# Stochastic timed automata

Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, Quentin Menet,
Christel Baier, Marcus Groesser, Marcin Jurdzinski

Logical Methods in Computer Science
Vol. 10(4:6)2014, pp. 1–73
www.lmcs-online.org

Submitted Mar. 28, 2013
Published Dec. 9, 2014

# STOCHASTIC TIMED AUTOMATA

NATHALIE BERTRAND [a], PATRICIA BOUYER [b], THOMAS BRIHAYE [c], QUENTIN MENET [d], CHRISTEL BAIER [e], MARCUS GRÖSSER [f], AND MARCIN JURDZIŃSKI [g]

[a] Inria Rennes, France
*e-mail address*: nathalie.bertrand@inria.fr

[b] LSV, CNRS & ENS Cachan, France
*e-mail address*: bouyer@lsv.ens-cachan.fr

[c,d] Université de Mons, Belgium
*e-mail address*: {thomas.brihaye,quentin.menet}@umons.ac.be

[e,f] TU Dresden, Germany
*e-mail address*: {baier,groesser}@tcs.inf.tu-dresden.de

[g] University of Warwick, UK
*e-mail address*: mju@dcs.warwick.ac.uk

Abstract. A stochastic timed automaton is a purely stochastic process defined on a timed automaton, in which both delays and discrete choices are made randomly. We study the almost-sure model-checking problem for this model, that is, given a stochastic timed automaton $\mathcal{A}$ and a property $\varphi$, we want to decide whether $\mathcal{A}$ satisfies $\varphi$ with probability 1. In this paper, we identify several classes of automata and of properties for which this can be decided. The proof relies on the construction of a finite abstraction, called the thick graph, that we interpret as a finite Markov chain, and for which we can decide the almost-sure model-checking problem. Correctness of the abstraction holds when automata are almost-surely fair, which we show, is the case for two large classes of systems, single-clock automata and so-called weak-reactive automata. Techniques employed in this article gather tools from real-time verification and probabilistic verification, as well as topological games played on timed automata.

*2012 ACM CCS:* [**Theory of computation**]: Semantics and reasoning—Program reasoning—Program verification; [**Software and its engineering**]: Software organization and properties—Software functional properties—Formal methods—Model checking.

*Key words and phrases:* Timed automata, Model checking, Probability, Topology.

LOGICAL METHODS IN COMPUTER SCIENCE
DOI:10.2168/LMCS-10(4:6)2014

## 1. Introduction

**Timed automata and their extensions.** In the last twenty years a huge effort has been made to design expressive models for representing computerised systems. As part of this effort the model of timed automata [AD90, AD94] has been proposed in the early 90's as a suitable model for representing systems with real-time constraints. Numerous works have focused on that model, and it has received an important tool support, with for instance the development of tools like Uppaal [BDL+06] or Kronos [BDM+98].

Given the success of the timed-automata-based technology for verifying real-time systems, several extensions have been proposed, with the aim of representing more faithfully real systems. They include timed games [AMPS98] for modeling control problems and priced timed automata [ALP01, BFH+01, BFLM11] for modeling various quantities in timed systems, like energy consumption.

**Stochastic extensions of timed automata.** Many applications like communication protocols require models integrating both real-time constraints and randomised aspects (see e.g. [Sto03]). The development of such models and corresponding verification algorithms is a challenging task, since it requires combining techniques from both fields of real-time verification and probabilistic verification. In the literature we distinguish two main different approaches.

A first approach consists in modeling the system as a purely stochastic process, and to express soft real-time constraints in the property that is checked. A model of choice for the system is that of continuous-time Markov chains (CTMC for short), while a rather wide spectrum of property formalisms has been considered, going from the logic CSL (continuous stochastic logic) and extensions thereof [ASSB00, BHHK03, DHS09, ZJNH11] to (deterministic) timed automata [CHKM11]. In this context several exact and approximate model-checking algorithms have been developed.

Another approach consists in integrating both features into a complex model (e.g. an extension of timed automata or Petri nets with stochastic evolution rules), and to analyse this model. This allows one to represent hard timing contraints such as deadlines. In this article we focus on automata-based models, and therefore only review related work on models based on timed automata. Such models include probabilistic timed automata [KNSS02] where discrete distributions are assigned to actions and for which the tool Prism [KNP11] has been developed. Delays or durations of events can also be made randomised. This is done for instance in [ACD91, ACD92] and later in [KNSS00], yielding either independent events and exact model-checking algorithms (for a probabilistic and timed extension of computation tree logic), or approximate model-checking algorithms.

The current work follows this last approach, and surveys and extends results based on the model of *stochastic timed automata*. This model has been proposed and studied in a series of papers [BBB+07, BBB+08, BBJM12]. The semantics of a stochastic timed automaton is a purely stochastic process based on a timed automaton, in which both delays and discrete choices are made randomly. This model has later been extended with nondeterminism and interaction [BF09, BS12], but in this article we focus on the original purely stochastic model.

**Overview of the contributions.** In this article we are interested in the almost-sure model-checking of stochastic timed automata. This problem asks, given a stochastic timed automaton $\mathcal{A}$ and a property $\varphi$, whether $\mathcal{A}$ almost-surely satisfies $\varphi$ (that is, with probability 1). Our approach to solve this problem relies on the construction of a finite Markov chain $\mathsf{MC}(\mathcal{A})$[1] on which we will check whether $\varphi$ almost-surely holds or not. We will then say that the abstraction $\mathsf{MC}(\mathcal{A})$ is correct w.r.t. $\varphi$ whenever $\varphi$ almost-surely holds equivalently in $\mathcal{A}$ and in $\mathsf{MC}(\mathcal{A})$. Unfortunately, this will not be the case in general, and beyond the introduction of stochastic timed automata, the main goal of this article is to identify subclasses of stochastic timed automata and subclasses of properties for which the abstraction $\mathsf{MC}(\mathcal{A})$ is correct.

More precisely, we show that $\mathsf{MC}(\mathcal{A})$ is a correct abstraction w.r.t. property $\varphi$ in the following cases: (i) if $\varphi$ is a safety property, (ii) if $\varphi$ is an $\omega$-regular (or $\mathsf{LTL}$) property and $\mathcal{A}$ is a single-clock stochastic timed automaton, and (iii) if $\varphi$ is an $\omega$-regular (or $\mathsf{LTL}$) property and $\mathcal{A}$ belongs to a subclass of stochastic timed automata called *weak reactive*. In fact, cases (ii) and (iii) are consequences of a more general result stating that if the runs in a stochastic timed automaton $\mathcal{A}$ are almost-surely fair[2], then $\mathsf{MC}(\mathcal{A})$ is a correct abstraction. The results then follow from the (highly non trivial) proof that both weak reactive and single-clock stochastic timed automata are almost-surely fair.

We also establish the exact complexity of the almost-sure model-checking problem in the three above cases. More precisely, we prove that the almost-sure model-checking problem is (i) $\mathsf{PSPACE}$-complete on stochastic timed automata against safety properties, (ii) $\mathsf{PSPACE}$-complete (resp. $\mathsf{NLOGSPACE}$-complete) on single-clock stochastic timed automata against properties given as $\mathsf{LTL}$ formulas (resp. $\omega$-regular properties), and (iii) $\mathsf{PSPACE}$-complete on weak reactive stochastic timed automata against $\omega$-regular properties. We finally extend this last result to specifications given as deterministic timed automata. Let us point out that the decidability status of the almost-sure model checking problem for $\mathsf{LTL}$ properties on the general class of stochastic timed automata is still an open problem.

**A model which relaxes timed automata assumptions.** Let us mention that one initial motivation for defining stochastic timed automata was the robustness of timed systems. Indeed, the model of timed automata is an idealised mathematical model, which makes strong assumptions on the behaviour of the represented real system: it assumes for instance infinite precision of the clocks, instantaneous events and communications, whereas a real system will have slightly different behaviours (like measure time with digital clocks). This topic of research is very rich, and many models and results have already been described.

We review some of the frameworks which have been studied in this context, but will not give a long list of references. We better point to a survey made in 2011 [Mar11], and to a recent PhD thesis [San13], which review in details the literature on the subject. Let us first mention two models of implementable controllers proposed in [DDR04] and in [SBM11], where constraints and precision of clocks are somewhat relaxed. In this framework, if the model satisfies a property, then, on a simple model of processor, its implementation will also satisfy this property. This implementation model induces a very strong notion of robustness, suitable for really critical systems (like rockets or X-by-wire systems in cars), but maybe

[1] In the core of the article, $\mathsf{MC}(\mathcal{A})$ is the so-called thick graph $\mathcal{G}_{\mathrm{t}}(\mathcal{A})$, that we interpret as a finite Markov chain, putting the uniform distributions over edges.

[2] Roughly, a run is fair if any edge which is enabled infinitely often is taken infinitely often.

too strong for less critical systems (like mobile phones or network applications). Another robustness model has been proposed at the end of the 90's in [GHJ97] with the notion of tube acceptance: a metric is put on the set of traces of the timed automaton, and a trace is robustly accepted if and only if a tube around that trace is classically accepted. This language-focused notion of acceptance is however not completely satisfactory for implementability issues, because it does not take into account the structure of the automaton.

In this context, the model of stochastic timed automata alleviates some disadvantages to the strong mathematical assumptions made in timed automata. First, randomising delays and the choice of transitions removes unlikely behaviours (like those requiring satisfaction of very precise clock constraints), and only important and meaningful sets of behaviours are then taken into account in the verification process. Then, the assumptions made in timed automata mentioned above lead to the existence of unreal(istic) behaviours of the model, such as Zeno behaviours[3] that one would like to ignore. We will then realise that, unless the underlying timed automaton is inherently Zeno, the probability of Zeno behaviours will be 0 (at least in the classes of models we have identified). This allows us to convincingly claim that stochastic timed automata can be used as a possible solution for relaxing side-effects of mathematical assumptions made in timed automata.

As a motivating example, we describe a model of the IPv4 Zeroconf protocol using stochastic timed automata, see Figure 1. This protocol aims at configuring IP addresses in a local network of appliances. When a new appliance is plugged, it selects an IP address at random, and broadcasts several probe messages to the network to know whether this address is already used or not. If it receives in a bounded delay an answer from the network informing that the IP is already used, then a new IP address is chosen. It may be the case that messages get lost, in which case there is an error. In [BvdSHV03], a simple model for the IPv4 Zeroconf protocol is given as a discrete-time Markov chain, which abstracts away timing constraints. Using stochastic timed automata, expressing the delay bound is feasible.

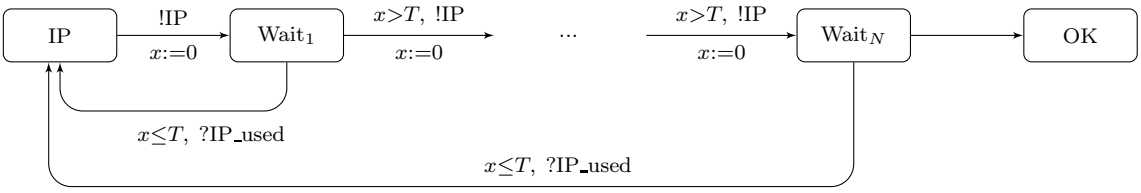



Figure 1: Modelling the IPv4 Zeroconf using stochastic timed automata.

The example of Figure 1 illustrates an important feature of stochastic timed automata. Compared to CTMC-like models, stochastic timed automata allow one to express hard timing contraints such as deadlines (constraint $x \leq T$ in this example). Another important feature of stochastic timed automata, as we will show in this article, is that the almost-sure satisfaction of properties is independent of the precise probability distributions over delays. This is a major advantage since it avoids the problem of finding realistic probability distributions which is known to be a difficult task, see *e.g.* [BDE+14].

[3]That is, time-converging behaviours.

**The Cantor topology: a useful tool.** In a former paper [VV06], Varacca and Völzer show a strong correspondence between a standard Markov-chain-based probabilistic semantics of a finite automaton, and the Cantor topology over the set of infinite executions of this automaton. They show in particular that almost-sure sets of executions (that is, sets of executions which have probability 1) coincide with topologically large sets of executions. Following this idea we also define a topological semantics *à la* Cantor for a timed automaton. In our framework, the above equivalence does not always hold, but in many cases however we will be able to prove it. This characterisation is incredibly useful in order to prove our results. The key tool in our techniques is a topological game called Banach-Mazur game [Oxt57].

**Related work.** The literature on stochastic processes is huge. We already mentioned several related works, but we would like to discuss a bit more the works [DHS09, CHKM11], which we think are the closest to the current article. In both papers the model is that of CTMCs. Timing constraints are expressed in the properties, either given as deterministic timed automata [CHKM11] or as an extension of $\mathsf{CSL}$ called $\mathsf{CSL}_{\mathsf{TA}}$ [DHS09], which extends $\mathsf{CSL}$ with properties given as single-clock deterministic timed automata.

Paper [CHKM11] is interested in quantitative model-checking, that is, given a CTMC $\mathcal{C}$ and a property given as a deterministic (Muller) timed automaton $\mathcal{A}$, the aim is to compute the probability that runs of $\mathcal{C}$ are accepted by $\mathcal{A}$. This probability is characterised using Volterra integral equations, which can be transformed into linear equations when $\mathcal{A}$ has a unique clock. Therefore quantitative verification can be done for single-clock specifications but can only be approximated in the general case. Our results are somehow incomparable since we allow for a more general model (stochastic timed automata instead of CTMCs) but prove decidability only for the qualitative model-checking problem.

Paper [DHS09] is interested in model-checking of CTMCs against properties expressed as formulas of $\mathsf{CSL}_{\mathsf{TA}}$. This logic involves probability formulas, and uses single-clock deterministic timed automata as predicates. Model-checking of the general logic can be approximated, but if formulas only have qualitative subformulas, the exact model-checking can be decided. We do not consider logics, but we allow general deterministic timed automata in our specifications.

**Organisation.** Section 2 summarises our notations for timed automata, and specifications languages (such as $\mathsf{LTL}$ and $\omega$-regular properties). Section 3 presents stochastic timed automata, the notion of almost-sure satisfaction and the almost-sure model-checking problem, while Section 4 presents the topological semantics and the notion of large satisfaction. In Section 5, we define a finite abstraction of a stochastic timed automaton, named *thick graph*, which will be essential in order to solve the almost-sure model-checking problem. In Section 6, we show that the topological and the probabilistic semantics coincide first if we restrict to safety properties and then for $\omega$-regular properties but under the restriction that the system is almost-surely fair. In Section 7, we identify two subclasses of stochastic timed automata which are almost-surely fair, namely weak reactive and single-clock. Finally, the algorithmic issues and the complexity results are given in Section 8. To improve readability of the article, technical proofs are postponed to the Appendix.

This article presents results from [BBB+07, BBB+08, BBJM12] in a uniform way, provides the complete proofs, and generalises the results from [BBJM12] to a larger class of stochastic timed automata.

## 2. Preliminaries

2.1. **The timed automaton model.** We denote by $X = \{x_1, \dots, x_k\}$ a finite set of *clocks*. A *clock valuation* over $X$ is a mapping $\nu : X \to \mathbb{R}_+$, where $\mathbb{R}_+$ denotes the set of nonnegative reals. We write $\mathbb{R}_+^X$ for the set of clock valuations over $X$, and $\mathbf{0}_X$ (or simply $\mathbf{0}$ if $X$ is clear in the context) for the valuation assigning 0 to every clock of $X$. Given a clock valuation $\nu$ and $\tau \in \mathbb{R}_+$, $\nu + \tau$ is the clock valuation defined by $(\nu + \tau)(x) = \nu(x) + \tau$ for every $x \in X$. If $Y \subseteq X$, the valuation $[Y \leftarrow 0]\nu$ is the valuation $\nu'$ such that $\nu'(x) = 0$ if $x \in Y$, and $\nu'(x) = \nu(x)$ otherwise. A *guard* over $X$ is a finite conjunction of expressions of the form $x \sim c$ where $x \in X$ is a clock, $c \in \mathbb{N}$ is an integer, and $\sim$ is one of the symbols $\{<, \leq, =, \geq, >\}$. We denote by $\mathcal{G}(X)$ the set of guards over $X$. The satisfaction relation for guards over clock valuations is defined in a natural way, and we write $\nu \models g$ if the clock valuation $\nu$ satisfies the guard $g$. We denote by $\mathsf{AP}$ a finite set of atomic propositions.

We now define the timed automaton model, which has been introduced in the early nineties [AD90, AD94].

**Definition 2.1.** A *timed automaton* over $\mathsf{AP}$ is a tuple $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ such that: (i) $L$ is a finite set of locations, (ii) $X$ is a finite set of clocks, (iii) $E \subseteq L \times \mathcal{G}(X) \times 2^X \times L$ is a finite set of edges, (iv) $\mathcal{I} : L \to \mathcal{G}(X)$ assigns an invariant to each location, and $(v)$ $\mathcal{L} : L \to 2^{\mathsf{AP}}$ is a labelling function.

We may omit the labelling function (in case we are only interested in an internal accepting condition, *i.e.* that only depends on the locations). Note that we could also specify an initial location, but that will not be really useful later, that is why we removed that component from standard timed-automata definition.

If $e$ is an edge of $\mathcal{A}$, we write $\mathsf{source}(e)$ (resp. $\mathsf{target}(e)$) the source (resp. target) of $e$ defined by $\ell$ (resp. $\ell'$) if $e = (\ell, g, Y, \ell')$. The semantics of a timed automaton $\mathcal{A}$ is a timed transition system $T_{\mathcal{A}}$ whose states are pairs $s = (\ell, v) \in L \times \mathbb{R}_+^X$ with $v \models \mathcal{I}(\ell)$, and whose transitions are of the form $(\ell, v) \xrightarrow{\tau, e} (\ell', v')$ if there exists an edge $e = (\ell, g, Y, \ell')$ such that for every $0 \leq \tau' \leq \tau$, $v + \tau' \models \mathcal{I}(\ell)$, $v + \tau \models g$, $v' = [Y \leftarrow 0]v$, and $v' \models \mathcal{I}(\ell')$. We extend the labelling function $\mathcal{L}$ to states: $\mathcal{L}((\ell, v)) = \mathcal{L}(\ell)$ for every state $(\ell, v)$. A finite (resp. infinite) *run* $\varrho$ of $\mathcal{A}$ is a finite (resp. infinite) sequence of transitions, *i.e.*,

$$\varrho = s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} s_2 \dots$$

We write $\mathsf{Runs}_f(\mathcal{A}, s_0)$ (resp. $\mathsf{Runs}(\mathcal{A}, s_0)$) for the set of finite runs (resp. infinite runs) of $\mathcal{A}$ from state $s_0$. If $\varrho$ is a finite run in $\mathcal{A}$, we write $\mathsf{last}(\varrho)$ for the last state of $\varrho$. For $s$ a state of $\mathcal{A}$, $(e_i)_{1 \leq i \leq n}$ a finite sequence of edges of $\mathcal{A}$, and $\mathcal{C}$ a constraint over $n$ variables $(t_i)_{1 \leq i \leq n}$, the *(symbolic) path* starting from $s$, determined by $(e_i)_{1 \leq i \leq n}$, and constrained by $\mathcal{C}$, is the following set of runs:

$$\pi_{\mathcal{C}}(s, e_1 \dots e_n) \;=\; \{\varrho = s \xrightarrow{\tau_1, e_1} s_1 \dots \xrightarrow{\tau_n, e_n} s_n \in \mathsf{Runs}_f(\mathcal{A}, s) \;\mid (\tau_i)_{1 \leq i \leq n} \models \mathcal{C}\},$$

where $(\tau_i)_{1 \leq i \leq n} \models \mathcal{C}$ stands for "$\tau_i$'s satisfy the constraint $\mathcal{C}$" with the intuitive meaning.

