Arnaud Carayol, Christof Löding, and Olivier Serre. “Pure Strategies in Imperfect Information Stochastic Games”. In: *Fundam. Inform.* 160.4 (2018), pp. 361–384. DOI: 10.3233/FI-2018-1687. URL: https://doi.org/10.3233/FI-2018-1687.

[CY90] Costas Courcoubetis and Mihalis Yannakakis. “Markov Decision Processes and Regular Events (Extended Abstract)”. In: *Automata, Languages and Programming, 17th International Colloquium, ICALP90, Warwick University, England, UK, July 16-20, 1990, Proceedings.* 1990, pp. 336–349. DOI: 10.1007/BFb0032043. URL: https://doi.org/10.1007/BFb0032043.

[CY95] Costas Courcoubetis and Mihalis Yannakakis. “The Complexity of Probabilistic Verification”. In: *J. ACM* 42.4 (1995), pp. 857–907. DOI: 10.1145/210332.210339. URL: http://doi.acm.org/10.1145/210332.210339.

[ER59] P Erdos and Alfréd Rényi. “On Cantor’s series with convergent $\sum 1/q_n$”. In: *Ann. Univ. Sci. Budapest. Eötvös. Sect. Math* 2 (1959), pp. 93–109.

[Grö08] Marcus Thomas Größer. “Reduction methods for probabilistic model checking”. PhD thesis. Dresden University of Technology, 2008. URL: http://d-nb.info/991169859.

[GTW02] Erich Grädel, Wolfgang Thomas, and Thomas Wilke, eds. *Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001]*. Vol. 2500. Lecture Notes in Computer Science. Springer, 2002. ISBN: 3-540-00388-6. DOI: 10.1007/3-540-36387-4. URL: https://doi.org/10.1007/3-540-36387-4.

[Kle06] Achim Klenke. *Wahrscheinlichkeitstheorie*. Springer, 2006. ISBN: 978-3-642-36017-6.

[KV14] Dileep Kini and Mahesh Viswanathan. “Probabilistic Automata for Safety LTL Specifications”. In: *Verification, Model Checking, and Abstract Interpretation - 15th International Conference, VMCAI 2014, San Diego, CA, USA, January 19-21, 2014, Proceedings*. 2014, pp. 118–136. DOI: 10.1007/978-3-642-54013-4_7. URL: https://doi.org/10.1007/978-3-642-54013-4%5C_7.

[KV99] Orna Kupferman and Moshe Y. Vardi. “Church’s problem revisited”. In: *Bulletin of Symbolic Logic* 5.2 (1999), pp. 245–263. URL: http://www.math.ucla.edu/%5C%7Easl/bsl/0502/0502-004.ps.

[Löd12] Christof Löding. “Automata on Infinite Trees”. (unpublished) Survey on tree automata. 2012.

[Löd99] Christof Löding. “Optimal Bounds for Transformations of omega-Automata”. In: *Foundations of Software Technology and Theoretical Computer Science, 19th Conference, Chennai, India, December 13-15, 1999, Proceedings*. 1999, pp. 97–109. DOI: 10.1007/3-540-46691-6_8. URL: https://doi.org/10.1007/3-540-46691-6_8.

[Mar75] Donald A. Martin. “Borel Determinacy”. In: *Annals of Mathematics* 102.2 (1975), pp. 363–371. ISSN: 0003486X. URL: http://www.jstor.org/stable/1971035.

[MS87] David E. Muller and Paul E. Schupp. “Alternating Automata on Infinite Trees”. In: *Theor. Comput. Sci.* 54 (1987), pp. 267–276. DOI: 10.1016/0304-3975(87)90133-2. URL: https://doi.org/10.1016/0304-3975(87)90133-2.

[MS95] David E. Muller and Paul E. Schupp. “Simulating Alternating Tree Automata by Nondeterministic Automata: New Results and New Proofs of the Theorems of Rabin, McNaughton and Safra”. In: *Theor. Comput. Sci.* 141.1&2 (1995), pp. 69–107. DOI: 10.1016/0304-3975(94)00214-4. URL: https://doi.org/10.1016/0304-3975(94)00214-4.

[NPZ17] Gethin Norman, David Parker, and Xueyi Zou. “Verification and control of partially observable probabilistic systems”. In: *Real-Time Systems* 53.3 (2017), pp. 354–402. DOI: 10.1007/s11241-017-9269-4. URL: https://doi.org/10.1007/s11241-017-9269-4.