If $\mathcal{C}$ is equivalent to 'true', we simply write $\pi(s, e_1 \dots e_n)$. Let $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(s, e_1 \dots e_n)$ be a finite symbolic path, we define the *cylinder* generated by $\pi_{\mathcal{C}}$ as:

$$\mathsf{Cyl}(\pi_{\mathcal{C}}) = \{\varrho \in \mathsf{Runs}(\mathcal{A}, s) \mid \exists \varrho' \in \mathsf{Runs}_f(\mathcal{A}, s), \text{finite prefix of } \varrho, \text{ s.t. } \varrho' \in \pi_{\mathcal{C}}\}.$$

In the following, we will also use infinite symbolic paths defined, given $s$ a state of $\mathcal{A}$ and $(e_i)_{i\geq 1}$ an infinite sequence of edges, as:

$$\pi(s, e_1 \dots) = \{\varrho = s \xrightarrow{\tau_1, e_1} s_1 \dots \in \mathsf{Runs}(\mathcal{A}, s)\}.$$

If $\varrho \in \mathsf{Runs}(\mathcal{A}, s)$, we write $\pi_\varrho$ for the unique symbolic path containing $\varrho$. Given $s$ a state of $\mathcal{A}$ and $e$ an edge, we define $I(s, e) = \{\tau \in \mathbb{R}_+ \mid \exists s' \text{ s.t. } s \xrightarrow{\tau, e} s'\}$ and $I(s) = \bigcup_e I(s, e)$. Note that $I(s, e)$ is an interval, whereas $I(s)$ is a finite union of intervals.

The timed automaton $\mathcal{A}$ is *non-blocking* if, for every state $s$, $I(s) \neq \emptyset$. The timed automaton $\mathcal{A}$ is *reactive* if, for every state $s$, $I(s) = \mathbb{R}_+$; in this case we may omit the invariant function and simply write $\mathcal{A} = (L, X, E, \mathcal{L})$.

2.2. **The timed region automaton.** The well-known region automaton construction is a finite abstraction of timed automata which can be used for verifying many properties like $\omega$-regular untimed properties [AD94]. Roughly, the region automaton of $\mathcal{A}$ is the quotient of $T_{\mathcal{A}}$ by a finite-index equivalence relation over clock valuations. Here we will use a timed version of this construction, that we define now.

Let $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ be a timed automaton, and write $M$ for the maximal constant used in guards and invariants in $\mathcal{A}$. We define its region equivalence $\equiv_{\mathcal{A}}$ over the set of valuations $\mathbb{R}_+^X$ as follows: given $v, v' \in \mathbb{R}_+^X$, $v \equiv_{\mathcal{A}} v'$ if and only if the following conditions are satisfied:

- for every $x \in X$, either $v(x), v'(x) > M$, or $\lfloor v(x) \rfloor = \lfloor v'(x) \rfloor$, and in the last case, $\{v(x)\} = 0$ iff $\{v'(x)\} = 0$;[4]
- for every $x, y \in X$ such that $v(x), v(y) \leq M$, $\{v(x)\} \leq \{v(y)\}$ iff $\{v'(x)\} \leq \{v'(y)\}$.

This equivalence relation has finite index, the equivalence classes are called *regions*, and we write $R_{\mathcal{A}}$ for the set of regions. If $v$ is a valuation, we write $[v]_{\mathcal{A}}$ or simply $[v]$ for the (unique) region to which $v$ belongs. Also, for $r$ a region, $\mathsf{guard}(r)$ denotes the minimal guard characterising $r$.

**Remark 2.2.** The above region equivalence is the most standard one, but several rougher equivalences could also be used, as soon as they yield a time-abstract bisimilar quotient. For instance, for single-clock timed automata, we will later use a rougher notion of region equivalence [LMS04] that will improve the complexity of our algorithms.

The *timed region automaton* of $\mathcal{A}$ is the timed automaton $\mathsf{R}(\mathcal{A}) = (Q, X, T, \kappa, \lambda)$ such that $Q = L \times R_{\mathcal{A}}$, and:

- $\kappa((\ell, r)) = \mathcal{I}(\ell)$, and $\lambda((\ell, r)) = \mathcal{L}(\ell)$ for all $(\ell, r) \in L \times R_{\mathcal{A}}$;
- $T \subseteq (Q \times \mathsf{guard}(R_{\mathcal{A}}) \times E \times 2^X \times Q)$, and $(\ell, r) \xrightarrow{\mathsf{guard}(r''), e, Y} (\ell', r')$ is in $T$ iff there exists $e = \ell \xrightarrow{g, Y} \ell'$ in $E$ s.t. there exist $v \in r$, $\tau \in \mathbb{R}_+$ with $(\ell, v) \xrightarrow{\tau, e} (\ell', v')$, $v + \tau \in r''$ and $v' \in r'$.

As an example, a timed automaton and its associated timed region automaton are depicted in Figure 2. We recover the usual region automaton of [AD94] by labelling the transitions

[4] $\lfloor \cdot \rfloor$ (resp. $\{\cdot\}$) denotes the integral (resp. fractional) part.

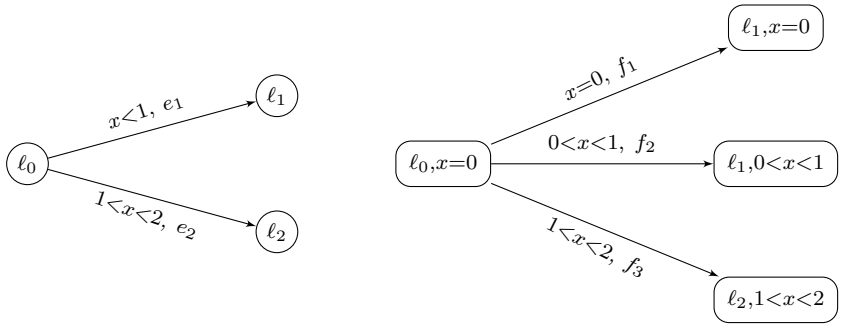



Figure 2: An automaton and its timed region automaton.

with '$e$' instead of '$\mathsf{guard}(r''), e, Y$', and by interpreting $\mathsf{R}(\mathcal{A})$ as a finite automaton. The above timed interpretation satisfies strong timed bisimulation properties that we do not detail here. To every finite symbolic path $\pi((\ell, v), e_1 \dots e_n)$ in $\mathcal{A}$ corresponds a finite set of paths $\pi(((\ell, [v]), v), f_1 \dots f_n)$ in $\mathsf{R}(\mathcal{A})$, each one corresponding to a choice in the regions that are crossed. If $\varrho$ is a run in $\mathcal{A}$, we denote $\iota(\varrho)$ its unique image in $\mathsf{R}(\mathcal{A})$. Note that if $\mathcal{A}$ is non-blocking (resp. reactive), then so is $\mathsf{R}(\mathcal{A})$.

In the rest of the paper we assume that timed automata are non-blocking, even though general timed automata could also be handled (but at a technical extra cost).

2.3. **Specification languages.** We fix a finite set of atomic propositions $\mathsf{AP}$, and a timed automaton $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ over $\mathsf{AP}$.

2.3.1. *Properties over* $\mathsf{AP}$*..* A *property over* $\mathsf{AP}$ is a subset $P$ of $\left(2^{\mathsf{AP}}\right)^\omega$. An infinite run $\varrho = s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} \dots$ satisfies the property whenever $\mathcal{L}(s_0)\mathcal{L}(s_1)\mathcal{L}(s_2)\dots \in P$.

More generally a *timed property over* $\mathsf{AP}$ is a subset $P$ of $2^{\mathsf{AP}} \cdot \left(\mathbb{R}_+ \cdot 2^{\mathsf{AP}}\right)^\omega$. An infinite run $\varrho = s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} \dots$ satisfies the property $P$ whenever $\mathcal{L}(s_0)\,\tau_1\,\mathcal{L}(s_1)\,\tau_2\,\mathcal{L}(s_2)\dots \in P$.

In both cases, we write $\varrho \models P$ if $\varrho$ satisfies the property $P$, and we write:

$$[\![P]\!]_{\mathcal{A},s} \stackrel{\text{def}}{=} \{\varrho \in \mathsf{Runs}(\mathcal{A}, s) \mid \varrho \models P\}\,.$$

**Remark 2.3.** Obviously, timed properties generalise (untimed) ones.

2.3.2. *$\omega$-regular properties.* $\omega$-regularity is a standard notion in computer science to characterise simple sets of infinite behaviours. We will only define here $\omega$-regularity for untimed properties, though the concept exists for timed properties as well.

Typical $\omega$-regular properties are Büchi and Muller properties. A *Büchi property* over $\mathsf{AP}$ is a(n untimed) property $P$ such that there exists $F \subseteq \mathsf{AP}$ with $P = \{u_0 u_1 \dots \mid \{j \mid F \cap u_j \neq \emptyset\}$ is infinite$\}$. A *Muller property* over $\mathsf{AP}$ is a property $P$ such that there exists $\mathcal{F} \subseteq 2^{\mathsf{AP}}$ with $P = \{u_0 u_1 \dots \mid \{j \mid u_j \in \mathcal{F}\}$ is infinite$\}$.

An $\omega$-regular property will be said *internal* for $\mathcal{A}$ whenever there is a bijection $\beta$ between $L$ and $\mathsf{AP}$, and for each $\ell \in L$, $\mathcal{L}(\ell)$ is the singleton $\{\beta(\ell)\}$. That is, this allows to specify which states are visited infinitely often. In that case, we will interpret such properties on

timed automata even though no labelling function has been given (it is then implicit). It is well known that (untimed) automata equipped with internal Büchi or Muller acceptance conditions capture untimed $\omega$-regular properties. This is also the case for deterministic Muller automata.

2.3.3. *Safety, reachability, and prefix-independent properties.* We now define simple $\omega$-regular properties.

According to [CMP92], a property $P$ over $\mathsf{AP}$ is a *safety property* whenever for every $w = u_0u_1\ldots \in \left(2^{\mathsf{AP}}\right)^\omega$, $w \notin P$ iff there exists $i$ such that for every $w' = u_0\ldots u_iu'_{i+1}u'_{i+2}\ldots \in \left(2^{\mathsf{AP}}\right)^\omega$, $w' \notin P$. That is, a safety property is violated by a finite prefix. A *simple safety property* $P$ is characterised by $F \subseteq \mathsf{AP}$, and is defined by $P = \{u_0u_1\ldots \mid \forall j,\ u_j \in F\}$.

The negation of a safety property is a *reachability property*: $P$ is a reachability property whenever for every $w = u_0u_1\ldots \in \left(2^{\mathsf{AP}}\right)^\omega$, $w \in P$ iff there exists $i$ such that for every $w' = u_0\ldots u_iu'_{i+1}u'_{i+2}\ldots \in \left(2^{\mathsf{AP}}\right)^\omega$, $w' \in P$. That is, a reachability property is validated by a finite prefix. A *simple reachability property* $P$ is characterised by $F \subseteq \mathsf{AP}$, and is defined by $P = \{u_0u_1\ldots \mid \exists j,\ u_j \in F\}$.

Another interesting notion is the one of *prefix-independent property* $P$, which is such that for every $w = u_0u_1\ldots \in \left(2^{\mathsf{AP}}\right)^\omega$, $w \in P$ iff for every $i$, $w' = u_iu_{i+1}\ldots \in P$. That is, the property is satisfied or not independently of its prefix. In particular, Büchi and Muller properties are prefix-independent. Note that a property $P$ is prefix-independent if and if its validity only depends on the set of elements of $2^{\mathsf{AP}}$ which is encountered infinitely often.

2.3.4. *The temporal logic* $\mathsf{LTL}$. We consider the linear temporal logic $\mathsf{LTL}$ [Pnu77] over $\mathsf{AP}$, defined inductively as:

$$\mathsf{LTL} \ni \varphi \ ::= \ p \ \mid \ \varphi \vee \varphi \ \mid \ \varphi \wedge \varphi \ \mid \ \neg\varphi \ \mid \ \varphi\,\mathbf{U}\,\varphi$$

where $p \in \mathsf{AP}$ is an atomic proposition. We use classical shorthands like $\mathtt{tt} \stackrel{\text{def}}{=} p \vee \neg p$, $\mathtt{ff} \stackrel{\text{def}}{=} p \wedge \neg p$, $\mathbf{F}\,\varphi \stackrel{\text{def}}{=} \mathtt{tt}\,\mathbf{U}\,\varphi$, and $\mathbf{G}\,\varphi \stackrel{\text{def}}{=} \neg\mathbf{F}\,(\neg\varphi)$. We assume the reader is familiar with the semantics of $\mathsf{LTL}$, that we interpret here on infinite runs of a timed automaton.

Each formula $\varphi$ of $\mathsf{LTL}$ gives rise to a property $P_\varphi$, in the sense given above. Let $w = u_0u_1\ldots \in \left(2^{\mathsf{AP}}\right)^\omega$, then:

$$\begin{aligned}
w \in P_p &\quad\Leftrightarrow\quad p \in u_0\\
w \in P_{\varphi_1\vee\varphi_2} &\quad\Leftrightarrow\quad w \in P_{\varphi_1} \cup P_{\varphi_2}\\
w \in P_{\varphi_1\wedge\varphi_2} &\quad\Leftrightarrow\quad w \in P_{\varphi_1} \cap P_{\varphi_2}\\
w \in P_{\neg\varphi} &\quad\Leftrightarrow\quad w \notin P_\varphi\\
w \in P_{\varphi_1\,\mathbf{U}\,\varphi_2} &\quad\Leftrightarrow\quad \exists i \geq 0 \text{ s.t. } w_{\geq i} \in P_{\varphi_2} \text{ and } \forall 0 \leq j < i,\ w_{\geq j} \in P_{\varphi_1}
\end{aligned}$$

where $w_{\geq k} = u_ku_{k+1}\ldots$ for every index $k$.

The semantics of a formula $\varphi$ over infinite runs of $\mathcal{A}$ is derived from that of property $P_\varphi$. One can easily be convinced that we recover the standard semantics of $\mathsf{LTL}$. If $\varphi$ is an $\mathsf{LTL}$ formula and $\varrho \in \mathsf{Runs}(\mathcal{A}, s)$, we write $\varrho \models \varphi$ whenever $\varrho \models P_\varphi$. We also write $[\![\varphi]\!]_{\mathcal{A},s}$ for $[\![P_\varphi]\!]_{\mathcal{A},s}$.

2.3.5. *Specifications given as deterministic timed automata.* A *specification $\omega$-regular timed automaton* is a tuple $\mathcal{B} = (\mathsf{L}, \mathsf{i}_0, \mathsf{X}, \mathsf{AP}, \mathsf{E}, \mathcal{F})$ such that:

- $\mathsf{L}$ is a finite set of locations, and $\mathsf{i}_0 : 2^{\mathsf{AP}} \to \mathsf{L}$ is an input function;
- $\mathsf{X}$ is a finite set of clocks;
- $\mathsf{AP}$ is a finite set of atomic propositions;
- $\mathsf{E} \subseteq \mathsf{L} \times \mathcal{G}(\mathsf{X}) \times 2^{\mathsf{AP}} \times 2^{\mathsf{X}} \times \mathsf{L}$ is a finite set of edges;
- $\mathcal{F}$ is an internal $\omega$-regular prefix-independent condition;
- it is deterministic: for all edges $(\mathsf{l} \xrightarrow{\mathsf{g}_1,u,\mathsf{Y}_1} \mathsf{l}_1)$ and $(\mathsf{l} \xrightarrow{\mathsf{g}_2,u,\mathsf{Y}_2} \mathsf{l}_2)$ in $\mathsf{E}$, $\mathsf{g}_1 \wedge \mathsf{g}_2$ is not satisfiable;
- it is complete: for every every $\mathsf{l} \in \mathsf{L}$, for every $u \in 2^{\mathsf{AP}}$, for every $\mathsf{v} \in \mathbb{R}_+^{\mathsf{X}}$, for every $\tau \in \mathbb{R}_+$, there exists $(\mathsf{l} \xrightarrow{\mathsf{g},u,\mathsf{Y}} \mathsf{l}') \in \mathsf{E}$ such that $\mathsf{v} + \tau \models \mathsf{g}$.

Runs in $\mathcal{B}$ will be defined in a very similar way as runs in standard timed automata. Only labels of transitions will be slightly different. The runs of $\mathcal{B}$ are therefore of the form:

$$(\mathsf{l}_0, \mathsf{v}_0) \xrightarrow{\tau_1,u_1} (\mathsf{l}_1, \mathsf{v}_1) \xrightarrow{\tau_2,u_2} \dots$$

where conditions on valuations are those expected, and labels $u_i$'s are those given by the edges that are taken. Such a run is accepted by $\mathcal{B}$ whenever the sequence $(\mathsf{l}_i)_{i\geq 0}$ satisfies the $\omega$-regular condition $\mathcal{F}$.

Such a specification automaton $\mathcal{B}$ naturally gives rise to a timed property $P_{\mathcal{B}}$ defined as follows. Let $w = u_0\tau_1u_1\tau_2\ldots \in 2^{\mathsf{AP}} \cdot \left(\mathbb{R}_+ \cdot 2^{\mathsf{AP}}\right)^{\omega}$. There is a unique run $\kappa_w = (\mathsf{l}_0, \mathbf{0}_{\mathsf{X}}) \xrightarrow{\tau_1,u_1} (\mathsf{l}_1, \mathsf{v}_1) \xrightarrow{\tau_2,u_2} \dots$ in automaton $\mathcal{B}$ where $\mathsf{l}_0 = \mathsf{i}_0(u_0)$. The existence of $\kappa_w$ follows from the completeness of $\mathcal{B}$ and its uniqueness from the determinism. Then, $w \in P_{\mathcal{B}}$ iff $\kappa_w$ is an accepting run in $\mathcal{B}$.

The semantics of a specification timed automaton $\mathcal{B}$ over infinite runs of $\mathcal{A}$ is derived from that of property $P_{\mathcal{B}}$. If $\mathcal{B}$ is a specification timed automaton and $\varrho \in \mathsf{Runs}(\mathcal{A}, s)$, we write $\varrho \models \mathcal{B}$ whenever $\varrho \models P_{\mathcal{B}}$. We also write $[\![\mathcal{B}]\!]_{\mathcal{A},s}$ for $[\![P_{\mathcal{B}}]\!]_{\mathcal{A},s}$.

**Remark 2.4.** If the accepting condition $\mathcal{F}$ is a Büchi (or Muller) condition, then $\mathcal{B}$ will be called a specification Büchi (or Muller) timed automaton. If $\mathsf{X} = \emptyset$, we will speak of a specification $\omega$-regular (untimed) automaton. It is well known [VW94] and [GThW02, Chapter 3] that for any $\mathsf{LTL}$ formula $\varphi$, there is a (deterministic) specification Muller untimed automaton $\mathcal{B}_\varphi$ that characterises $\varphi$, that is: for every run $\varrho$, $\varrho \models \varphi$ iff $\varrho \models \mathcal{B}_\varphi$. In that case, obviously, $[\![\varphi]\!]_{\mathcal{A},s} = [\![\mathcal{B}_\varphi]\!]_{\mathcal{A},s}$ for every $\mathcal{A}$ and $s$.

## 3. Stochastic timed automata: the semantics

In this section we define a probabilistic semantics for timed automata. Probabilities will rule time elapsing as well as choices between enabled events. This will define a purely stochastic process: intuitively, from a state, we will first randomly choose a delay among all possible delays, then we will randomly choose an edge among all those which are enabled.

The sequel assumes some basics of measure theory and probabilities, that can be found in classical text books.

3.1. **Probability measure on runs of a timed automaton.** Let $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ be a timed automaton. We will assign probability distributions from every state of $\mathcal{A}$ both over delays and over enabled moves. Let $s$ be a state of $\mathcal{A}$. The probability distribution from $s$ over delays is a probability measure $\mu_s$ over $\mathbb{R}_+$ (equipped with the standard Borel $\sigma$-algebra) which satisfies the following requirements, denoted $(\star)$ in the sequel:

**(H1):** $\mu_s(I(s)) = \mu_s(\mathbb{R}_+) = 1$,[5]

**(H2):** Writing $\lambda$ for the standard Lebesgue measure on $\mathbb{R}_+$, if $\lambda(I(s)) > 0$, then $\mu_s$ is equivalent[6] to $\lambda$ on $I(s)$; Otherwise, $\mu_s$ is equivalent to the uniform distribution over points of $I(s)$.

This last condition denotes some kind of fairness w.r.t. enabled transitions when only punctual delays are possible, in that we cannot disallow one transition by putting a probability 0 to delays enabling that transition.

We also assume a probability distribution $p_s$ over edges, such that for every edge $e$, $p_s(e) > 0$ iff $e$ is enabled in $s$ (*i.e.*, $s \xrightarrow{e} s'$ for some $s'$). Moreover, to simplify, we assume that $p_s$ is given by weights on transitions, as it is classically done for resolving non-determinism: we associate with each edge $e$ a weight $w(e) > 0$, and for every state $s$, for every edge $e$, $p_s(e) = 0$ if $e$ is not enabled in $s$, and $p_s(e) = w(e)/(\sum_{e' \text{ enabled in } s} w(e'))$ otherwise. As a consequence, if $s$ and $s'$ are region equivalent, then for every edge $e$, $p_s(e) = p_{s'}(e)$.

**Definition 3.1** (Stochastic timed automaton)**.** A *stochastic timed automaton* is a tuple $\langle \mathcal{A}, \mu, w \rangle$ consisting of a timed automaton $\mathcal{A}$ equipped with probability measures $\mu = (\mu_s)_{s \in L \times \mathbb{R}_+^X}$ satisfying $(\star)$, and positive weights $w = (w_e)_{e \in E}$.

Note that when measures are clear or implicit in the context, we will simply write $\mathcal{A}$ for $\langle \mathcal{A}, \mu, w \rangle$. We now show how we define a probability measure $\mathbb{P}_\mathcal{A}$ on infinite runs of $\langle \mathcal{A}, \mu, w \rangle$.

We fix a stochastic timed automaton $\langle \mathcal{A}, \mu, w \rangle$, which satisfies $(\star)$. We define a measure, that we also note $\mathbb{P}_\mathcal{A}$, over finite symbolic paths from state $s$ as follows:

$$\mathbb{P}_\mathcal{A}(\pi(s, e_1 \dots e_n)) \;=\; \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}_\mathcal{A}(\pi(s_t, e_2 \dots e_n))\, \mathrm{d}\mu_s(t)$$

where $s \xrightarrow{t} (s+t) \xrightarrow{e_1} s_t$, and we initialise with $\mathbb{P}_\mathcal{A}(\pi(s)) = 1$. The formula for $\mathbb{P}_\mathcal{A}$ relies on the fact that the probability of taking transition $e_1$ at time $t$ coincides with the probability of waiting $t$ time units and then choosing $e_1$ among the enabled transitions, *i.e.*, $p_{s+t}(e_1)\mathrm{d}\mu_s(t)$. Note that, time passage and actions are independent events.

The value $\mathbb{P}_\mathcal{A}(\pi(s, e_1 \dots e_n))$ is the result of $n$ successive one-dimensional integrals, but it can also be viewed as the result of an $n$-dimensional integral. Hence, we can easily extend the above definition to finite constrained paths $\pi_\mathcal{C}(s, e_1 \dots e_n)$ when $\mathcal{C}$ is Borel-measurable. This extension to constrained paths is needed to measure rather complex properties, like Zeno behaviours or those expressed as specification timed automata. The measure $\mathbb{P}_\mathcal{A}$ can then be defined on cylinders, letting $\mathbb{P}_\mathcal{A}(\mathsf{Cyl}(\pi)) = \mathbb{P}_\mathcal{A}(\pi)$ if $\pi$ is a finite (constrained) symbolic path. Finally we extend $\mathbb{P}_\mathcal{A}$ in a standard and unique way to the $\sigma$-algebra generated by these cylinders (using Caratheodory's theorem), that we note $\Omega^s_\mathcal{A}$.

We first check that $\mathbb{P}_\mathcal{A}$ defined as such is a probability measure.

**Proposition 3.2.** For every state $s$ of $\mathcal{A}$, $\mathbb{P}_\mathcal{A}$ is a probability measure over $(\mathsf{Runs}(\mathcal{A}, s), \Omega^s_\mathcal{A})$.

[5]Note that this is possible, as we assume $\mathcal{A}$ is non-blocking, hence $I(s) \neq \emptyset$ for every state $s$ of $\mathcal{A}$.

[6]Two measures $\nu$ and $\nu'$ are *equivalent* whenever for each measurable set $A$, $\nu(A) = 0 \Leftrightarrow \nu'(A) = 0$.

The proof of this proposition justifies *a posteriori* the above construction for the probability measure $\mathbb{P}_{\mathcal{A}}$. It goes as follows: first prove that $\mathbb{P}_{\mathcal{A}}$ is a probability measure on the set of constrained symbolic paths of length $n$ (for all $n$), then extend this result to the ring generated by all constrained symbolic paths and finally use Caratheodory's extension theorem to establish that $\mathbb{P}_{\mathcal{A}}$ is a probability measure on the set of all runs. The complete proof is rather technical, and therefore postponed to Appendix A, page 41.

**Example 3.3.** Consider the running stochastic timed automaton $\mathcal{A}_{\mathsf{running}}$ on Figure 3. Assume for all states $s_t = (\ell_0, t)$ both uniform distributions over delays and discrete moves: $\mu_{s_0} = \lambda$ is the uniform distribution over $[0,1]$ and $\mu_{s_t} = \frac{\lambda}{1-t}$ is the uniform distribution over $[t,1]$; the weight of each edge is 1. Then $\mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\mathsf{Cyl}(\pi(s_0, e_1e_1))) = \frac{1}{4}$ and $\mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\pi(s_0, {e_1}^\omega)) = 0$. Indeed:

$$\begin{aligned}
\mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\pi(s_0, e_1e_1)) &= \int_{t\in I(s_0,e_1)} p_{s_0+t}(e_1)\, \mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\pi((\ell_0,t),e_1))\, \mathrm{d}\mu_{s_0}(t) \\
&= \int_0^1 \frac{1}{2}\, \mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\pi((\ell_0,t),e_1))\, \mathrm{d}\lambda(t) \\
&= \frac{1}{2}\int_0^1 \left( \int_{t\in I(s_t,e_1)} p_{s_t+u}(e_1)\, \mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\pi((\ell_1,u)))\, \mathrm{d}\mu_{s_t}(u) \right) \mathrm{d}\lambda(t) \\
&= \frac{1}{2}\int_0^1 \left( \int_t^1 \frac{1}{2}\, \frac{1}{1-t}\, \mathrm{d}\lambda(u) \right) \mathrm{d}\lambda(t) = \frac{1}{4}.
\end{aligned}$$

In a similar way one can show that $\mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\pi(s_0, {e_1}^n)) = \frac{1}{2^n}$, for $n \in \mathbb{N}$; and thus conclude that $\mathbb{P}_{\mathcal{A}_{\mathsf{running}}}(\pi(s_0, {e_1}^\omega)) = 0$. ⌟

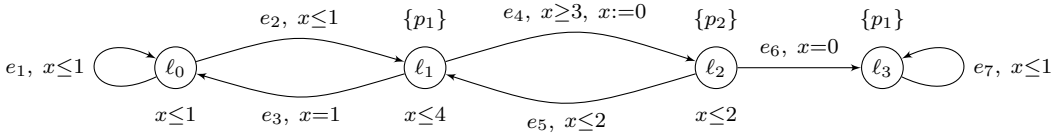



Figure 3: The stochastic timed automaton $\mathcal{A}_{\mathsf{running}}$.

3.2. **Measuring Zeno runs.** In timed automata, and more generally in continuous-time models, some behaviours are *Zeno*. Recall that a run $\varrho = s_0 \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} \dots$ of a timed automaton is *Zeno* if $\sum_{i=1}^{\infty} \tau_i < \infty$ (*i.e.*, infinitely many actions happen in a finite amount of time). Zeno behaviours are problematic since they most of the time have no physical interpretation. As argued in [DP03], some fairness constraints are often put on executions, enforcing non-Zeno behaviours, but in probabilistic systems, probabilities are supposed to replace fairness assumptions, and it is actually the case in continuous-time Markov chains in which Zeno runs are negligible (that is, have probability 0) [BHHK03].

We observe that, for any stochastic timed automaton $\mathcal{A}$, the set of Zeno behaviours from a state $s$ is measurable. It can indeed be expressed as

$$\bigcup_{M\in\mathbb{N}} \bigcap_{n\in\mathbb{N}} \bigcup_{(e_1,\dots,e_n)\in E^n} \pi_{\tau_1+\dots+\tau_n\le M}(s, e_1 \dots e_n)\,.$$

It therefore makes sense to compute the probability of Zeno behaviours, and to check whether Zeno behaviours are negligible or not. Being negligible would be a desirable property, as argued before. However, in general this is hopeless since some timed automata are *inherently* Zeno. For instance, all runs are Zeno in the automaton consisting of a single location with a non-resetting loop guarded by $x \leq 1$. In the following, we will discuss Zenoness for several classes of stochastic timed automata.

In the rest of the paper, if $\mathcal{A}$ is a stochastic timed automaton and $\varrho$ is a run of $\mathcal{A}$, we write $\varrho \models \mathsf{Zeno}$ whenever $\varrho$ is Zeno. If $s$ is a state of $\mathcal{A}$, we then also write $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno})$ for the probability of the set of Zeno runs in $\mathcal{A}$ from $s$. We associate the following decision problem, that we call *almost-sure non-Zenoness* problem: given $\mathcal{A}$ a stochastic timed automaton, and $s$ a state of $\mathcal{A}$, does $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = 0$?

3.3. **From timed automata to timed region automata.** In this part we establish a strong relation between a stochastic timed automaton and its stochastic timed region automaton. We let $\langle \mathcal{A}, \mu^{\mathcal{A}}, w^{\mathcal{A}} \rangle$ be a stochastic timed automaton. The structure of the corresponding stochastic timed region automaton is obviously $\mathsf{R}(\mathcal{A})$. We need now to choose properly probability measures $\mu^{\mathsf{R}(\mathcal{A})}$ and weights $w^{\mathsf{R}(\mathcal{A})}$ for $\mathsf{R}(\mathcal{A})$ so that measures of runs are preserved *via* the mapping $\iota$ mentioned in Section 2.2. We assume that the probability measures in $\mathsf{R}(\mathcal{A})$ satisfy the following conditions: for every state $s$ in $\mathcal{A}$, $\mu_s^{\mathcal{A}} = \mu_{\iota(s)}^{\mathsf{R}(\mathcal{A})}$, and for every edge $e \in E$, $w^{\mathcal{A}}(e) = w^{\mathsf{R}(\mathcal{A})}(f)$ whenever $f$ corresponds to $e$. Under those conditions we show the following transfer properties between $\mathcal{A}$ and $\mathsf{R}(\mathcal{A})$.

**Lemma 3.4.** Let $\langle \mathcal{A}, \mu^{\mathcal{A}}, w^{\mathcal{A}} \rangle$ be a stochastic timed automaton, and let $\langle \mathsf{R}(\mathcal{A}), \mu^{\mathsf{R}(\mathcal{A})}, w^{\mathsf{R}(\mathcal{A})} \rangle$ be the corresponding stochastic timed region automaton as defined above. Then, for every set $S$ of runs in $\mathcal{A}$ we have: $S \in \Omega_{\mathcal{A}}^s$ iff $\iota(S) \in \Omega_{\mathsf{R}(\mathcal{A})}^{\iota(s)}$, and in this case $\mathbb{P}_{\mathcal{A}}(S) = \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(S))$.

To establish Lemma 3.4 it is sufficient to prove that the measures coincide on finite constrained paths, since it implies that they agree on cylinders and by uniqueness of the extension on any measurable set of infinite runs. The complete proof is given in Appendix A, page 43.

Thanks to Lemma 3.4, we will be able to lift results proven on $\mathsf{R}(\mathcal{A})$ to $\mathcal{A}$.

3.4. **Almost-sure satisfaction.** Let $\langle \mathcal{A}, \mu, w \rangle$ be a stochastic timed automaton over $\mathsf{AP}$ and $s$ be a state of $\mathcal{A}$.

We refine the notion of timed properties that were defined in Section 2.3. Let $P \subseteq 2^{\mathsf{AP}} \cdot \left(\mathbb{R}_+ \cdot 2^{\mathsf{AP}}\right)^{\omega}$ be a timed property over $\mathsf{AP}$. We say that $P$ is $(\mathcal{A}, s)$-*measurable* whenever $[\![P]\!]_{\mathcal{A},s} \in \Omega_{\mathcal{A}}^s$. We say $P$ is $\mathcal{A}$-*measurable* (resp. *measurable*) whenever it is $(\mathcal{A}, s)$-measurable for every state $s$ (resp. it is $\mathcal{A}$-measurable for every $\mathcal{A}$).

The following lemma establishes the measurability of several classes of properties, and is proven in Appendix A, page 44.

**Lemma 3.5.** $\omega$-regular properties and properties given as $\mathsf{LTL}$ formulas are measurable. Timed properties given as specification Büchi or Muller timed automata are measurable.

In the sequel, if $P$ is a measurable property over $\mathsf{AP}$, we write $\mathbb{P}_{\mathcal{A}}(s \models P)$ for $\mathbb{P}_{\mathcal{A}}\{\varrho \in \mathsf{Runs}(\mathcal{A}, s) \mid \varrho \models P\}$.

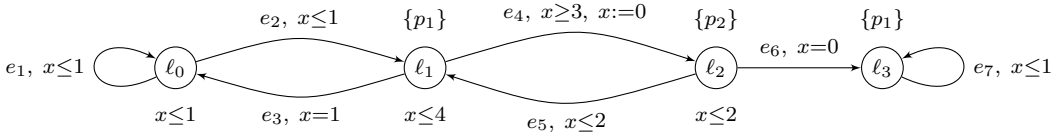
**Definition 3.6.** Let $s$ be a state of $\mathcal{A}$. Assume $P$ is an $(\mathcal{A}, s)$-measurable property over $\mathsf{AP}$. We say that $\mathcal{A}$ *almost-surely satisfies* $P$, from $s$, and we then write $\mathcal{A}, s \models_{\mathbb{P}} P$, whenever $\mathbb{P}_{\mathcal{A}}(s \models P) = 1$. The *almost-sure model-checking problem* asks, given $\mathcal{A}$, $s$ and $P$, whether $\mathcal{A}, s \models_{\mathbb{P}} P$.

The following corollary is an immediate consequence of Lemma 3.4.