[Pit07] Nir Piterman. “From Nondeterministic Büchi and Streett Automata to Deterministic Parity Automata”. In: *Logical Methods in Computer Science* 3.3 (2007). DOI: 10.2168/LMCS-3(3:5)2007. URL: https://doi.org/10.2168/LMCS-3(3:5)2007.

[Rab68] Michael O. Rabin. “Decidability of second-order theories and automata on infinite trees”. In: *Bull. Amer. Math. Soc.* 74.5 (Sept. 1968), pp. 1025–1029. URL: https://projecteuclid.org:443/euclid.bams/1183529958.

[Rab70] Michael O. Rabin. “Weakly Definable Relations and Special Automata”. In: *Mathematical Logic and Foundations of Set Theory*. Ed. by Yehoshua Bar-Hillel. Vol. 59. Studies in Logic and the Foundations of Mathematics. Elsevier, 1970, pp. 1–23.

[Rab72] Michael Oser Rabin. *Automata on Infinite Objects and Church’s Problem*. Boston, MA, USA: American Mathematical Society, 1972. ISBN: 0821816632.

[Ras+07] Jean-François Raskin et al. “Algorithms for Omega-Regular Games with Imperfect Information”. In: *Logical Methods in Computer Science* 3.3 (2007). DOI: 10.2168/LMCS-3(3:4)2007. URL: https://doi.org/10.2168/LMCS-3(3:4)2007.

[Sch06] Sven Schewe. “Synthesis for Probabilistic Environments”. In: *Automated Technology for Verification and Analysis, 4th International Symposium, ATVA 2006, Beijing, China, October 23-26, 2006.* 2006, pp. 245–259. DOI: 10.1007/11901914_20. URL: https://doi.org/10.1007/11901914_20.

[Tar72] R. Tarjan. “Depth-First Search and Linear Graph Algorithms”. In: *SIAM Journal on Computing* 1.2 (1972), pp. 146–160. DOI: 10.1137/0201010. URL: https://doi.org/10.1137/0201010.

[Tho97] Wolfgang Thomas. “Languages, Automata, and Logic”. In: *Handbook of Formal Languages: Volume 3 Beyond Words*. Ed. by Grzegorz Rozenberg and Arto Salomaa. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 389–455. ISBN: 978-3-642-59126-6. DOI: 10.1007/978-3-642-59126-6_7. URL: https://doi.org/10.1007/978-3-642-59126-6_7.

[Wei12] Thomas Weidner. “Probabilistic Automata and Probabilistic Logic”. In: *Mathematical Foundations of Computer Science 2012 - 37th International Symposium, MFCS 2012, Bratislava, Slovakia, August 27-31, 2012. Proceedings*. 2012, pp. 813–824. DOI: `10.1007/978-3-642-32589-2\_70`. URL: `https://doi.org/10.1007/978-3-642-32589-2%5C_70`.

[Zie98] Wieslaw Zielonka. “Infinite Games on Finitely Coloured Graphs with Applications to Automata on Infinite Trees”. In: *Theor. Comput. Sci.* 200.1-2 (1998), pp. 135–183. DOI: `10.1016/S0304-3975(98)00009-7`. URL: `https://doi.org/10.1016/S0304-3975(98)00009-7`.

A. Proof of Theorem 36

In the following we consider POSGs in more detail. Initially, we modularise some notions by de-coupling an arena and the objective and initial state of the game. Also we generalize our definition by allowing EVE and ADAM to be differently informed (by $\sim_E$ and $\sim_A$ respectively): We consider an arena

$$G = \left(S, E, A, (\tau_{e,a})_{e \in E, a \in A}, \sim_E, \sim_A\right).$$

A POSG $\mathcal{G}$ forms by fixing an initial state $s_0 \in S$ and an objective Acc and setting $\mathcal{G} = (G, s_0, \mathrm{Acc})$. Additionally, we introduce the notion of games with multiple initial locations as $\mathcal{G} = (G, B, \mathrm{Acc})$ for $B \subseteq S$ such that in an initial step ADAM may choose any $b \in B$ and subsequently EVE and ADAM compete in the game $\mathcal{G}_b = (G, b, \mathrm{Acc})$. Naturally, EVE has an almost-surely (positively) winning strategy in $\mathcal{G}_B$ if and only if she has an almost-surely (positively) winning strategy in $\mathcal{G}_b$. In the following we also consider conditions which are evaluated within finitely many steps (in contrast to Muller-, Rabin-, Parity- or Büchi-conditions), namely

Reachability-condition: for a set $R \subseteq S$ define

$$\mathrm{Acc}_{\text{Reachability}}(R) = (S \setminus R)^* \, RS^\omega,$$

Safety-condition: for a set $Z \subseteq S$ we define

$$\mathrm{Acc}_{\text{Safety}}(Z) = Z^\omega.$$

A.1. Finite Steps to Reachability

We introduce an observation about winning games with associated Reachability-condition:

Proposition 13. *Given $B \subseteq S$ and $R \subseteq S$ such that* EVE *has a positively winning strategy f in the Reachability-game $\mathcal{G} = (G, B, R)$. Then, there is an $N \in \mathbb{N}$ and some probability $\epsilon_B > 0$ that bounds from below the probability of the event that a state in R is visited in $\mathcal{G}_B$ if* EVE *plays f.*

Proof. We fix the notion $p_N^{g_N,s}$ as the probability of reaching R in less than N steps if ADAM initially chose s and plays according to g_N. From this we derive $p_N^{g_N} = \min_{s \in B} \{p_N^{g_N,s}\}$ and claim that there is one $N_0 > 0$ that renders $p_N^{g_N} > 0$ for all possible g_N. For the sake of contradiction we assume this is not the case, i.e. for all N there is one strategy g_N rendering $p_N^{g_N} = 0$. Since there are infinitely many considered N but only finitely many $s \in B$ infinitely often $p_N^{g_N}$ becomes 0 from starting in one particular $b \in B$. Thus, we fix such a b and obtain that for more and more increasing values for N there is always a strategy g_N to obtain $p_N^{g_N,b} = 0$. Moreover, we can claim this for every N (since the Reachability-condition can never be undone once it is achieved and there is always a higher $N' > N$ for which Adam can avoid R which also holds in turn for N).

Since there are only finitely possible decisions in A for the game $\mathcal{G}_b$ but we consider infinitely many strategies (for all $N > 0$ there is g_N)) infinitely many must agree what to do in the initial situation; say all these strategies make decision $c \in A$. For any possible subsequent game situation, i.e. the game moved from b to one $s_1, \dots, s_n \in S$ with $\tau_{f(\epsilon),c}(b, s_i) > 0$ for all $1 \leq i \leq n$, we know that $s_1, \dots, s_n \notin R$ since $p_N^{g_N,b} = 0$ for all these g_N. For every $1 \leq i \leq n$ we consider again infinitely many strategies (all g_N with $N > 1$ which initially chose c; by choice of c this is indeed an infinite collection of strategies). Since there are again only finitely many possible choices we obtain by the same argument one choice c' on which infinitely many g_N agree on. This argument can be iterated countably many times and we may construct with the choices the strategies agree upon one particular strategy, say g, which avoids R all together. But for this one g we can argue that $p_{N,b}^g = 0$ for all $N > 0$. Naturally, playing g with an additional initial choice of b renders g winning against f in the positive Reachability-game $\mathcal{G}_B$ contradicting the prerequisite that f is positively winning.

Hence, we can assured the existence of $N_0 > 0$ which bounds the number rounds any counter strategy may avoid R in $\mathcal{G}_B$. Since there are only finitely many choices in a game up to N_0 rounds starting in any $s \in B$ we may fix these choices as (partial) strategies of Adam in a set G_{s,N_0}. Setting $\epsilon_B = \min \left\{ p_{N_0}^g : g \in G_{s,N_0} \right\}$ yields the desired result. □

A.2. Perfectly Informed Adam

We consider a restricted Reachability-condition in n-steps, i.e.

$$\mathrm{Acc}^n_{\mathrm{Reachability}}(R) = \bigcup_{0 \leq i \leq n} S^i R S^\omega.$$

We show

Proposition 14. *For a $K \subseteq S$ such that $K \subseteq [s]_{\sim_E}$ for one $s \in S$ and $n > 0$* Eve *positively wins $\left(G, K, \mathrm{Acc}^n_{\mathrm{Reachability}}(R)\right)$ if and only if there exists an action $e \in E$ and $K' \subseteq S$ such that*