**Corollary 3.7.** Let $\mathcal{A}$ be a stochastic timed automaton, $s$ a state of $\mathcal{A}$, and $\varphi$ a measurable property over $\mathsf{AP}$. Then,

$$\mathcal{A}, s \models_{\mathbb{P}} \varphi \;\Leftrightarrow\; \mathsf{R}(\mathcal{A}), \iota(s) \models_{\mathbb{P}} \varphi\,.$$

**Example 3.8.** Consider $\mathcal{A}_{\mathsf{running}}$ again from Figure 3, and reproduced below, with initial state $s_0 = (\ell_0, 0)$ and assuming uniform distributions over delays and uniform distribution over discrete moves in all states. Then, $\mathcal{A}_{\mathsf{running}}, s_0 \models_{\mathbb{P}} \mathbf{F}\,(p_1 \wedge \mathbf{G}\,(p_1 \Rightarrow \mathbf{F}\,p_2))$. Indeed, in state $(\ell_0, \nu)$ with $0 \leq \nu \leq 1$, the probability of firing $e_2$ (after some delay) is always $1/2$ (guards of $e_1$ and $e_2$ are the same, there is thus a uniform distribution over the two edges), the location $\ell_1$ is eventually reached with probability 1. In $\ell_1$, the transition $e_3$ will unlikely happen, because its guard $x = 1$ is too much "small" compared to the guard $x \geq 3$ of the transition $e_4$. The same phenomenon arises in location $\ell_2$ between the transitions $e_5$ and $e_6$. In conclusion, the runs of the timed automaton $\mathcal{A}_{\mathsf{running}}$ (from $s_0$) almost surely follow sequences of transitions of the form $e_1{}^* e_2 (e_4 e_5)^\omega$. Hence, with probability 1, the formula $\mathbf{F}\,(p_1 \wedge \mathbf{G}\,(p_1 \Rightarrow \mathbf{F}\,p_2))$ is satisfied. Note that the latter formula is not satisfied in $\mathcal{A}_{\mathsf{running}}$ from $s_0$ (under the classical $\mathsf{LTL}$ semantics), since some runs violate it: '*staying in* $\ell_0$ *forever*', '*reaching* $\ell_3$', etc... All these counter-examples are unlikely and vanish thanks to our probabilistic semantics. ⌟




Our aim is to *decide* the almost-sure model-checking problem. It is clear that given a measurable property $P$, the value $\mathbb{P}_{\mathcal{A}}(s \models P)$ depends on the measures $\mu$ and $w$. However, we show later that whether $\mathcal{A}, s \models_{\mathbb{P}} P$ is independent of the precise values of $\mu$ and $w$. To prove this, we design a finite abstraction, independent of $\mu$ and $w$, which is correct for deciding the almost-sure model-checking problem in a number of classes of stochastic timed automata.

## 4. A topological semantics

In [VV06] almost-sure model-checking of concurrent reactive systems is characterised by a topological notion: largeness is qualitative and captures the notion of "many runs". Inspired by that work, we propose a topological semantics for timed automata, based on the notion of large sets, which will help us characterise almost-sure sets. In our context also, the topological semantics is purely qualitative but nevertheless gives information on "how big" a set of paths satisfying a given property is.

In this section, relying on a notion of thickness for symbolic paths, we first define a natural topology over infinite runs of a given timed automaton. This topology induces a

*large* semantics: an $\omega$-regular property is satisfied if "most of the runs" satisfy it. As pointed out already in [VV06], *largeness*, and its complement *meagerness*, are better appropriate than density to express a notion such as "most of the runs". Indeed, "small" sets can be dense, and the complement of a dense set can also be dense (*e.g.* $\mathbb{Q}$ in $\mathbb{R}$), whereas it is not the case for large sets. Let us start this section by recalling the notion of large sets and their characterisation using Banach-Mazur games.

4.1. **Largeness and the Banach-Mazur game.** We refer to [Mun00] for basic notions of topology (topological space, interior, closure, *etc*). However we recall here the more specific notion of *largeness* and also provide its elegant characterisation in terms of *Banach-Mazur games* [Oxt57].

4.1.1. *Some topological notions.* Let $(A, \mathcal{T})$ be a topological space. If $B \subseteq A$, we denote by $\mathring{B}$ (resp. $\overline{B}$) the *interior* (resp. *closure*) of $B$. Let us recall that a set $\mathcal{T}' \subseteq \mathcal{T}$ is called a *basis* for the topology $\mathcal{T}$ if every open set (*i.e.*, elements of $\mathcal{T}$) can be obtained as the union of elements of $\mathcal{T}'$. In this case, the elements of $\mathcal{T}'$ are called *basic opens*. A set $B \subseteq A$ is *nowhere dense* if the interior of the closure of $B$ is empty, *i.e.*, $\mathring{\overline{B}} = \emptyset$. A set is *meagre* if it is a countable union of nowhere dense sets. Finally, a set is *large* if its complement is meagre.

**Example 4.1.** Let $\mathbb{R}$ be the set of real numbers equipped with its natural topology (that is, basic open sets are the open intervals). The set of integers $\mathbb{Z}$ is nowhere dense in $\mathbb{R}$. The set of rational numbers $\mathbb{Q}$ is dense (in $\mathbb{R}$) however $\mathbb{Q}$ is meagre since is a countable union of singletons (which are clearly nowhere dense sets); this implies that $\mathbb{R} \setminus \mathbb{Q}$ is large. ⌟

4.1.2. *Banach-Mazur game.* Although the notion of largeness is quite abstract, it admits a very nice characterisation in terms of a two-player game, known as *Banach-Mazur game*.

**Definition 4.2** (Banach-Mazur game)**.** Let $(A, \mathcal{T})$ be a topological space and $\mathcal{B}$ be a family of subsets of $A$ satisfying the two following properties:

- for all $B \in \mathcal{B}$, $\mathring{B} \neq \emptyset$, and
- for all $O$ a non-empty open set of $A$, there exists $B \in \mathcal{B}$ such that $B \subseteq O$.

Fix $C$ a subset of $A$. Two players alternate their moves: Player 1 starts and chooses an element $B_1$ of $\mathcal{B}$; Player 2 then responds by choosing an element $B_2$ of $\mathcal{B}$ such that $B_1 \supseteq B_2$; Then Player 1 chooses $B_3$ in $\mathcal{B}$ such that $B_2 \supseteq B_3$, and so on. This way, they define a non increasing sequence of sets $B_i$:

$$A \supseteq B_1 \supseteq B_2 \supseteq B_3 \cdots$$

where the $B_{2i+1}$'s (resp. $B_{2i}$'s) are chosen by Player 1 (resp. Player 2), for $i \in \mathbb{N}$. Player 1 wins the game if the intersection of all $B_i$'s intersects $C$, *i.e.*,

$$\bigcap_{i=1}^{\infty} B_i \cap C \neq \emptyset\,.$$

Otherwise, Player 2 wins the game.

Notice that typical examples of family $\mathcal{B}$ are provided by topology bases.

Banach-Mazur games are not always determined, even for simple topological spaces (see [Oxt57, Remark 1]). Still a natural question is to know when the players have winning strategies. The following result gives a partial answer:

**Theorem 4.3** (Banach-Mazur [Oxt57])**.** Player 2 has a winning strategy in the Banach-Mazur game with target set $C$ if and only if $C$ is meagre.

To illustrate Theorem 4.3 we give the following simple example.

**Example 4.4.** Let $(\mathbb{R}, \mathcal{T})$ be the set of real numbers equipped with the natural topology, $\mathcal{B}$ be the family of open intervals with rational bounds, and $C$ be the open set $(0, 1)$. Intuitively, $(0, 1)$ is not meagre, and to prove it using Theorem 4.3, it is sufficient to show that Player 2 does not have a winning strategy in the Banach-Mazur game on $\mathcal{B}$ with target set $C$.

Assume Player 1's first move is to pick the set $C$ for $B_1$. From then on, the game takes place within $C$, and the only way for Player 2 to win is to build a sequence of $B_i$'s converging to the empty set. Let us now explain how Player 1 can prevent this from happening. Let $B_{2i} = (a_i, b_i)$ be the $i$th set chosen by Player 2. Player 1's response to $(a_i, b_i)$ is the interval $B_{2i+1} = (a_i + \epsilon_i, b_i - \epsilon_i)$, where $\epsilon_i = \frac{b_i - a_i}{3}$. Notice that $B_{2i+1}$ is a regular move for Player 1 since $B_{2i} \supseteq B_{2i+1}$ and $B_{2i+1} \in \mathcal{B}$. Notice also that the closed set $F_i = [a_i + \frac{\epsilon_i}{2}, b_i - \frac{\epsilon_i}{2}]$ satisfies the following inclusions: $B_{2i} \supseteq F_i \supseteq B_{2i+1}$. Given any strategy of Player 2, using the above strategy Player 1 ensures:

$$\bigcap_{i=1}^{\infty} B_i \cap C \;=\; \bigcap_{i=1}^{\infty} F_i.$$

This intersection is guaranteed to be non-empty by Heine-Borel theorem, since the sequence is included in $[0, 1]$ which is a compact set. As a consequence, Player 2 does not have a winning strategy, and thus $C$ is non meagre. ⌟

**Remark 4.5.** Surprisingly, in general, there is no relation between meagre and open sets. Indeed non meagre open sets obviously exist, but one can identify topological spaces where open sets are meagre. Let us consider the set of rational numbers $\mathbb{Q}$ with its natural topology (whose basic open sets are the open intervals). In this topology all sets, and in particular all open sets, are countable and thus meagre.

A topological space in which all non-empty open sets are not meagre is called a *Baire space*.[7] As we have just noticed, not all topological spaces are Baire spaces. Indeed, under their natural topologies, $\mathbb{Q}$ is not a Baire space whereas $\mathbb{R}$ is (a proof follows the same ideas than in Example 4.4). This remark suggests that largeness and meagerness are even more relevant in Baire spaces.

4.2. **Thick and thin symbolic paths.** We fix a timed automaton $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ for the rest of the section. In order to attach a topology to sets of infinite runs in $\mathcal{A}$, we first define thick symbolic paths. In $\mathbb{R}^n$, open sets are among those sets of maximal dimension. Symbolic paths do not exactly lie in $\mathbb{R}^n$, but we will see that each symbolic constrained path of length $n$ can be embedded in some ambient space $\mathbb{R}^m$, with $m \leq n$. *Thick* symbolic paths will then naturally arise as symbolic paths of maximal dimension in their ambient space.

[7] In modern definitions, a topological space is a Baire space if each countable union of closed sets with an empty interior has an empty interior. However, the two definitions coincide, see [Mun00, p.295].

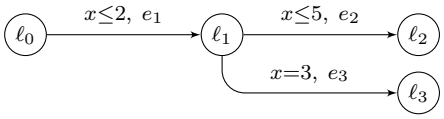



Figure 4: Thick and thin symbolic paths on an example.

Before going to the definition, let us explain through an example the intuition behind this notion.

**Example 4.6.** Consider the single-clock timed automaton depicted in Figure 4, $s_0 = (\ell_0, 0)$ and $\pi$ be the (unconstrained) symbolic path $\pi(s_0, e_1e_2)$. One can naturally associate a polyhedron of $(\mathbb{R}_+)^2$ with $\pi$:

$$\begin{aligned}\mathsf{Pol}(\pi) &= \{(\tau_1,\tau_2) \in (\mathbb{R}_+)^2 \mid \varrho = s_0 \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} s_2\}\\ &= \{(\tau_1,\tau_2) \in (\mathbb{R}_+)^2 \mid (0 \leq \tau_1 \leq 2) \wedge (0 \leq \tau_1+\tau_2 \leq 5)\}\end{aligned}$$

$\mathsf{Pol}(\pi)$ has dimension 2 in $\mathbb{R}^2$. Since it is of maximal dimension, we say that the symbolic path $\pi$ is *thick*. Consider now the symbolic path $\pi' = \pi(s_0, e_1e_3)$. The polyhedron $\mathsf{Pol}(\pi')$ associated with $\pi'$ has dimension 1, and is somehow embedded in a two-dimensional space (due to the existence of the edge $e_2$). In that case, we say that it is *thin*. ⌟

The above example is simplistic and could give the wrong impression that symbolic paths with singular transitions (*i.e.*. transitions that do not increase the dimension of the polyhedron) are necessarily thin; or equivalently that in order to be thick, a symbolic path of length $n$ should have an associated polyhedron of dimension $n$. This is not always the case, and singular transitions can play an important role. Consider a slight modification of the automaton of Figure 4 where edge $e_1$ is guarded by $x = 2$. In this modified automaton, $\pi(s_0, e_1e_2)$ is still thick, $\pi(s_0, e_1e_3)$ is thick too and the dimension of the ambient space of any symbolic path of length 2 is 1.

To formally define thick and thin paths, we introduce the notion of associated polyhedron, and some notations. Given $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(s, e_1 \dots e_n)$ a constrained path of a timed automaton $\mathcal{A}$, its *associated polyhedron* is defined as follows:

$$\mathsf{Pol}(\pi_{\mathcal{C}}) = \{(\tau_i)_{1\leq i\leq n} \in (\mathbb{R}_+)^n \mid s \xrightarrow{\tau_1,e_1} s_1 \cdots \xrightarrow{\tau_n,e_n} s_n \in \pi_{\mathcal{C}}(s, e_1 \dots e_n)\}\,.$$

Moreover, for each $0 < i \leq n$, we write $\mathcal{C}_i$ for the constraint induced by the projection of $\mathsf{Pol}(\pi_{\mathcal{C}})$ over the variables corresponding to the $i$ first coordinates (with the convention that $\mathcal{C}_0$ is true).

**Definition 4.7.** The constrained path $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(s, e_1 \dots e_n)$ is *thin* whenever there exists some index $1 \leq i \leq n$ such that

$$\dim\Big(\mathsf{Pol}\big(\pi_{\mathcal{C}_i}(s, e_1 \dots e_{i-1}e_i)\big)\Big) < \dim\Big(\bigcup_e \mathsf{Pol}\big(\pi_{\mathcal{C}_{i-1}}(s, e_1 \dots e_{i-1}e)\big)\Big)\,.$$

Otherwise $\pi_{\mathcal{C}}$ is *thick*.

Clearly enough all extensions of thin symbolic paths are thin as well. Let us examine an example illustrating some subtlety of this notion.

**Example 4.8.** Consider the timed automaton depicted in Figure 5 where $e_i$ denotes the transition ending in $\ell_i$, for $i = 1, 2, 3$. Consider the (unconstrained) symbolic paths $\pi(s_0, e_1e_2)$ and $\pi(s_0, e_1e_3)$, where $s_0 = (\ell_0, 0)$, and let us argue that $\pi(s_0, e_1e_2)$ is thin,

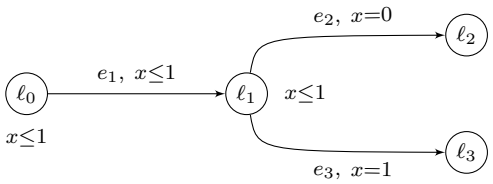



Figure 5: A subtle example illustrating thickness and thinness.

whereas $\pi(s_0, e_1e_3)$ is thick. Intuitively, this difference comes from the fact that "reaching $\ell_2$" is only possible when transition $e_1$ has been taken exactly when the value of the clock $x$ is 0, although "reaching $\ell_3$" is always possible after transition $e_1$ has been taken. Formally: $\mathsf{Pol}(\pi(s_0, e_1e_2)) = \{(0,0)\}$ whereas $\mathsf{Pol}(\pi(s_0, e_1e_3)) = \{(\tau_1, \tau_2) \in [0,1]^2 \mid \tau_1 + \tau_2 = 1\}$. Hence,

$$\dim\Big(\mathsf{Pol}\big(\pi(s_0, e_1e_2)\big)\Big) < \dim\Big(\mathsf{Pol}\big(\pi(s_0, e_1e_3)\big)\Big) = \dim\Big(\bigcup_e \mathsf{Pol}\big(\pi(s_0, e_1e)\big)\Big)$$

proving the desired result. ⌟

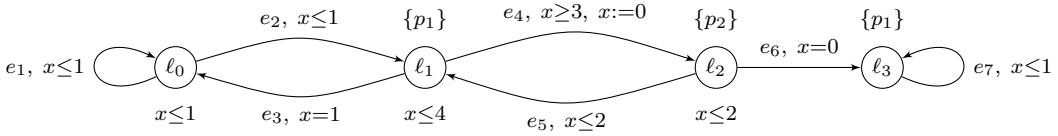
The latter example shows that thickness cannot be tested locally: edges $e_2$ and $e_3$ are both guarded by equality constraints, but do not behave the same with respect to thickness. This phenomenon cannot happen in timed region automata, in which, as we shall establish later (see Proposition 5.4, page 21), thickness coincides with local thickness.

The notion of thickness naturally extends to infinite symbolic paths.

**Definition 4.9.** An infinite symbolic path $\pi_{\mathcal{C}}(s, e_1e_2\ldots)$ is *thick* if for all $n \geq 1$, $\pi_{\mathcal{C}}(s, e_1 \ldots e_n)$ is thick. Otherwise, it is *thin*.

We illustrate these notions on our running example.

**Example 4.10.** On $\mathcal{A}_{\mathsf{running}}$ of Figure 3, also reproduced below, with $s_0 = (\ell_0, 0)$, let us explain why $\pi(s_0, {e_1}^\omega)$ is thick and $\pi(s_0, e_2e_3e_1^\omega)$ is thin. First observe that all finite prefixes




$\pi(s_0, e_1^n)$ of $\pi(s_0, {e_1}^\omega)$ are thick. Indeed,

$$\mathsf{Pol}(\pi(s_0, {e_1}^n)) = \{(\tau_1, \ldots, \tau_n) \in (\mathbb{R}_+)^n \mid (0 \leq \tau_1 \leq 1) \wedge \cdots \wedge (0 \leq \tau_1 + \cdots + \tau_n \leq 1)\}\ .$$

Thus, clearly enough $\dim(\mathsf{Pol}(\pi(s_0, {e_1}^n))) = n$, which is maximal for an $n$-dimension polyhedron. This proves that $\pi(s_0, {e_1}^n)$ is thick, for all $n \in \mathbb{N}$, and thus $\pi(s_0, {e_1}^\omega)$ is thick too.

Consider now the infinite path $\pi(s_0, e_2e_3e_1^\omega)$ and show that it is thin by exhibiting a thin finite prefix. Observe that:

$$\mathsf{Pol}(\pi(s_0, e_2e_3)) = \{(\tau_1, \tau_2) \mid (0 \leq \tau_1 \leq 1) \wedge (0 \leq \tau_1 + \tau_2 = 1)\}\,,$$
$$\mathsf{Pol}(\pi(s_0, e_2e_4)) = \{(\tau_1, \tau_2) \mid (0 \leq \tau_1 \leq 1) \wedge (3 \leq \tau_1 + \tau_2)\}\,,$$

thus $\dim(\mathsf{Pol}(\pi(s_0, e_2e_3))) = 1 < \dim(\mathsf{Pol}(\pi(s_0, e_2e_4))) = 2$ which implies that $\pi(s_0, e_2e_3)$ is thin. Hence we conclude that $\pi(s_0, e_2e_3e_1^\omega)$ is thin. ⌟

4.3. **A topology on infinite runs.** The goal of this subsection is to define a topology on the set of infinite runs of a given timed automaton. Keeping our analogy with $\mathbb{R}^n$, where the open sets are among the sets of maximal dimension, we use the notion of thickness introduced in the latter subsection in order to define a topology on runs. More precisely, the basic open sets will be cylinders over thick constrained symbolic paths whose associated polyhedra are open in their ambient spaces.