- Eve *positively wins $\left(G, K', \mathrm{Acc}^{n-1}_{\mathrm{Reachability}}(R)\right)$,*
- *for every $s \in K \setminus R$ and $a \in A$ there is $s' \in K'$ such that $\tau_{a,e}(s, s') > 0$.*

Proof. For the implication consider EVE's positive winning strategy f in

$$\left(G, K, \mathrm{Acc}^{n}_{\mathrm{Reachability}}(R)\right).$$

Regardless of the initial choice $b \in K \setminus R$ of ADAM, EVE consistently chooses the same $e = f([b]_{\sim_E})$ since $[b]_{\sim_E} = [b]_{\sim_E}$ for all $b, b' \in K$. Naturally, since f is positively winning then there is for every $a \in A$ one $s^a_b \in \mathrm{support}(\tau_{e,a}(b, \cdot))$ such that EVE positively wins $\left(G, s^a_b, \mathrm{Acc}^{n-1}_{\mathrm{Reachability}}(R)\right)$ (otherwise ADAM may win by choosing b, playing a and then the winning riposte in $\left(G, s^a_b, \mathrm{Acc}^{n-1}_{\mathrm{Reachability}}(R)\right)$). We gather

$$K' = \bigcup_{a \in A, b \in K} s^a_b$$

and obtain the claimed properties for K' and e as above.

Consider the properties on K' and $e \in E$ satisfied. Hence, for every initial choice $b \in K$ and $a \in A$ there is a positive probability to move to K'. Playing the corresponding winning strategy from $\left(G, K', \mathrm{Acc}^{n-1}_{\mathrm{Reachability}}(R)\right)$ after the initial decision e yields a positively winning strategy in the original game. □

We use this proposition to prove the following

Theorem 41. *For a given arena G EVE construct*

$$\mathcal{W}_0 = \mathrm{Pot}(R)$$

and

$$\begin{aligned}\mathcal{W}_{i+1} = \{K \in \mathrm{Pot}(S) \mid & \textit{for all } k \in K \textit{ exists } e \in E, K' \in \mathcal{W}_i \textit{ s.t.} \\ & \textit{for all } s \in \left([k]_{\sim_E} \setminus R\right), a \in A \textit{ exists } t \in K' \textit{ with } \tau_{e,a}(s,t) > 0\},\end{aligned}$$

and their limit $\mathcal{W}$. For every non-empty $K \in \mathrm{Pot}(S)$ holds that EVE *has a positively winning strategy f in the Reachability-game (G, K, R) if and only if $K \in \mathcal{W}$. Moreover, it can be decided in time exponential in $|S|$ if such an f exists and if so it can be constructed using* $\mathrm{Pot}(S)$ *states and reaching R with a positive probability in* $|\mathrm{Pot}(S)|$ *moves.*

Proof. Using Proposition 14 we obtain by induction to n that every non-empty $K \in \mathrm{Pot}(S)$ is part of $\mathcal{W}_n$ if and only if EVE positively wins $\left(G, K, \mathrm{Acc}^{n}_{\mathrm{Reachability}}(R)\right)$.

Conversely, assuming $K \in \mathcal{W}$ we construct a strategy for EVE as follows: EVE stores K and firstly, identifies the level n_K where K became part of $\mathcal{W}$ and secondly, for every equivalence class in K the set K' which witnessed that K belongs to $\mathcal{W}_{n_K}$. Notably, for all these K' holds $n_{K'} < n_K$. EVE plays the associated action $e_{K'}$ and updates the memory to K'. This strategy enforces by definition of K and K' a positive probability to move to one of these K' (depending on the initially observed equivalence class). Repeating this argument eventually leads to sets $T \in \mathcal{W}_0$ since we decreasingly move through the stages of $\mathcal{W}$. Therefore, the corresponding strategy enforces a positive probability to end up in R from where EVE may play on arbitrarily. □

A.3. Automaton-compatible strategies

We introduce the notion of *automaton-compatible* strategies. We define

$$\mathcal{T} = \left(Q, \Sigma_E \times [S]_{\sim_E}, q_0, q_s, \delta, \lambda\right)$$

where $\Sigma_E \times [S]_{\sim_E}$ is the input alphabet, $q_0, q_s \in Q$ a start and a sink state respectively, $\delta : Q \times \Sigma_E \times [S]_{\sim_E} \to Q$ a deterministic transition function and $\lambda : Q \to \mathrm{Pot}(\Sigma_E)$ a labelling of the states of $\mathcal{T}$. Additionally, we enforce

- $\lambda(q) = \emptyset$ if and only if $q = q_s$,
- $\delta(q, (\sigma, x)) = q_s$ if and only if $\sigma \notin \delta(q)$ for all $q \in Q$, $\sigma \in \Sigma_E$ and $x \in [S]_{\sim_E}$.

This automaton $\mathcal{T}$ associates with any play $u \in S^*$ in $\mathcal{G}$ precisely one $q \in Q$ and moreover, a labelling $\lambda(q)$. Any strategy f of Eve is considered compatible with $\mathcal{T}$ if for any play $u \in S^*$ Eve plays an action from the associated labelling. One important automaton is associated with Eve's knowledge of the current situation of the game, i.e. the states she consideres possible given her observations. In general, this knowledge expands beyond the observed equivalence class since the history might render certain states within the current equivalence class impossible to be the current state. This induces one special automaton structure $\mathcal{K}$: Knowledge is encoded as sets of states Eve considers possible. Fix an initial knowledge $K_0 \subseteq S$ such that $K_0 \subseteq [s]_{\sim_E}$ for one $s \in S$. Inductively, we construct for any sequence $u \in [S]^*_{\sim_E}$ the current knowledge of Eve $\mathrm{currentKnowledge}_E^{K_0}$. Initially holds $\mathrm{currentKnowledge}_E^{K_0}([s_0]_{\sim_E}) = K_0$ and at every step the knowledge is updated by $\mathrm{updateKnowledge}_E$ which computes all possible results considering the available information:

$$\begin{aligned}&\mathrm{updateKnowledge}_E(K, e, [s]_{\sim_E})\\ &\qquad = \left\{t \in [s]_{\sim_E} \mid \text{there is } o \in K \text{ and } a \in A \text{ s. t. } \tau_{e,a}(o,t) > 0\right\}\end{aligned}$$

for the current knowledge K, the chosen action $e \in E$ of Eve and the following observation $[s]_{\sim_E}$. Hence, we obtain inductively

$$\mathrm{currentKnowledge}(u \cdot s) = \mathrm{updateKnowledge}(\mathrm{currentKnowledge}(u), e, [s]_{\sim_E})$$

where $e = f(u)$ is the choice of Eve's strategy f. We define an associated (unlabelled) knowledge-automaton for Eve for a fixed initial knowledge K_0 and loosing knowledge K_s

$$\mathcal{T}_{\mathcal{K}} = \left(\mathrm{Pot}(S), E \times [S]_{\sim_E}, K_0, K_s, \mathrm{updateKnowledge}\right).$$

A.3.1. Automaton-Game Product

For a given automaton

$$\mathcal{T} = \left(Q, \Sigma_E \times [S]_{\sim_E}, q_0, q_s, \delta, \lambda\right)$$

and arena

$$G = \left(S, E, A, (\tau_{e,a})_{e \in E, a \in A}, \sim_E, \sim_A\right)$$

we define a product

Definition 36.
Given G and $\mathcal{T}$ as above, we define

$$G \otimes \mathcal{T} = \left(S \times Q, E, A, \left(\tau'_{e,a}\right)_{e \in E, a \in A}, \sim'_E, \sim'_A\right)$$

with

$$\tau'_{e,a}((s,q),(z,p)) = \begin{cases} \tau_{e,a}(s,z) & \text{if } p = \delta(q,(e,[z]_{\sim_E})), \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\begin{array}{lll} (s,q) \sim'_E (z,p) & \text{if and only if} & s \sim_A z \text{ and } q = p, \\ (s,q) \sim'_A (z,p) & \text{if and only if} & s = z, q = p. \end{array}$$

Note that $\sim'_A$ is the equality relation and therefore Adam is not restricted in his observations whatsoever. We say Adam is *perfectly informed.*

We observe the following result:

Lemma 32. Eve *has a $\mathcal{T}$-compatible strategy in the Safety-game (G, B, Z) if and only if she has a surely winning strategy in the Safety-game $(G \otimes \mathcal{T}, B \times \{q_0\}, S \times (Q \setminus \{q_s\}))$.*