**Definition 4.11.** Let $s$ be a state of $\mathcal{A}$. Let $\mathcal{T}^s_{\mathcal{A}}$ be the topology over the set of runs of $\mathcal{A}$ starting in $s$ defined with the following basic open sets:[8] either the set $\mathsf{Runs}(\mathcal{A}, s)$, or the empty set $\emptyset$, or the cylinders $\mathsf{Cyl}(\pi_{\mathcal{C}})$ where $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(s, e_1e_2 \dots e_n)$ is a finite constrained symbolic path of $\mathcal{A}$ such that: (i) $\pi_{\mathcal{C}}$ is thick, (ii) $\mathcal{C}$ is Borel-measurable, and (iii) $\mathsf{Pol}(\pi_{\mathcal{C}})$ is open in $\mathsf{Pol}(\pi)$ for the classical topology on $\mathbb{R}^n$, where $\pi = \pi(s, e_1e_2 \dots e_n)$.

Before illustrating our definition, let us draw the reader's attention on the two following points. First notice that Definition 4.11 only makes sense if the intersection of two basic open sets is still a basic open set; This is proven in the Appendix as Lemma B.1 (page 45). Second, regarding our initial objective of expressing a notion of "most of the runs" using *largeness*, we need, for consistency, the space to be Baire (see Remark 4.5); This is stated below and proven in Appendix B, page 46.

**Proposition 4.12.** For every state $s$ of $\mathcal{A}$, the topological space $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$ is a Baire space.

Now that the validity of our topology is clear, we illustrate it on our running example.

**Example 4.13.** On the running automaton $\mathcal{A}_{\mathsf{running}}$ of Figure 3 with initial state $s_0 = (\ell_0, 0)$, the set $C \stackrel{\text{def}}{=} \bigcup_{i\in\mathbb{N}} \pi(s_0, {e_1}^i e_2(e_4e_5)^\omega)$ is large. To prove it (or equivalently to prove that the complement of $C$ is meagre) we use a Banach-Mazur game, and show that Player 2 has a strategy to avoid the complement of $C$, that is to reach $C$. The game is played with the basic open sets of $\mathsf{Runs}(\mathcal{A}_{\mathsf{running}}, s_0)$. A winning strategy for Player 2 goes as follows:

- Assume Player 1 has chosen a cylinder $\mathsf{Cyl}(\pi(s_0, {e_1}^{n_1}))$, for some $n_1 \in \mathbb{N}_0$ (if Player 1 leaves $\ell_0$ at her first move, we skip the first move of Player 2)
- Player 2 chooses $\mathsf{Cyl}(\pi(s_0, {e_1}^{n_1}e_2))$,
- Notice that Player 1 is not allowed to extend the symbolic path $\pi(s_0, {e_1}^{n_1}e_2)$ with sequences of transitions including $e_3$ or $e_6$, since both symbolic paths $\pi(s_0, {e_1}^{n_1}e_2e_3)$ and $\pi(s_0, {e_1}^{n_1}e_2e_4e_6)$ are thin. Therefore Player 1 can only play moves of the form $\mathsf{Cyl}(\pi(s_0, {e_1}^{n_1}e_2(e_4e_5)^{n_2}))$ or $\mathsf{Cyl}(\pi(s_0, {e_1}^{n_1}e_2(e_4e_5)^{n_2}e_4))$.
- Player 2 then responds $\mathsf{Cyl}(\pi(s_0, {e_1}^{n_1}e_2(e_4e_5)^{n_3}))$, with $n_3 > n_2$.

One can easily be convinced that repeating infinitely often the last two moves, the play forms a run of $C$, proving that Player 2 wins the game and thus that $C$ is large.

Notice that both players could also play with constrained paths. This would not be interesting for Player 1, since it may cause the intersection to be empty (in which case Player 2 wins as well). ⌟

[8] We recall that open sets of $\mathcal{T}^s_{\mathcal{A}}$ are then built from those basic open sets using union.

We now give a simple characterisation of the basic open sets considered in Definition 4.11, whose proof is given in Appendix B, page 48.

**Lemma 4.14.** In the topological space $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$, a finite symbolic path $\pi$ defines a basic open set if and only if there exist an open constraint $\mathcal{C}$ of $\mathbb{R}^n$ and thick edges $e_1, \dots, e_n$ such that $\pi = \pi_{\mathcal{C}}(s, e_1 \dots e_n)$.

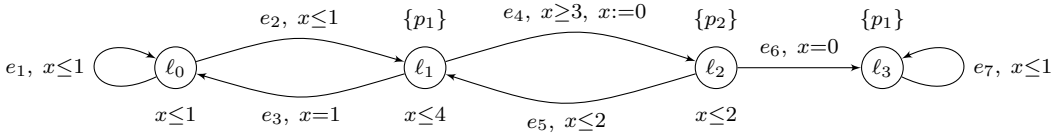
4.4. **Large satisfaction.** We are now in a position where we can define a notion of large satisfaction.

**Definition 4.15.** Let $s$ be a state of $\mathcal{A}$, and $P$ be a timed property over $\mathsf{AP}$. We say that $\mathcal{A}$ *largely satisfies* $P$ from $s$, and write $\mathcal{A}, s \mathrel{|\!\approx_{\mathcal{T}}} P$, if the set $\{\varrho \in \mathsf{Runs}(\mathcal{A}, s) \mid \varrho \models P\}$ is topologically large (in $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$).

Let us illustrate this notion of large satisfaction on our running example.

**Example 4.16.** On the running example $\mathcal{A}_{\mathsf{running}}$ depicted below, with $s_0 = (\ell_0, 0)$,

$$\mathcal{A}_{\mathsf{running}}, s_0 \mathrel{|\!\approx_{\mathcal{T}}} \mathbf{F}\,(p_1 \wedge \mathbf{G}\,(p_1 \Rightarrow \mathbf{F}\,p_2))\,.$$




Indeed, in Example 4.13, we proved that the set $C \stackrel{\text{def}}{=} \{\pi(s_0, {e_1}^i e_2 (e_4 e_5)^\omega) \mid i \in \mathbb{N}\}$ is large. Moreover each run of $C$ satisfies $\varphi \stackrel{\text{def}}{=} \mathbf{F}\,(p_1 \wedge \mathbf{G}\,(p_1 \Rightarrow \mathbf{F}\,p_2))$, and thus $\mathcal{A}_{\mathsf{running}}, s_0 \mathrel{|\!\approx_{\mathcal{T}}} \varphi$, since largeness is closed under subsumption. Note that the previous formula is not satisfied with the classical $\mathsf{LTL}$ semantics. '*Staying in* $\ell_0$ *forever*', '*reaching* $\ell_3$', etc are examples of behaviours in $\mathcal{A}_{\mathsf{running}}$ that violate the $\mathsf{LTL}$ formula $\varphi$. ⌟

4.5. **From timed automata to timed region automata.** As in the context of probabilities, we can relate the topologies in $\mathcal{A}$ and in $\mathsf{R}(\mathcal{A})$. Although the topological spaces given by $\mathcal{A}$ and $\mathsf{R}(\mathcal{A})$ are not homeomorphic, the topologies in $\mathcal{A}$ and in $\mathsf{R}(\mathcal{A})$ are somehow equivalent, as stated by the next lemma. This will allow us to lift result from $\mathsf{R}(\mathcal{A})$ to $\mathcal{A}$.

**Lemma 4.17.** Let $\iota : \mathsf{Runs}(\mathcal{A}, s) \to \mathsf{Runs}(\mathsf{R}(\mathcal{A}), \iota(s))$ be the projection of runs in $\mathcal{A}$ onto the region automaton $\mathsf{R}(\mathcal{A})$. Then $\iota$ is continuous, and for every non-empty open set $O \in \mathcal{T}^s_{\mathcal{A}}$, $\widehat{\iota(O)^\circ} \neq \emptyset$.

The proof of this lemma is given in Appendix B, page 48.

**Remark 4.18.** Note that $\iota : \mathsf{Runs}(\mathcal{A}, s) \to \mathsf{Runs}(\mathsf{R}(\mathcal{A}), \iota(s))$ is not an homeomorphism since $\iota^{-1} : \mathsf{Runs}(\mathsf{R}(\mathcal{A}), s) \to \mathsf{Runs}(\mathcal{A}, \iota^{-1}(s))$ is not continuous. Indeed, let us consider the automaton $\mathcal{A}$ of Figure 2, page 8, with $s_0 = (\ell_0, 0)$. The set of runs $O = \mathsf{Cyl}(\pi(s_0, e_1))$ is open in $\mathcal{T}^{s_0}_{\mathcal{A}}$ since $\pi(s_0, e_1)$ is a thick symbolic path. However, $\iota(\mathsf{Cyl}(\pi(s_0, e_1))) = \mathsf{Cyl}(\pi(s_0, f_1)) \cup \mathsf{Cyl}(\pi(s_0, f_2))$ is not open in $\mathcal{T}^{\iota(s_0)}_{\mathsf{R}(\mathcal{A})}$ as $\pi(s_0, f_1)$ is thin and hence $\mathsf{Cyl}(\pi(s_0, f_1))$ is not a basic open. Thus $\iota(O)$ is not open and $\iota^{-1}$ is not continuous.

Lemma 4.17 allows one to simulate a Banach-Mazur game from $\mathcal{A}$ to $\mathsf{R}(\mathcal{A})$ and *vice-versa*. Therefore, the large satisfaction relations in $\mathcal{A}$ and $\mathsf{R}(\mathcal{A})$ coincide, see Appendix B, page 50.

**Proposition 4.19.** Let $s$ be a state of $\mathcal{A}$, and $P$ be a timed property over $\mathsf{AP}$. Then,

$$\mathcal{A}, s \not\approx_{\mathcal{T}} P \;\Leftrightarrow\; \mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathcal{T}} P\,.$$

## 5. Construction of the thick graph

In this section, we construct the so-called thick graph. The idea is to remove *locally thin* edges from the region automaton, and it will be used to characterise (globally) thin paths.

We fix a timed automaton $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$, and we start by defining a local notion of thinness.

**Definition 5.1.** Let $e$ be an edge of $\mathsf{R}(\mathcal{A})$, and $q$ its source. We say $e$ is *thin* whenever $\dim(I(s,e)) < \dim(I(s))$ for some (or equivalently, for every) $s \in q$. Otherwise the edge is said *thick*.

An equivalent definition is that an edge $e$ with source $q$ is thin whenever for every $s \in q$, the length-1 constrained path $\pi(s,e)$ is thin. That is why this notion of thinness is local. Next we will write $\dim(I(q,e))$ (resp. $\dim(I(q))$) instead of $\dim(I(s,e))$ (resp. $\dim(I(s))$) for every $s \in q$ since this is independent on $s \in q$.

**Definition 5.2.** The *thick graph* of $\mathcal{A}$, denoted $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$, is obtained from $\mathsf{R}(\mathcal{A})$ by deleting all the thin edges.

In particular, $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ has only thick edges. We first state a lemma, which explains how the dimension of symbolic paths grows in the (timed) region automaton. Its proof can be found in Appendix C, page 50

**Lemma 5.3.** Let $\pi(s, e_1 \dots e_n)$ be a symbolic path of $\mathsf{R}(\mathcal{A})$. Assuming $\mathsf{Pol}(\pi(s, e_1 \dots e_n)) \neq \emptyset$ and letting $q$ be the target region of $\pi(s, e_1 \dots e_{n-1})$,

$$\dim(\mathsf{Pol}(\pi(s, e_1 \dots e_n))) = \dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1}))) + \dim(I(q, e_n)).$$

That is, we are able to compute the global dimension of a symbolic path, given the dimension of each of its edges. Notice that this is only true in the region automaton. As an example, in the timed automaton of Example 4.8, $\mathsf{Pol}(\pi(s, e_1 e_2)) = \{(0,0)\}$, whereas $\mathsf{Pol}(\pi(s, e_1)) = [0,1]$.

The following proposition states the correctness of the thick graph, in the sense that a symbolic path is (globally) thin if and only if it traverses a (locally) thin edge. Its proof is given in Appendix C, page 51.

**Proposition 5.4.** Let $\pi = \pi(s, e_1 \dots e_n)$ be a symbolic path in $\mathsf{R}(\mathcal{A})$. Then, $\pi$ is thin in $\mathsf{R}(\mathcal{A})$ iff there exists $1 \leq i \leq n$ such that $e_i$ is thin.

The thick graph will be used in the next section for characterising large and almost-sure satisfaction. It will later be used for algorithmic issues.

**Example 5.5.** We illustrate the construction of the thick graph on the running example. Figure 6 represents the classical region graph where thick (resp. thin) edges are depicted bold (resp. dashed). From that region graph, by removing thin edges and keeping only the states reachable from the initial one, we obtain the thick graph, represented on Figure 7. ⌟

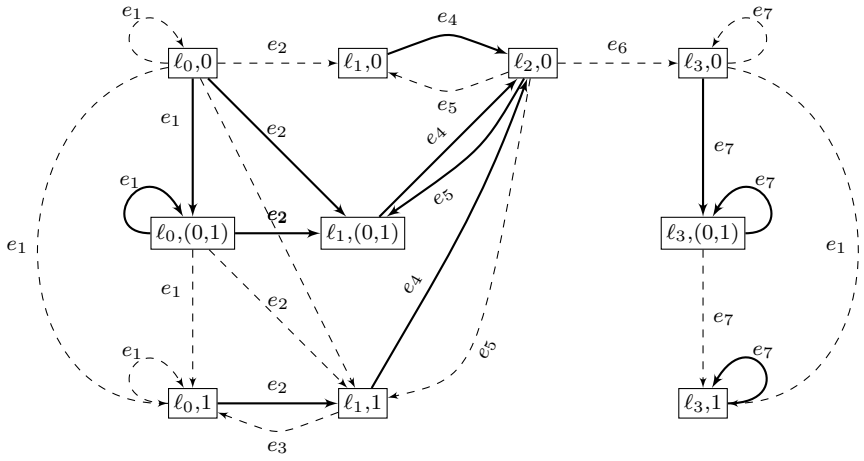



Figure 6: Construction of the thick graph on the running example.

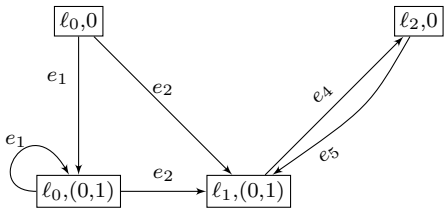



Figure 7: $\mathcal{G}_{\mathsf{t}}(\mathcal{A}_{\mathsf{running}})$, the thick graph for the running example.

**Remark 5.6.** In the following, we will denote by $\mathsf{MC}(\mathcal{A})$ the finite Markov chain obtained by taking $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ as the support of the Markov chain, and assuming uniform distributions over edges. In the next sections, we write $\mathsf{MC}(\mathcal{A}), s \not\approx P$ whenever the finite Markov chain $\mathsf{MC}(\mathcal{A})$ almost-surely satisfies property $P$ from state $s$.

## 6. When do the large and the almost-sure satisfaction coincide?

We know that large satisfaction and almost-sure satisfaction coincide for finite automata for several classes of properties [VV06]. We want here to discuss situations where almost-sure and large satisfactions also match in our context. This will help giving algorithmic solutions to the almost-sure model-checking problem using the thick graph.

We fix for this section a stochastic timed automaton $\langle \mathcal{A}, \mu, w \rangle$.

### 6.1. **Safety properties.**

We first compare the two semantics in the restricted case of safety properties. Let us first state this simple result that, in a region automaton, a finite symbolic path has probability 0 iff it is thin. This is the first easy link we can make between the probabilities and the topology. Note that this correspondence does not depend on the choice of the probability distributions.

**Proposition 6.1.** Consider a finite symbolic path $\pi = \pi(s, e_1 \dots e_n)$ in $\mathsf{R}(\mathcal{A})$. Then, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi)) > 0$ iff $\pi$ is thick. Equivalently, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi)) = 0$ iff $\pi$ is thin.

This proposition relies on condition $(\star)$ on the probability distributions $\mu$ (*cf* page 11): if an edge is thick, it is either because it has dimension 1, or because it has dimension 0, but all other outgoing edges also have dimension 0. In the first case, the measure $\mu$ must be equivalent to the Lebesgue measure, and in the second case, it will give a positive mass to the edge. The full details of the proof are given in Appendix D.1, page 52.

Using this result, we show that the large and the almost-sure satisfaction always coincide when we restrict to safety properties.

**Theorem 6.2.** Let $s$ be a state of $\mathcal{A}$, and $P$ be a(n untimed) safety property over $\mathsf{AP}$. Then the four following properties are equivalent:

(a) $\mathcal{A}, s \models\!\approx_{\mathbb{P}} P$;
(b) $\mathcal{A}, s \models\!\approx_{\mathcal{T}} P$;
(c) every infinite thick symbolic path $\pi$ from $\iota(s)$ in $\mathsf{R}(\mathcal{A})$ satisfies $P$;
(d) every infinite path $\pi$ from $\iota(s)$ in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ satisfies $P$;
(e) $\mathsf{MC}(\mathcal{A}), \iota(s) \models\!\approx P$.

This result relies on the fact that safety properties are violated by finite prefixes. Proposition 6.1 then tells us that such finite prefixes yield positive probability whenever they are thick. We can then play a Banach-Mazur game in the topological space of the automaton, where only thick paths can be used by the players as moves. This allows to show that property $\neg P$ is meagre iff $P$ is not violated by thick prefixes, that is when $\neg P$ has probability 0. The details of the proof are given in Appendix D.1, on page 52.

As said, the proof of Theorem 6.2 heavily relies on the fact that witnesses of violation (resp. validation) for safety (resp. reachability) properties are finite prefixes. Not surprisingly, Theorem 6.2 does not hold for general $\mathsf{LTL}$ or $\omega$-regular properties, for which the violation cannot always be witnessed by finite prefixes. As an example, consider the timed automaton of Figure 8, and the property $\mathbf{F}\, p$. The probability of $\mathbf{F}\, p$ is indeed 1 in this automaton, although the infinite symbolic path $\pi((\ell_0, 0), e_0^\omega)$ violates $\mathbf{F}\, p$ and is thick.

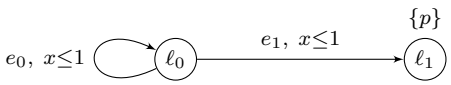



Figure 8: A counterexample for Theorem 6.2 beyond safety.

This kind of behaviours motivates the restriction to fair paths, which is rather natural since probabilities and strong fairness are closely related [Pnu83, PZ93, BK98].