Proof. By the definition of $\mathcal{T}$ any action that is not compatible with λ yields q_s as a result. Moreover, $\mathcal{T}$ is a deterministic automaton, hence any strategy of Eve in (G, B, Z) that is $\mathcal{T}$-compatible can be used equally in $(G \otimes \mathcal{T}, B \times \{q_0\}, S \times (Q \setminus \{q_s\}))$ since by compatibility with $\mathcal{T}$ a movement to q_s is not possible. On the other hand, we may translate any strategy in $(G \otimes \mathcal{T}, B \times \{q_0\}, S \times (Q \setminus \{q_s\}))$ to (G, B, Z). Since $S \times \{q_s\}$ is avoided this strategy clearly is $\mathcal{T}$-compatible. $\square$

This allows us to deduce for a fixed $\mathcal{T}$

Theorem 42. *When* Adam *is perfectly informed, one can decide in time polynomial in $2^{|S|}$ and polynomial in Q whether* Eve *has a $\mathcal{T}$-compatible strategy that is positively winning in the Reachability-game (G, B, R). If such a strategy exists, one can construct one that uses memory of size polynomial in $|Q|$ and exponential in $|S|$.*

Proof. We consider the knowledges (in the sense of $\mathcal{K}$) of Eve in the arena $G \otimes \mathcal{T}$. We observe that the associated observation-relation $\sim'_E$ in $G \otimes \mathcal{T}$ makes the current state transparent to Eve. Therefore, it is sufficient to capture the set of states of G that Eve considers possible and the unique current state of $\mathcal{T}$. Hence, knowledges can be encoded as elements in $\mathrm{Pot}(S) \times Q$. We fix the largest subset $\mathbb{K}$ of these knowledges of Eve and an associated mapping $\lambda : \mathbb{K} \to \mathrm{Pot}(E)$ such that:

- for all $K \in \mathbb{K}$ holds that $K \cap S \times \{q_s\} = \emptyset$, i.e. Eve avoids states beyond her Safety-region in $G \otimes \mathcal{T}$ and

- for all $K \in \mathbb{K}$ it holds that

$$\text{updateKnowledge}_E(K, e, [s]_{\sim_E}) \in \mathbb{K} \setminus \{\emptyset\}$$

for every $e \in \lambda(K)$, i.e. actions of Eve may not move her outwards these "safe" knowledges.

Merging all elements of $(\text{Pot}(S) \times Q) \setminus \mathbb{K}$ in K_s allows to obtain the associated automaton $\mathcal{T}_{\mathbb{K}}$ with labelling λ. Considering the knowledge structures is sufficient for Reachability-conditions (cp. [Ras+07]). Again, we embed $\mathcal{T}_{\mathbb{K}}$ into the arena and obtain

Lemma 33. Eve *has a* $\mathcal{T}$ *compatible positively winning strategy in the Reachability-game* $\mathcal{G} = (G, B, R)$ *if and only if she has a positively winning strategy in the Reachability-game* $\mathcal{G}' = (G \otimes \mathcal{T}_{\mathbb{K}}, B \times \{K_0\}, R \times (\mathbb{K} \setminus K_0))$.

This positively winning strategy in $\mathcal{G}'$ *can be translated to a positively winning strategy in* (G, B, R) *with memory of size* $\mathcal{O}(N \cdot |\text{Pot}(S)| \cdot |Q|)$ *where* N *is the memory size of the positively winning strategy in* $\mathcal{G}'$.

Proof. Positively winning in $\mathcal{G}'$ can be restricted to those that respect the actions given in λ regarding the $\mathbb{K}$-component of the current position. Otherwise this component becomes K_s which traps the play and does not allow to play for the Reachability target anymore, but by definition of $\mathbb{K}$ and λ even loosing plays may respect λ. By Lemma 32 we obtain the $\mathcal{T}$-compability of the strategy. Simulating $\mathcal{T}_{\mathbb{K}}$ allows to translate this strategy to the original game. Conversely, any positively winning $\mathcal{T}$-compatible strategy in (G, B, R) induce a corresponding strategy in $\mathcal{G}'$. □

We may use Theorem 41 to obtain a strategy for Eve in the game $\mathcal{G}'$ from Lemma 33 since Adam is perfectly informed there to obtain the desired result. □

A.4. Better Informed Adam

Consider an automaton $\mathcal{T}$ as before and an arena G as before but we additionally consider $\sim_A \subseteq \sim_E$, i.e. if Adam considers two states equal so does Eve but not necessarily the other way around. Adam is considered *better informed* than Eve in such a game. Naturally, this case includes equally informed players, i.e. $\sim_A = \sim_E$. Assuming Adam is better informed that Eve allows to argue that knowledge of Adam is always at least as good as knowledge of Eve. This leads to

Theorem 43. *When* Adam *is more informed than* Eve, *one can decide in time polynomial in* $2^{2^{|S|}}$ *and polynomialn in* $|Q|$ *whether* Eve *has a* $\mathcal{T}$*-compatible strategy that is positively winning in the reachability game* (G, B, R). *If such a strategy exists, we can construct one with memory polynomial in* $|Q|$ *and doubly exponential in* $|S|$.

Proof. We fix

$$\mathcal{H} = \left\{ A \in \operatorname{Pot} S \mid \text{there exists } s \in S \text{ s.t. } A \subseteq [s]_{\sim_A} \right\}$$

and an associated operator $h : \mathcal{H} \times E \times A \to \operatorname{Pot}(\mathcal{H})$ with

$$h(H, a, e) = \left\{ H' \cap [s]_{\sim_A} : \text{for all } H' = \bigcup_{s \in H} \operatorname{support}(\tau_{e,a}(s, \cdot)) \text{ and } s \in S \right\}.$$

From which we define a new arena $G' = \left(\mathcal{H}, E, A, \left(\tau'_{e,a}\right)_{e \in E, a \in A}, \sim'_E, \sim'_A \right)$ with

$$\tau'_{e,a}(H, H') = \begin{cases} \frac{1}{|h(H,e,a)|} & \text{if } H' \in h(H, e, a), \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\begin{array}{llll} H_1 \sim'_E H_2 & \text{if and only if} & s_1 \sim_E s_2 & \text{for all } s_1 \in H_1, s_2 \in H_2, \\ H_1 \sim'_A H_2 & \text{if and only if} & H_1 = H_2. & \end{array}$$

Note that Adam is perfectly informed and the equivalence classes of $\sim'_E$ can be identified with equivalence classes of $\sim_E$ since Eve is less informed than Adam and therefore all elements in any $H \in \mathcal{H}$ are indistinguishable for Eve. We define $\nu(K) = \{\{s\} : s \in K\}$ and obtain

Proposition 15. Eve *has a positively winning $\mathcal{T}$-compatible strategy in the Reachability-game $\mathcal{G} = (G, B, R)$ if and only if she has a positively winning $\mathcal{T}$-compatible strategy in the Reachability-game $\mathcal{G}' = (G', \nu(B), R')$ with $R' = \{H \in \mathcal{H} \mid H \cap R \neq \emptyset\}$.*

Proof. First, by the observation that we idenfity $H \in \mathcal{H}$ for Eve as their associated equivalence classes we conclude that $\mathcal{T}$-compatibility translates for strategy in both games.

Consider any positively winning $\mathcal{T}$-compatible strategy f in $\mathcal{G}$. By Proposition 13 we obtain a bound N and a positive probability ϵ_N such that Eve visits R in N steps with probability at least ϵ_N. We observe that given f Adam is informed of Eve's actions and may compute very at every moment $h(H, e, a)$. Moreover, given any strategy of Adam for the first N steps in $\mathcal{G}'$ computes precisely the states reachable with positive probability in $\mathcal{G}$ given Eve translates f and Adam his strategy. Naturally, one of these states contains a state in R' by the choice of N.

On the other hand, assume f is $\mathcal{T}$-compatible and positively winning in $\mathcal{G}'$. Assuming that there is g in $\mathcal{G}$ to avoid R indefinitely, This strategy may be equally be played in $\mathcal{G}'$. Here it must not cause a situation where R' is avoided indefinitely since f is assumed to be winning. Nevertheless, the elements of the states in $\mathcal{G}'$ that are reached with positive probability coincide with those states in $\mathcal{G}$ that are reached with positive probability. This contradicts either f positively winning in $\mathcal{G}'$ or the existence of g in $\mathcal{G}$. □