6.2. **Restriction to fairness: the case of prefix-independent properties.** Motivated by the counterexample of Figure 8, we define a natural notion of fairness for infinite symbolic paths in timed automata.

**Definition 6.3.** An infinite region path $\pi = q_0 \xrightarrow{e_1} q_1 \xrightarrow{e_2} q_2 \dots$ of $\mathsf{R}(\mathcal{A})$ is *fair* iff for every thick edge $e$, if $e$ is enabled in infinitely many $q_i$ ($i \in \mathbb{N}$), then $e_i = e$ for infinitely many $i \in \mathbb{N}$.

Fairness extends to runs and symbolic paths in an obvious way as detailed below. Region paths and symbolic paths in $\mathsf{R}(\mathcal{A})$ are closely related: to any non-empty symbolic path $\pi(s, e_1e_2\dots)$, we associate a unique region path $q_0 \xrightarrow{e_1} q_1 \xrightarrow{e_2} q_2\dots$ with $s \in q_0$ and $q_i$ is the target region of edge $e_i$. We then say that a symbolic path $\pi(s, e_1e_2\dots)$ in $\mathsf{R}(\mathcal{A})$ is fair whenever its corresponding region path is fair. Finally we say that an infinite run $\varrho$ in $\mathsf{R}(\mathcal{A})$ is fair whenever $\pi_\varrho$ (its underlying symbolic path) is fair. Obviously, the set of fair infinite runs from $s$ is measurable (that is, in $\Omega^s_{\mathsf{R}(\mathcal{A})}$), as fairness is an $\omega$-regular property over infinite paths. We now turn the definition from $\mathsf{R}(\mathcal{A})$ to $\mathcal{A}$: an infinite run $\varrho$ in $\mathcal{A}$ is fair whenever $\iota(\varrho)$ is fair.

We write $\mathsf{fair}$ for this property, that is if an infinite run $\varrho$ (in $\mathcal{A}$ or in $\mathsf{R}(\mathcal{A})$) is fair, then we write $\varrho \models \mathsf{fair}$. We then write $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair})$ (resp. $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{fair})$) for the probability of fair runs in $\mathcal{A}$ from $s$ (resp. in $\mathsf{R}(\mathcal{A})$ from $\iota(s)$). We say that $\mathcal{A}$ (resp. $\mathsf{R}(\mathcal{A})$) is *almost-surely fair* from $s$ (resp. $\iota(s)$) whenever $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$ (resp. $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{fair}) = 1$).

Let us state the following straightforward lemma, which gives a useful characterisation of thick and fair symbolic paths in the region automaton.

**Lemma 6.4.** Let $s$ be a state of $\mathcal{A}$, and $P$ be a prefix-independent property over $\mathsf{AP}$. Then, the two following properties are equivalent:

(i) all BSCCs[9] reachable from $[s]$ in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ satisfy $P$;[10]
(ii) every infinite thick **and fair** symbolic path $\pi$ from $\iota(s)$ in $\mathsf{R}(\mathcal{A})$ satisfies $P$;
(iii) $\mathsf{MC}(\mathcal{A}), \iota(s) \not\approx P$.

We write $\S(\mathcal{A}, s, P)$ for this property.

6.2.1. *Fairness and topology.* Even though fairness is introduced because of the probabilities, we first realise that adding fairness to paths allows to characterise the large satisfaction. More precisely, we prove the following result, which is rather similar to Theorem 6.2, when restricted to topology.

**Theorem 6.5.** Let $s$ be a state of $\mathcal{A}$, and $P$ be a prefix-independent property over $\mathsf{AP}$. Then:

$$\mathcal{A}, s \not\approx_{\mathcal{T}} P \;\Leftrightarrow\; \S(\mathcal{A}, s, P)\,.$$

*Proof.* First assume that $\mathcal{A}, s \not\approx_{\mathcal{T}} P$. This equivalently means that $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathcal{T}} P$ (Lemma 4.19), that is, $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$ is meagre. We now apply the characterisation of meagre sets *via* Banach-Mazur games (Theorem 4.3), where the players play with basic open sets of $(\mathsf{Runs}(\mathsf{R}(\mathcal{A}), \iota(s)), \mathcal{T}^{\iota(s)}_{\mathsf{R}(\mathcal{A})})$. Note that for every basic open set $\pi_{\mathcal{C}}(s, e_1 \dots e_n)$ in the above topology, all edges $e_1, \dots, e_n$ are thick, the open set can therefore "be read" in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$.

In this game, Player 2 has a strategy to ensure that $\bigcap_i B_i \cap [\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)} = \emptyset$, where the $B_i$'s are the moves in the game. Let us denote $\Sigma_2$ this winning strategy. Fix any BSCC $C$ of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ reachable from $[s]$, and let $\{\ell_1, \dots, \ell_p\}$ be an enumeration of the locations in $C$. In order to prove that $C$ satisfies $P$, we will build a symbolic path $\pi$, played according to $\Sigma_2$ (and thus satisfying $P$), witnessing that $C$ satisfies $P$. Let Player 1 play as follows (against $\Sigma_2$): at her first move, Player 1 chooses $B_1$ leading to $\ell_1$; then, no matter which

[9] BSCC stands for 'bottom strongly connected component', see *e.g.* [CLRS09].

[10] We say that a prefix-independent property $P$ is satisfied by a BSCC $C$ whenever every run visiting infinitely often all states of $C$ satisfies $P$. As $P$ is prefix-independent, this reduces to the existence of a run satisfying $P$, which visits infinitely often all states of $C$.

$B_2$ Player 2 chooses, Player 1 chooses $B_3$ (longer than $B_2$) leading to $\ell_2$; *etc.* Furthermore applying a technique similar to the proof of Proposition 4.12 (compactification), Player 1 can ensure that $\bigcap_i B_i \neq \emptyset$. We then get that $\emptyset \neq \bigcap_i B_i \subseteq [\![P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$, since $\Sigma_2$ is winning for Player 2. Let $\pi$ be the infinite symbolic path underlying $\bigcap_i B_i$. As $P$ is a property over $\mathsf{AP}$, $\pi \models P$. Furthermore, $\pi$ visits all locations of $C$ (since Player 1 has ensured visiting all the locations of $C$ infinitely often), which means that the BSCC $C$ satisfies $P$. We conclude that all BSCCs satisfy property $P$.

Conversely, assume that every BSCC of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ satisfies $P$. It is then easy to provide a winning strategy for Player 2 in the same Banach-Mazur game as described above. Once a BSCC $C$ has been reached (after Player 1's first move or Player 2's first move), Player 2 will ensure to visit all the locations of $C$ (as Player 1 did in the above proof). The resulting infinite path will satisfy $P$ (by hypothesis), which implies the winning condition for Player 2. □

6.2.2. *What about fairness and probabilities?* The natural question is: can we fully extend Theorem 6.2 and therefore prove that almost-sure satisfaction is equivalent to (§)? We concentrate on the following equivalence:

$$\mathcal{A}, s \not\approx_{\mathbb{P}} P \;\Leftrightarrow\; \begin{cases}\text{every infinite thick } \textbf{and fair} \text{ symbolic path } \pi \\ \text{from } \iota(s) \text{ in } \mathsf{R}(\mathcal{A}) \text{ satisfies } P\end{cases} \qquad (\ddagger)$$

We show now that this equivalence is unfortunately not true in general. The counter-example to that equivalence is much more surprising than that of Figure 8. The reasons why it fails is rather subtle and is due to complex *time converging* behaviours. As pointed out in [CHR02], timed automata admit various time converging behaviours, and, in our context, some of these behaviours can lead to "big" sets of *unfair* executions. Inspired by an example presented in [CHR02], we design a two-clock timed automaton $\mathcal{A}_{\mathsf{unfair}}$ (see Figure 9) for which the equivalence (‡) does not hold. In this automaton, every (infinite) fair and thick symbolic path satisfies $\mathbf{G}\,\mathbf{F}\,p_1 \wedge \mathbf{G}\,\mathbf{F}\,p_2$, and in particular $\mathbf{F}\,p_2$. Thus, letting $\varphi = \mathbf{F}\,p_2$, the right-hand side of equivalence (‡) is true. However, when $\mathcal{A}_{\mathsf{unfair}}$ is equipped with uniform distributions, starting in $s_0 = (\ell_0, (0,0))$, one can show that the probability of the symbolic path $\pi(s_0, (e_3e_4e_5)^\omega)$ is positive and therefore $\mathcal{A}_{\mathsf{unfair}}, s_0 \not\approx_{\mathbb{P}} \mathbf{F}\,p_2$. We notice that this implies $\mathbb{P}_{\mathcal{A}_{\mathsf{unfair}}}(s_0 \models \mathsf{fair}) < 1$.

All this is proven formally in Appendix D.2, page 53.

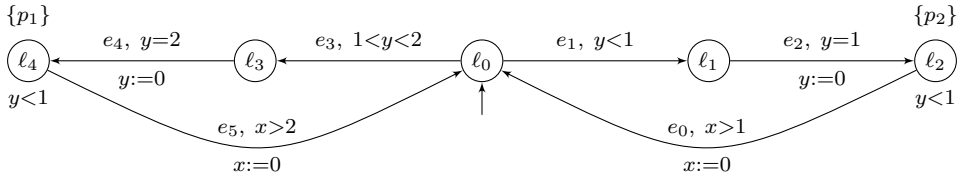



Figure 9: A two-clock automaton, $\mathcal{A}_{\mathsf{unfair}}$ with non negligible set of unfair runs.

6.2.3. *When fairness is almost-sure.* We will show that a sufficient condition to have (‡) is to have $\mathbb{P}_{\mathcal{A}}(s_0 \models \mathsf{fair}) = 1$. We can now state the following crucial theorem:

**Theorem 6.6.** Let $s$ be a state of $\mathcal{A}$, and $P$ be a prefix-independent (untimed) property over $\mathsf{AP}$. Assuming $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$, the following holds:

$$\mathcal{A}, s \models_{\mathbb{P}} P \;\Leftrightarrow\; \S(\mathcal{A}, s, P)\,.$$

*Proof.* First define the property $\mathsf{thick}$ as follows: letting $\varrho$ be a run in $\mathsf{R}(\mathcal{A})$, $\varrho \models \mathsf{thick}$ iff $\pi_\varrho$ is thick. Applying Theorem 6.2, we get that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{thick}) = 1$.

We now prove that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models P) > 0$ iff there exists an infinite symbolic path $\pi$ from $s$, which is thick and fair, and such that $\pi \models P$. This will imply the expected result. We prove the two implications separately.

- *Proof of the left-to-right implication.* Let us assume that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models P) > 0$. We have seen that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{thick}) = 1$. Therefore, thanks to the fact that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair}) = 1$, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models P) = \mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models P \wedge \mathsf{fair} \wedge \mathsf{thick})$. Hence,
$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models P \wedge \mathsf{fair} \wedge \mathsf{thick}) > 0.$$
In particular, there exists a fair thick infinite path from $s$ which satisfies $P$.
- *Proof of the right-to-left implication.* Let $\pi = \pi(s_0, e_1 e_2 \ldots)$ be a fair thick symbolic path in $\mathsf{R}(\mathcal{A})$ satisfying $P$. We consider the thick graph $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ of $\mathcal{A}$. Since $\pi$ is thick, $\pi$ is also a path in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$. Let us consider the strongly connected components of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$. As $\pi$ is a fair path, it eventually reaches a BSCC in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ and from then on takes each edge of the BSCC infinitely often. Otherwise, this would mean that $\pi$ ignores a thick edge forever, which would contradict the fairness assumption. Let $B_\pi$ be the BSCC that $\pi$ eventually reaches and $\pi_{\mathsf{pref}}$ the shortest prefix of $\pi$ leading from $s$ to $B_\pi$ (note that it is thick). Consider the following set of paths in $\mathsf{R}(\mathcal{A})$:
$$S \stackrel{\text{def}}{=} \{\pi' \in \mathsf{Cyl}(\pi_{\mathsf{pref}}) \mid \pi' \text{ is thick and fair}\}\,.$$
Since $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{thick}) = 1$ and $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair}) = 1$, we deduce that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(S) = \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi_{\mathsf{pref}}))$. Moreover since $\pi_{\mathsf{pref}}$ is thick, we obtain $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(S) > 0$. It now suffices to observe that all paths in $S$ satisfy $P$. Indeed, the satisfiability of prefix-independent (untimed) properties over $\mathsf{AP}$ only depends on the set of states that are visited infinitely often, and all paths in $S$ visit infinitely often exactly the states in $B_\pi$, and $\pi \models P$.

□

6.2.4. *Conclusion.* We can conclude this subsection by stating the main result concerning prefix-independent properties. It is a direct consequence of Theorems 6.5 and 6.6.

**Corollary 6.7.** Let $s$ be a state of $\mathcal{A}$, and $P$ be a prefix-independent (untimed) property over $\mathsf{AP}$. Assuming $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$, the following equivalences hold:

$$\mathcal{A}, s \models_{\mathbb{P}} P \;\Leftrightarrow\; \mathcal{A}, s \models_{\mathcal{T}} P \;\Leftrightarrow\; \S(\mathcal{A}, s, P)\,.$$

Note that characterisation (§) will help with algorithmic issues. Complexity issues will be discussed in Section 8.

6.3. **Extension to richer properties.** In this section we extend the previous study to properties which are richer than prefix-independent properties, in particular to LTL properties and properties expressed as specification timed automata.

Let $\langle \mathcal{A}, \mu, w \rangle$ be a stochastic timed automaton with $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$, and let $\mathcal{B} = (\mathsf{L}, \mathsf{i}_0, \mathsf{X}, \mathsf{AP}, \mathsf{E}, \mathcal{F})$ be a specification Büchi or Muller timed automaton. We build the product stochastic timed automaton $\langle \mathcal{A} \ltimes \mathcal{B}, \overline{\mu}, \overline{w} \rangle$ as follows. The timed automaton $\mathcal{A} \ltimes \mathcal{B} = (\overline{L}, X \cup \mathsf{X}, \overline{E}, \overline{\mathcal{I}})$ (with no labelling function) is such that:

- $\overline{L} = L \times \mathsf{L}$;
- $\overline{E}$ is composed of the following edges: if $e \stackrel{\text{def}}{=} (\ell \xrightarrow{g,Y} \ell') \in E$ then for all $\mathsf{e} \stackrel{\text{def}}{=} (\mathsf{l} \xrightarrow{\mathsf{g},\lambda(\ell),\mathsf{Y}} \mathsf{l}') \in \mathsf{E}$, there is an edge $((\ell, \mathsf{l}) \xrightarrow{g \wedge \mathsf{g}, Y \cup \mathsf{Y}} (\ell', \mathsf{l}'))$ in $\overline{E}$, which we write $e_{\mathsf{e}}$;
- $\overline{\mathcal{I}}((\ell, \mathsf{l})) = \mathcal{I}(\ell)$ for every $(\ell, \mathsf{l}) \in \overline{L}$.

The measures $\overline{\mu}$ and the weights $\overline{w}$ are such that:

- $\overline{\mu}_{(\ell,\mathsf{l})} = \mu_\ell$ for every $(\ell, \mathsf{l}) \in \overline{L}$, and
- $\overline{w}_{\overline{e}} = w_e$ for every edge $\overline{e} \in \overline{E}$ which comes from edge $e$.

Given a state $s = (\ell, v)$ of $\mathcal{A}$, we define the initial state in $\mathcal{A} \ltimes \mathcal{B}$ as $\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s) \stackrel{\text{def}}{=} ((\ell, \mathsf{i}_0(\mathcal{L}(s))), v\mathbf{0}_{\mathsf{X}})$; that is, we start in $\mathcal{B}$ from the location specified by the label of state $s$ (this is $\mathsf{i}_0(\mathcal{L}(s))$), with all clocks of $\mathcal{B}$ set to 0.

Note that any run $\varrho$ in $\mathcal{A}$ from state $s$ has a unique image in $\mathcal{A} \ltimes \mathcal{B}$ from $\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s)$, denoted $\varrho^{\mathcal{B}}$ (since $\mathcal{B}$ is complete and deterministic). Note that the converse also holds: for any run $\varrho'$ in $\mathcal{A} \ltimes \mathcal{B}$ from some $\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s)$, there is a unique preimage $\varrho$ in $\mathcal{A}$ from $s$, such that $\varrho' = \varrho^{\mathcal{B}}$. We define the $\omega$-regular property $P_{\mathcal{A}\ltimes\mathcal{B}}$ in $\mathcal{A} \ltimes \mathcal{B}$ as the lifting of $\mathcal{F}$ in $\mathcal{A} \ltimes \mathcal{B}$ (an infinite run in $\mathcal{A} \ltimes \mathcal{B}$ satisfies $P_{\mathcal{A}\ltimes\mathcal{B}}$ whenever its projection on $\mathcal{B}$ satisfies the accepting condition $\mathcal{F}$). As $\mathcal{F}$ is an internal prefix-independent condition in $\mathcal{B}$, $P_{\mathcal{A}\ltimes\mathcal{B}}$ is an internal prefix-independent condition in $\mathcal{A} \ltimes \mathcal{B}$.

**Remark 6.8.** It should be clear enough that $\mathcal{A} \ltimes \mathcal{B}$ is non-blocking assuming $\mathcal{A}$ is. Moreover, for all states $s = (\ell, v)$ of $\mathcal{A}$, for all states $\mathsf{s} = (\mathsf{l}, \mathsf{v})$ of $\mathcal{B}$, and for every edge $e \in E$,

$$I(s, e) = \bigcup_{\mathsf{e} = (\mathsf{l} \xrightarrow{\mathsf{g},\lambda(\ell),\mathsf{Y}} \mathsf{l}')} I(((\ell, \mathsf{l}), (v\mathsf{v})), e_{\mathsf{e}})$$

This allows to properly define the probability measure $\mathbb{P}_{\mathcal{A}\ltimes\mathcal{B}}$.

We can now state the main theorem for specification timed automata, which uses this product construction. Its proof is given in Appendix D.3, page 55.

**Theorem 6.9.** Let $s$ be a state of $\mathcal{A}$, and $\mathcal{B}$ be a specification Büchi or Muller timed automaton. Assuming $\mathbb{P}_{\mathcal{A}\ltimes\mathcal{B}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s) \models \mathsf{fair}) = 1$, the following holds:

$$\mathcal{A}, s \not\approx_{\mathbb{P}} \mathcal{B} \;\Leftrightarrow\; \mathcal{A}, s \not\approx_{\mathcal{T}} \mathcal{B} \;\Leftrightarrow\; \S(\mathcal{A} \ltimes \mathcal{B}, \mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), P_{\mathcal{A}\ltimes\mathcal{B}})\,.$$

The above also applies to specification untimed automata. In particular, it also applies to specification automata corresponding to LTL formulas. It is now easy to be convinced that if $\mathcal{B}$ is a specification untimed automaton, then $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = \mathbb{P}_{\mathcal{A}\ltimes\mathcal{B}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s) \models \mathsf{fair})$ since $\mathcal{B}$ does not restrict guards of edges, and in particular

$$\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1 \;\Leftrightarrow\; \mathbb{P}_{\mathcal{A}\ltimes\mathcal{B}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s) \models \mathsf{fair}) = 1\,.$$

This allows to get the following important corollary, which characterises the almost-sure model-checking for LTL.

**Corollary 6.10.** Let $s$ be a state of $\mathcal{A}$, and $\varphi$ be an $\mathsf{LTL}$ formula over $\mathsf{AP}$. Assuming $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$, the following holds:

$$\mathcal{A}, s \bowtie_{\mathbb{P}} \varphi \;\Leftrightarrow\; \mathcal{A}, s \bowtie_{\mathcal{T}} \varphi \;\Leftrightarrow\; \S(\mathcal{A} \ltimes \mathcal{B}_\varphi, \mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}_\varphi}(s), P_{\mathcal{A}\ltimes\mathcal{B}_\varphi})\,.$$

## 7. Application to several classes of timed automata

In the previous section, we showed that, provided fairness is almost-sure, one could characterise almost-sure satisfaction and large satisfaction using thick paths. We will describe here two classes of stochastic timed automata for which this holds.

7.1. **Single-clock timed automata.** In this section, we focus on single-clock timed automata, and we show that, under some minor additional technical hypotheses, single-clock timed automata are almost-surely fair. In particular, Corollary 6.10 will apply, yielding the decidability of the almost-sure model-checking problem for $\omega$-regular properties on this class of stochastic timed automata.

Let $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ be a single-clock stochastic timed automaton. We assume the following conditions on $\mathcal{A}$, denoted (†):

**(H3):** For all $\ell \in L$, for all $[a, b] \subseteq \mathbb{R}_+$, the function $v \mapsto \mu_{(\ell,v)}([a, b])$ is continuous;

**(H4):** If $s' = s + t$ for some $t \geq 0$, and $0 \notin I(s + t', e)$ for every $0 \leq t' \leq t$, then $\mu_s(I(s, e)) \leq \mu_{s'}(I(s', e))$;

**(H5):** There is $0 < \lambda_0 < 1$ s.t. for every state $s$ with $I(s)$ unbounded, $\mu_s([0, 1/2]) \leq \lambda_0$.

**Remark 7.1.** The three last requirements are technical and needed to properly define a probability measure over infinite runs, but they are natural and easily satisfiable. For instance, a timed automaton equipped with uniform (resp. exponential[11]) distributions on bounded (resp. unbounded) intervals satisfy these conditions. If we assume exponential distributions on unbounded intervals, the very last requirement corresponds to the bounded transition rate condition in [DP03], required to have reasonable and realistic behaviours.

**Theorem 7.2.** Assuming $\mathcal{A}$ satisfies (†), if $s$ is a state of $\mathcal{A}$, $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$.

The proof of this theorem is very technical. We will describe the main ingredients of the proof in the core of the paper, and postpone all details to Appendix E.

Let $\{c_i \mid 0 \leq i \leq k\}$ be the set of constants appearing in guards of $\mathcal{A}$, assuming w.l.o.g. $c_0 = 0$. We know [LMS04] that the following intervals are regions for $\mathcal{A}$:

$$\{c_i\} \text{ for } 0 \leq i \leq k; \qquad (c_i, c_{i+1}) \text{ for } 0 \leq i < k; \qquad (c_k, +\infty)$$

We assume $\mathsf{R}(\mathcal{A})$ is built with these regions. It is polynomial-size (contrary to standard region automaton which is exponential-size).

The proof of the above theorem then relies on Lemma 7.3 below. A *subregion* of a region $q$ is a pair $(q, J)$ such that $J \subseteq q$ is an interval. If $s \in J$, we may write $s \in (q, J)$ as well. If $(q, J)$ and $(q', J')$ are subregions, we write $(q, J) \xrightarrow{e} (q', J')$ to express that $(q, v) \xrightarrow{\tau,e} (q', v')$ for some $v \in J$, $v' \in J'$ and $\tau \in \mathbb{R}_+$. In the sequel to ease the reading, we will use $\mathsf{LTL}$-like notations, like $\mathbb{P}_{\mathcal{A}}(s, \mathbf{G}\,\mathbf{F}\,(q, J) \xrightarrow{e} (q', J') \mid \mathbf{G}\,\mathbf{F}\,(q, J))$ to denote the conditional probability of the set of runs $s_0 \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} s_2 \cdots$ such that $s_0 = s$ and

[11] With bounded transition rates, see [DP03].

$\{s_i \xrightarrow{e_{i+1}} s_{i+1} \mid s_i \in J,\ e_{i+1} = e,\ \text{and } s_{i+1} \in J'\}$ is infinite, assuming that the set $\{s_i \mid s_i \in J\}$ is infinite. We will use other similar notations, that we expect are sufficiently explicit to be understandable.

**Lemma 7.3.**

(1) For every subregion $(q, J)$ of $q$ such that (i) $J$ is non-empty and open in $q$ (for the induced topology), and (ii) $\overline{J} \subseteq q$ is compact,
(2) for every thick edge $e$ enabled in $q$,
(3) for every subregion $(q', J')$ of $q'$ such that for every $s \in (q, J)$, $e(s) \cap J'$ is non-empty and open in $q'$ (for the induced topology), where $e(s) = \{s' \mid \exists \tau \in \mathbb{R}_+ \text{ s.t. } s \xrightarrow{\tau, e} s'\}$,
(4) for every state $s$ of $\mathsf{R}(\mathcal{A})$ such that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q, J)) > 0$,[12]

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q, J) \xrightarrow{e} (q', J') \mid \mathbf{G}\,\mathbf{F}\,(q, J)) = 1\,.$$

The idea of this lemma is to provide a lower-bound on the probability of firing the transition $(q, J) \xrightarrow{e} (q', J')$ each time we visit $(q, J)$. By thickness of $e$, we know that the probability at each visit is positive, but as $\overline{J}$ is compact, we infer a positive uniform lower-bound $\lambda$. This is the main ingredient to prove the result.

**Remark 7.4.** This lemma holds for all timed automata, not only one-clock timed automata.

We have shown the proof for a single edge, but this lemma can be extended straightforwardly to finite sequences of edges as follows:

**Lemma 7.5.** (1) For all regions $(q_i)_{0 \le i \le p}$,
(2) for all edges $(e_i)_{1 \le i \le p}$ such that $e_i$ is thick and enabled in $q_{i-1}$,
(3) for all subregions $((q_i, J_i))_{0 \le i \le p}$ such that for every $0 \le i < p$:
   (a) $J_i$ is non-empty and open in $q_i$ (for the induced topology),
   (b) $\overline{J_i} \subseteq q_i$ is compact, and
   (c) for every $s \in J_i$, $e_i(s) \cap J_{i+1}$ is non-empty and open, where $e_i(s) = \{s' \mid \exists \tau \in \mathbb{R}_+ \text{ s.t. } s \xrightarrow{\tau, e_i} s'\}$,
(4) for every state $s$ of $\mathcal{A}$ such that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q_0, J_0)) > 0$

it holds that:

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,\sigma \mid \mathbf{G}\,\mathbf{F}\,(q_0, J_0)) = 1$$

where $\sigma = (q_0, J_0) \xrightarrow{e_1} (q_1, J_1) \dots \xrightarrow{e_p} (q_p, J_p)$.

Now, we can turn back to Theorem 7.2.

*Sketch of proof of Theorem 7.2.* Let $s$ be a state in $\mathcal{A}$. We want to prove that $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$. We will equivalently prove $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{fair}) = 1$. To that purpose, we decompose the set of infinite runs in $\mathsf{R}(\mathcal{A})$ from $\iota(s)$ into:

($F_1$) the set of runs with infinitely many resets,
($F_2$) the set of runs with finitely many resets, and which are ultimately in the unbounded region $(c_k, +\infty)$,
($F_3$) the set of runs with finitely many resets, and which ultimately stay forever in a bounded region, either $\{c_i\}$ with $0 \le i \le k$, or $(c_i, c_{i+1})$ with $0 \le i < k$. We write $(F_3^{(c_i, c_{i+1})})$ (resp. $(F_3^{c_i})$) for condition $F_3$ restricted to $(c_i, c_{i+1})$ (resp. $\{c_i\}$).

[12] This is for the next conditional probability to be defined.

We write $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j)$ for the probability of the runs starting in $s$ and satisfying condition $F_j$. The three sets of runs above are measurable and partition the set of all runs. Hence $\sum_{j=1,2,3} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j) = 1$, and applying Bayes formula:

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair}) = \sum_{j=1,2,3} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_j) \cdot \mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j)\,. \qquad (\bullet)$$

We now distinguish between the three cases to prove that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_j) = 1$ (in case $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j) = 0$ we remove the corresponding term from $(\bullet)$).

**Case $F_1$.:** In that case, for all states $(q, 0)$ which are visited infinitely often we apply Lemma 7.5 to any sequence of thick edges, and get the expected result for $F_1$.

**Case $F_2$.:** Once the unbounded region is reached, precise values of clocks are irrelevant, and the timed automaton roughly behaves like a finite Markov chain, which yields the expected result for $F_2$.

**Case $F_3$.:** We consider runs which end up in a bounded region $r$. This case is only possible (with positive probability) if no thick edge is enabled infinitely often with a guard above $r$ (otherwise it would be taken infinitely often – due to the hypothesis $(H4)$). Therefore in this case as well the automaton ultimately behaves like a finite Markov chain, which allows to conclude with the expected property. □

Note that the one-clock hypothesis is crucial for cases $F_1$ and $F_3$.

*Checking almost-sure non-Zenoness in one-clock timed automata.* Note that we cannot obtain the result on Zeno behaviours from Corollary 6.10 since Zenoness is a timed property. We can neither use the product construction of Section 6.3 and Theorem 6.9 since this will increase the number of clocks to 2. Therefore we need a specific proof, that we present now.

We first show the following crucial lemma.

**Lemma 7.6.** Assuming $\mathcal{A}$ satisfies $(\dagger)$, if $s$ is a state of $\mathcal{A}$, then:

$$\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = \sum_{B \text{ Zeno BSCC of } \mathcal{G}_{\mathsf{t}}(\mathcal{A})} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathbf{F}\, B)$$

where a BSCC of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ is said Zeno whenever it is bounded and the clock is never reset.

To prove the lemma, we need to realise that runs ending up in the unbounded region or runs with infinitely many resets have probability 0 to be Zeno. Indeed, in the first case, there will be non constraint on the clock for taking transition, and in particular, with probability 1 a delay of 1 will elapse infinitely often; in the second case, infinitely often a guard of the form $0 < x < 1$ will be enabled, and therefore with probability 1, a delay of at least $1/2$ will be done, yielding with probability 1 a non-zeno run. It remains those runs ending up in a bounded region, which will correspond to the right-hand side of the equality in the statement (due to fairness, the runs end up with probability 1 in a BSCC). The details are given in the Appendix on page 59.

**Theorem 7.7.** Assuming $\mathcal{A}$ satisfies $(\dagger)$, if $s$ is a state of $\mathcal{A}$, then the three following properties are equivalent:

(a) $\mathcal{A}, s \not\approx_{\mathbb{P}} \neg\mathsf{Zeno}$;
(b) $\mathcal{A}, s \not\approx_{\mathcal{T}} \neg\mathsf{Zeno}$;
(c) no Zeno BSCC is reachable in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ from $\iota(s)$.

This theorem is a consequence of Lemma 7.6, and uses once more Banach-Mazur games for what concerns the large satisfaction. The details are also given in the Appendix on page 62.

**Corollary 7.8.** The almost-sure non-Zenoness problem is NLOGSPACE for single-clock stochastic timed automata which satisfy condition (†).

Condition $(c)$ in Theorem 7.7 can be checked in NLOGSPACE (recall that in that case $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ is polynomial in the size of $\mathcal{A}$).

7.2. **Reactive and weak-reactive timed automata.** Although fairness is not almost-sure in general for $n$-clock stochastic timed automata with $n \geq 2$ (see Figure 9), it is the case for the subclass of *reactive* (and weak-reactive) stochastic timed automata. Let us first recall that a (non-stochastic) timed automaton $\mathcal{A}$ is reactive if $I(s) = \mathbb{R}_+$ for all states $s$ of $\mathcal{A}$.

We first focus on the class of reactive stochastic timed automata, and later extend the results to the class of weak-reactive stochastic timed automata.

**Definition 7.9.** A stochastic timed automaton $\langle \mathcal{A}, \mu, w \rangle$ is *reactive* whenever the timed automaton $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ is reactive, and for every $\ell \in L$, there exists a probability distribution $\mu_\ell$ over $\mathbb{R}_+$, equivalent to the Lebesgue measure, such that for every $v \in \mathbb{R}_+^X$, $\mu_{(\ell,v)} = \mu_\ell$.

Note that for any constant $C$, for any $\ell \in L$, $\mu_\ell([0, C]) < 1$, this is due to the equivalence of $\mu_\ell$ with the Lebesgue measure. Note also that if $s = (\ell, v)$ and $s' = (\ell, v')$ are such that $v \cong_{\mathcal{A}} v'$, then $p_s(e) = p_{s'}(e)$ for every edge $e$.

**Example 7.10.** Examples of distributions over delays that respect the above conditions are exponential distributions, but we can think of many other kinds of distributions. Later, all our examples will use exponential distributions. Each such distribution is characterised by a positive parameter $\lambda_\ell$, and its density is $t \mapsto \lambda_\ell \cdot e^{-\lambda_\ell \cdot t}$. ⌟

Note that reactive stochastic timed automata generalise continuous-time Markov chains (CTMC for short). A CTMC is nothing else than a single-clock reactive stochastic timed automaton in which (i) on all transitions, the guard is trivial, and the clock is reset, and (ii) each location is assigned an exponential distribution over delays.

Let $\langle \mathcal{A}, \mu, w \rangle$ be a reactive stochastic timed automaton. The goal of this section is to prove the following result, which will allow to apply results of Section 6, and in particular Theorem 6.9.

**Proposition 7.11.** Let $s$ be a state of $\mathcal{A}$. Then $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$.

We first recall some basic probability results.

**Lemma 7.12.** Let $\mathbb{P}$ be a probability measure on some probabilistic space $\Omega$. Let $A$, $B$ and $C$ be measurable sets such that $\mathbb{P}(B) > 0$ and $\mathbb{P}(C) > 0$, then

(1) If $\mathbb{P}(A) = 1$, then $\mathbb{P}(A \mid B) = 1$.
(2) If $\mathbb{P}(A \mid B) = 1$, $\mathbb{P}(B \mid C) = 1$, then $\mathbb{P}(A \mid C) = 1$.
(3) If $\mathbb{P}(A \mid B) = 1$, $\mathbb{P}(A \mid C) = 1$, then $\mathbb{P}(A \mid B \cup C) = 1$.

We write $\langle \mathsf{R}(\mathcal{A}), \mu, w\rangle$ for the stochastic timed region automaton based on $\mathcal{A}$ (we abusively also write $\mu$ and $w$ since we will not use that notation for $\mathcal{A}$), and we write $Q$ for the set of locations of $\mathsf{R}(\mathcal{A})$, and $T$ for its set of edges. Thanks to Corollary 3.7, in order to prove Proposition 7.11, it is sufficient to prove that $\mathsf{R}(\mathcal{A})$ is almost-surely fair. In the following we denote by $M$ the maximal constant appearing in $\mathcal{A}$ (or $\mathsf{R}(\mathcal{A})$), and we write $\mathbb{P}$ instead of $\mathbb{P}_{\mathsf{R}(\mathcal{A})}$.

To prove almost-sure fairness in $\mathsf{R}(\mathcal{A})$, we have to show that for every thick edge $e$, the probability to visit $e$ infinitely often, knowing we visit $\mathsf{source}(e)$ infinitely often, is equal to 1. The key point of this proof lies in the fact that, since the automaton $\mathsf{R}(\mathcal{A})$ is reactive, there exists a subset of regions, called *memoryless*, that will be visited infinitely often with probability 1. A region $r$ is said *memoryless* whenever the following holds for every clock $x \in X$: either $v(x) = 0$ for every $v \in r$, or $v(x) > M$ for every $v \in r$. The interest of memoryless regions is that once you reach such a region the future (and its probability) is independent of both the finite prefix and the clock valuations. In particular, visiting infinitely often memoryless regions prevents converging phenomena as the one observed in Figure 9. For each memoryless region $r$, we distinguish a canonical valuation $v_r \in r$ defined by $v_r(x) = 0$ or $v_r(x) = M+1$ for every $x \in X$ (note that this valuation is uniquely defined). If $q = (\ell, r) \in Q$ is such that $r$ is memoryless, we distinguish the canonical configuration $s_q = (\ell, v_r)$ (or $s_q = ((\ell, r), v_r)$ in $\mathsf{R}(\mathcal{A})$).

The fact that memoryless regions are visited infinitely often almost-surely is formalised in Lemma 7.13. Then it remains to show that knowing we visit infinitely often such a memoryless region, we visit a reachable thick edge $e$ infinitely often with probability 1. To this end, we investigate the set of runs that visit infinitely often memoryless regions and $e$, and we conclude thanks to a judicious decomposition of this set and Borel-Cantelli lemma, this is formalised in Lemma 7.14. A sketch of proof is given for Lemma 7.13 and the complete proofs of both lemmas are provided in Appendix F, page 63.

In order to formalise these ideas, we need to introduce further notations. Let $s$ be a state of $\mathsf{R}(\mathcal{A})$, that we will take as initial. If $e$ is a thick edge in $T$, and $q \in Q$, we write $\mathfrak{R}^e(s)$ for the set of runs in $\mathsf{R}(\mathcal{A})$ that start in $s$ and take $e$ infinitely often, and $\mathfrak{R}^q(s)$ for the set of runs of $\mathsf{R}(\mathcal{A})$ that start in $s$ and visit $q$ infinitely often. In particular, we write $\mathfrak{R}^{\mathsf{source}(e)}(s)$ for the set of runs that start in $s$ and visit $\mathsf{source}(e)$ infinitely often (hence along which $e$ is enabled infinitely often).