Applying Theorem 42 to this $\mathcal{G}'$ gives the desired results. □

A.5. Almost-Surely Winning for Büchi-Conditions

We move from Reachability-conditions to Büchi-conditions, hence we fix an arena

$$G = \left(S, E, A, (\tau_{e,a})_{e\in E, a\in A}, \sim_E, \sim_A\right)$$

and an associated Büchi-set $F \subseteq S$. For this game we define *almost-surely winning knowledges* for EVE $\mathcal{K}^\top$, which is a collection of subsets $K \subseteq S$ such that every $K \subseteq [s]_\sim$ for one $s \in S$ and EVE almost-surely wins the Büchi-game (G, K, F). We characterize $\mathcal{K}^\top$ as a fixpoint of the operator $\Xi : \mathrm{Pot}(\mathrm{Pot}(S)) \to \mathrm{Pot}(\mathrm{Pot}(S))$ where for one $\mathcal{K} \subseteq \mathrm{Pot}(S)$ and $K \in \mathcal{K}$ we consider $K \in \Xi(\mathcal{K})$ if EVE has a positively winning strategy in the *Reachability*-game (A, K, F) such that her knowledge does never leave $\mathcal{K}$. We claim $\mathcal{K}^\top$ is the greatest fixpoint of Ξ and substantiate this claim with

Lemma 34. *$\mathcal{K}^\top$ indeed is the greatest fixpoint of Ξ.*

Proof. For any set of knowledges $\mathcal{K} \subseteq \mathrm{Pot}(S)$ we say some $K \in \mathcal{K}$ is $\mathcal{K}$-*good* if EVE has a positively winning strategy in the Reachability-game (G, K, F) which constantly produces knowledges of EVE which are in $\mathcal{K}$. We proceed in showing that any $K \in \mathcal{K}^\top$ is $\mathcal{K}^\top$-good implying that $\mathcal{K}^\top$ is a fixpoint of Ξ.

Lemma 35. *Fix any $K \in \mathcal{K}^\top$ and the associated almost-surely winning strategy f_K of* EVE *in the Büchi-game (G, K, F) with $e = f_K([k]_{\sim_E})$ for any $k \in K$ (by definition of $\mathcal{K}^\top$ all $k \in K$ are of the same equivalence class). Then, for every $a \in A$ and all $z \in \cup_{o\in K} \mathrm{support}(\tau_{e,a}(o, \cdot))$ holds*

$$\mathrm{updateKnowledge}(K, e, [z]_{\sim_E}) \in \mathcal{K}^\top.$$

Proof. Consider one $a \in A$ and $z \in \cup_{o\in K} \mathrm{support}(\tau_{e,a}(o, \cdot))$ as above. Fix $K' = \mathrm{updateKnowledge}(K, e, [z]_{\sim_E})$. Notably, there is a strategy for ADAM to force the play into z with positive probability. If EVE did not have a almost-surely winning strategy in the Büchi-game (G, z, F) with initial knowledge K' she would not have an almost-surely winning strategy in (G, K, F) to begin with. Hence $K' \in \mathcal{K}^\top$ and inductively we get that K is $\mathcal{K}^\top$-good since there are only finitely many steps until a state in F is visited (by Proposition 13) and all these steps ensure to stay in $\mathcal{K}^\top$ by the argument above. □

Hence, we established $\mathcal{K}^\top$ as fixpoint of Ξ only missing that it is the greatest fixpoint. Fix any other $\mathcal{K}$ with $\Xi(\mathcal{K}) = \mathcal{K}$ and by definition of Ξ every $K \in \mathcal{K}$ is $\mathcal{K}$-good and by Proposition 13 comes with associated N_K, ϵ_K. Set $N = \max\{N_K : K \in \mathcal{K}\}$ and $\epsilon = \min\{\epsilon_K : K \in \mathcal{K}\}$. Separating the game into sequences of length N where EVE considers her initial knowledge H and plays for N steps with strategy f_H, then she re-considers her initial knowledge H' and plays for N steps with strategy $f_{H'}$ and so on. This gives a strategy f and we claim f is almost-surely winning in the Büchi-game (G, K, F). By definition of f there is a probability of at least ϵ to visit a state in F all N steps. That states in F are almost-surely visited infinitely often is a direct consequence of Theorem 2. This entails that $K \in \mathcal{K}^\top$ concluding the proof. □

By obvious monotonicity of Ξ and $\mathcal{K}^\top \subseteq \mathrm{Pot}(S)$ we reach a fixpoint after at most $2^{|S|}$ applications of Ξ. Expressing that Eve's knowledge consistently agrees with some $\mathcal{K}$ can be done by defining an appropiate automaton $\mathcal{T}_\mathcal{K}$ with states exponential in S. Theorem 36 follows by using Theorem 43 to compute all steps of Ξ.