We fix a thick edge $e$ in $T$, and we let $\mathcal{Q}$ be the set of pairs $q = (\ell, r)$ where $r$ is memoryless and $\mathcal{Q}'$ the set of elements $q = (\ell, r) \in \mathcal{Q}$ such that

$$\mathbb{P}(\mathfrak{R}^q(s)) > 0 \quad \text{and} \quad \mathbb{P}(\mathfrak{R}_0^{q,e}(s_q)) > 0$$

where $\mathfrak{R}_0^{q,e}(s_q)$ is the set of runs that start from $s_q$ and take $e$ before any other visit to $q$.

**Lemma 7.13.** Assuming the above notations,

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s)\right) = 1\,.$$

*Sketch of proof.* We notice that the set of runs that delay infinitely many times more than $M$ time units before taking a transition is a subset of $\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s)$. Indeed, if a run $\varrho$ delays more than $M$ time units before taking the $n$-th transition then each clock is either reset on the $n$-th transition (hence its value is 0), or its value exceeds $M$. Now, as for every $\ell \in L$, we have assumed $\mu_\ell$ is equivalent to the Lebesgue measure on $\mathbb{R}_+$, it holds that $\mu_\ell([0, M]) < 1$.

We can then prove that the probability of the set of runs that delay only finitely many times more than $M$ time units is zero, since $L$ is finite, which concludes the proof. □

**Lemma 7.14.** Assuming the above notations, for all $q \in \mathcal{Q}'$

$$\mathbb{P}\left(\mathfrak{R}^e(s) \mid \mathfrak{R}^q(s)\right) = 1.$$

Now assuming Lemma 7.13 and Lemma 7.14, we will prove Proposition 7.11.

*Proof of Proposition 7.11.* We want to prove that the probability of being fair is 1, hence we want to prove that for every thick edge $e$ with $\mathbb{P}(\mathfrak{R}^{\mathsf{source}(e)}(s)) > 0$,

$$\mathbb{P}\left(\mathfrak{R}^e(s) \mid \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = 1\,.$$

Let $e$ be a thick edge with $\mathbb{P}(\mathfrak{R}^{\mathsf{source}(e)}(s)) > 0$. By Lemma 7.13, we have that

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s)\right) = 1, \tag{7.1}$$

and by Lemma 7.14, we have that

$$\mathbb{P}\left(\mathfrak{R}^e(s) \mid \mathfrak{R}^q(s)\right) = 1 \tag{7.2}$$

for any $q \in \mathcal{Q}'$. Applying Lemma 7.12 (point 3.), we deduce from Equation (7.2) that

$$\mathbb{P}\left(\mathfrak{R}^e(s) \mid \bigcup_{q\in\mathcal{Q}'} \mathfrak{R}^q(s)\right) = 1 \tag{7.3}$$

and applying Lemma 7.12 (point 1.), we deduce from Equation (7.1) that:

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s) \mid \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = 1. \tag{7.4}$$

Moreover, we can easily show that

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s) \mid \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = \mathbb{P}\left(\bigcup_{q\in\mathcal{Q}'} \mathfrak{R}^q(s) \mid \mathfrak{R}^{\mathsf{source}(e)}(s)\right). \tag{7.5}$$

Indeed, we just have to prove that

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s) \cap \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = \mathbb{P}\left(\bigcup_{q\in\mathcal{Q}'} \mathfrak{R}^q(s) \cap \mathfrak{R}^{\mathsf{source}(e)}(s)\right)$$

and it is thus sufficient to prove that

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}\setminus\mathcal{Q}'} \mathfrak{R}^q(s) \cap \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = 0\,.$$

However, if $q \in \mathcal{Q}\setminus\mathcal{Q}'$, we have $\mathbb{P}(\mathfrak{R}^q(s)) = 0$ or $\mathbb{P}(\mathfrak{R}_0^{q,e}(s_q)) = 0$. Now, if $\mathbb{P}(\mathfrak{R}^q(s)) = 0$, we have

$$\mathbb{P}\left(\mathfrak{R}^q(s) \cap \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = 0$$

and if $\mathbb{P}(\mathfrak{R}_0^{q,e}(s_q)) = 0$, we also have

$$\mathbb{P}\left(\mathfrak{R}^q(s) \cap \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = 0\,.$$

We therefore deduce from Equations (7.4) and (7.5) that

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}'} \mathfrak{R}^q(s) \mid \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = 1\,. \tag{7.6}$$

Applying Lemma 7.12 (point 2.), we get the expected result from Equations (7.3) and (7.6):

$$\mathbb{P}\left(\mathfrak{R}^e(s) \mid \mathfrak{R}^{\mathsf{source}(e)}(s)\right) = 1\,. \qquad \square$$

7.2.1. *Extension to weak reactive stochastic timed automata.* We now extend the almost-sure fairness from the subclass of reactive stochastic timed automata to the larger class of *weak reactive* stochastic timed automata, defined as follows.

**Definition 7.15.** A stochastic timed automaton $\langle \mathcal{A}, \mu, w\rangle$ with $\mathcal{A} = (L, X, E, \mathcal{I}, \mathcal{L})$ is said to be *weak reactive* whenever $L$ is the disjoint union of sets $L_u$ and $L_b$ such that $\ell \in L_u$ if and only if $I(\ell, v)$ is unbounded for all $v$ such that $v \models \mathcal{I}(\ell)$, and:

- for every pair $s = (\ell, v)$, $s' = (\ell, v')$ satisfying for every $x \in X$, $v(x) = v'(x)$ or $\min(v(x), v'(x)) > M$,
$$\mu_s = \mu_{s'}\,;$$
- there exists $0 < \lambda_0 \leq 1$ such that for every $\ell \in L_u$, for every $v \models \mathcal{I}(\ell)$, we have that
$$\mu_{(\ell,v)}\left([M, +\infty[\,\right) \geq \lambda_0\,;$$
- there exist $0 < \lambda_1 \leq 1$ and $N \geq 1$ such that for any $\ell \in L_b$, for every $v \models \mathcal{I}(\ell)$, we have that
$$\mathbb{P}_{\mathcal{A}}\Big(\bigcup_{(e_1,\ldots,e_N)\in E_u} \pi((\ell, v), e_1, \ldots, e_N)\Big) \geq \lambda_1$$
where $E_u = \{(e_1, \ldots, e_N) \mid \mathsf{target}(e_i) \in L_u \text{ for some } 1 \leq i < N\}$.

It is obvious that the class of reactive stochastic timed automata is a subclass of weak reactive stochastic timed automata. In fact, the main difference between these classes of automata lies on the existence of some states $s$ such that $I(s)$ is bounded.

The proof of almost-sure fairness of reactive stochastic timed automata is based on two lemmas: Lemma 7.13 and Lemma 7.14. The proof of Lemma 7.14 works in the same way for weak reactive stochastic timed automata, using the fact that for every pair $s = (\ell, v)$, $s' = (\ell, v')$ satisfying for every $x \in X$, $v(x) = v'(x)$ or $\min(v(x), v'(x)) > M$, we have $\mu_s = \mu_{s'}$. The proof of Lemma 7.13 can also be adapted in view of properties of weak reactive stochastic timed automata in order to obtain the following lemma:

**Lemma 7.16.** Let $\mathcal{A}$ be a weak reactive stochastic timed automaton and $s$ be a state of $\mathcal{A}$. Then

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s)\right) = 1\,.$$

Thanks to these two lemmas, we deduce the almost-sure fairness for weak reactive automata as in the case of reactive stochastic timed automata:

**Proposition 7.17.** Let $\mathcal{A}$ be a weak reactive stochastic timed automaton and $s_0$ be a state of $\mathcal{A}$. Then $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$.

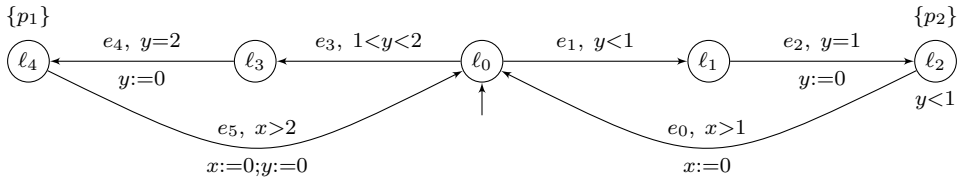



Figure 10: Automaton $\mathcal{A}^1_{\mathsf{unfair}}$.

We end up this subsection by exhibiting some examples of weak reactive stochastic timed automata:

**Example 7.18.**

(1) Let $\mathcal{A}$ be a stochastic timed automaton such that for any $\ell \in L_u$, there exists a probability distribution $\mu_\ell$ over $\mathbb{R}_+$, equivalent to the Lebesgue measure, such that for every $v \in \mathbb{R}_+^X$, $\mu_{(\ell,v)} = \mu_\ell$. If $L_b$ does not contain any cycle then $\mathcal{A}$ is weak reactive.

(2) Let $\lambda > 0$. The modification of $\mathcal{A}_{\mathsf{unfair}}$ given on Figure 10, where $\mu_{(\ell_4,v)}$ is the exponential distribution of parameter $\lambda$, is weak reactive.

The other modification of $\mathcal{A}_{\mathsf{unfair}}$ given on Figure 11, where $\mu_{(\ell_2,v)}$ is the exponential distribution of parameter $\lambda$, is not weak reactive. ⌟

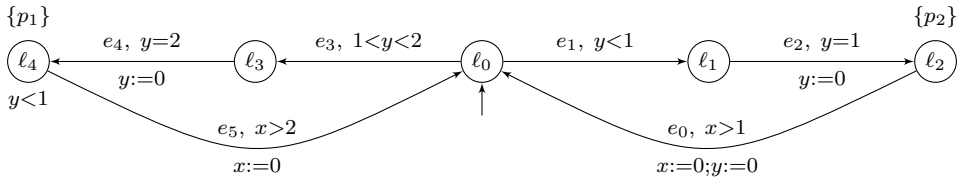



Figure 11: Automaton $\mathcal{A}^2_{\mathsf{unfair}}$.

*Discussion on Zenoness.* A side-result of the proofs of Lemmas 7.13 and 7.16 is that the set of Zeno runs in a (weak) reactive stochastic timed automaton is negligible, which implies in particular that the almost-sure non-Zenoness problem is trivial.

**Proposition 7.19.** Let $\mathcal{A}$ be a (weak) reactive stochastic timed automaton, and $s$ be a state of $\mathcal{A}$. Then $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = 0$.

Intuitively, this is because almost-surely, time will increase by a lower-bounded amount. The proof requires details of those of Lemmas 7.13 and 7.16, and is therefore postponed to the Appendix, on page 71.

## 8. Decidability and complexity results

We will use the thick graph construction for deciding almost-sure model-checking problem. For safety properties, we will use the characterisation given in Theorem 6.2, whereas we will use condition (§) for more general properties, under the assumption that fairness is almost-sure.

**Lemma 8.1.** Let $\mathcal{A}$ be a timed automaton over $\mathsf{AP}$, $s$ a state of $\mathcal{A}$, and $P$ be a Büchi or Muller property over $\mathsf{AP}$. Assume that $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ has size $f(\mathcal{A})$, then we can decide in nondeterministic $\log(f(\mathcal{A}))$-space whether condition $\S(\mathcal{A}, s, P)$ holds.

*Proof.* The algorithm for checking condition $\S(\mathcal{A}, s, P)$ is the following:

- guess a state $q$ of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$;
- check that there is a (thick) path from $[s]$ to $q$ in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$;
- check that $q$ belongs to a BSCC of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ which satisfies property $P$.

Note that as $P$ is prefix-independent, we can assume the thick path from $[s]$ to $q$ is simple. All this can be done on-the-fly and in non-deterministic $\log(f(\mathcal{A}))$-space. Note that checking $P$ in a BSCC can be done in $\log(f(\mathcal{A}))$-space as well. □

**Lemma 8.2.** Let $\mathcal{A}$ be a timed automaton over $\mathsf{AP}$, $s$ a state of $\mathcal{A}$, and $P$ be a simple safety property over $\mathsf{AP}$. Assume that $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ has size $f(\mathcal{A})$, then we can decide in non-deterministic $\log(f(\mathcal{A}))$-space whether there is an infinite path $\pi$ from $\iota(s)$ in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ which does not satisfy $P$.

*Proof.* Here is a possible algorithm with the expected complexity:

- guess a state $q$ of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$;
- check that there is a (thick) path from $[s]$ to $q$ in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$;
- check that this finite path violates property $P$.

Note that as $P$ is a simple safety property, we can assume that the path between $[s]$ and $q$ is simple. □

We now state the following lemma, whose proof follows from the definition of $\mathcal{A} \ltimes \mathcal{B}$ and from the definition of (weak) reactiveness. This will imply that, under the mentioned conditions, $\mathcal{A} \ltimes \mathcal{B}$ is almost-surely fair, which will allow to apply Theorem 6.9 to (weak) reactive stochastic timed automata.

**Lemma 8.3.** Let $\mathcal{A}$ be a (weak) reactive stochastic timed automaton, and $\mathcal{B}$ be a specification timed automaton. Then $\mathcal{A} \ltimes \mathcal{B}$ is a (weak) reactive stochastic timed automaton.

We apply the results from Section 6, and obtain the following principal theorem, which states decidability and complexity results for the almost-sure model-checking. Notice that in all those cases, almost-sure model-checking coincides with large model-checking.

**Theorem 8.4.**
(i) The almost-sure model-checking of stochastic timed automata for simple safety properties is PSPACE-complete.
(ii) The almost-sure model-checking of single-clock stochastic timed automata for Büchi or Muller properties, or for properties given as specification (untimed) automata, is NLOGSPACE-complete.[13]
(iii) The almost-sure model-checking of single-clock stochastic timed automata for properties given as LTL formulas is PSPACE-complete.
(iv) The almost-sure model-checking of (weak) reactive stochastic timed automata for Büchi or Muller properties or properties given as specification timed automata is PSPACE-complete.

All upper bounds are then obtained *via* Lemmas 8.1 and 8.2, since the size of the region automaton (and therefore the thick graph) is exponential [AD94], except for single-clock timed automata, where it is only polynomial-size [LMS04]. Note that if $\mathcal{A}$ is a timed automaton and $\mathcal{B}$ a specification automaton, then the size of $\mathcal{G}_{\mathsf{t}}(\mathcal{A} \ltimes \mathcal{B})$ is exponential in the two following cases: $\mathcal{B}$ is a specification timed automaton, and $\mathcal{B}$ is a specification untimed

[13]Note that simple safety or simple reachability properties can be expressed as small specification untimed automata, which yield an NLOGSPACE upper bound in those cases as well.

automaton of size at most exponential. The lower bounds are proven in Appendix G. Note that to establish the lower-bound in (i), the classical PSPACE-hardness proof of reachability in timed automata has to be adapted, since it is based on punctual guards that would yield negligible behaviours in the context of stochastic timed automata.

## 9. Conclusion

In this article we introduced and studied the model of stochastic timed automata that combines real-time constraints and probabilities. We considered the almost-sure model-checking problem and designed an abstraction that can be used to prove decidability of the above, provided fairness is almost-sure in the model. We identified two main classes of automata for which this is the case, the class of single-clock timed automata and that of weak reactive timed automata. In the two cases, the proof of almost-sure fairness is non-trivial and requires intricate arguments.

A remaining open problem is the decidability status of the almost-sure model-checking problem for the general class of stochastic timed automata, already for reachability properties.

As future work, we want to extend our study to quantitative model-checking, that is, compute the probability of a given property in an automaton. This has partly been solved for single-clock automata in [BBBM08], but more importantly, we would like to do it for the class of (weak) reactive stochastic timed automata, which allows for more complex timed constraints.

Compositionality is often a key for the description of real systems. Defining a composition of stochastic timed automata seems non-trivial in general, but the model of reactive stochastic timed automata seems to be well-suited for compositional design, since time can never be blocked by a component.

## Acknowledgement

Nathalie Bertrand and Patricia Bouyer were partly supported by ANR project ImpRo (number ANR-10-BLAN-0317). Patricia Bouyer was furthermore supported by the ERC Starting grant EQualIS (number 308087). Thomas Brihaye was partly supported by the ARC project (number AUWB-2010-10/15-UMONS-3), a grant "Mission Scientifique" from the F.R.S.-FNRS, the FRFC project (number 2.4545.11), and the EU-FP7 project CASSTING (number 601148). Quentin Menet was supported by a grant of FRIA. Christel Baier was partly supported by the German Research Council (DFG) through the collaborative research centre 912 Highly- Adaptive Energy-Efficient Computing (HAEC) and the cluster of excellence Centre for Advancing Electronics Dresden (cfAED), the EU-FP7 project MEALS (295261) and by the EU and the State Saxony through the ESF young researcher groups IMData (100098198) and SREX (100111037).

## References

[ACD91] Rajeev Alur, Costas Courcoubetis, and David L. Dill. Model-checking for probabilistic real-time systems. In *Proc. 18th International Colloquium on Automata, Languages and Programming (ICALP'91)*, volume 510 of *Lecture Notes in Computer Science*, pages 115–126. Springer, 1991.

[ACD92] Rajeev Alur, Costas Courcoubetis, and David L. Dill. Verifying automata specifications of probabilistic real-time systems. In *Proc. REX Workshop on Real-Time: Theory in Practice*, volume 600 of *Lecture Notes in Computer Science*, pages 28–44. Springer, 1992.

[AD90] Rajeev Alur and David L. Dill. Automata for modeling real-time systems. In *Proc. 17th International Colloquium on Automata, Languages and Programming (ICALP'90)*, volume 443 of *Lecture Notes in Computer Science*, pages 322–335. Springer, 1990.

[AD94] Rajeev Alur and David L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.

[AL02] Luca Aceto and François Laroussinie. Is your model-checker on time? On the complexity of model-checking for timed modal logics. *Journal of Logic and Algebraic Programming*, 52–53:7–51, 2002.

[ALP01] Rajeev Alur, Salvatore La Torre, and George J. Pappas. Optimal paths in weighted timed automata. In *Proc. 4th International Workshop on Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *Lecture Notes in Computer Science*, pages 49–62. Springer, 2001.

[AMPS98] Eugene Asarin, Oded Maler, Amir Pnueli, and Joseph Sifakis. Controller synthesis for timed automata. In *Proc. IFAC Symposium on System Structure and Control*, pages 469–474. Elsevier Science, 1998.

[ASSB00] Adnan Aziz, Kumud Sanwal, Vigyan Singhal, and Robert K. Brayton. Model-checking continuous-time Markov chains. *ACM Transactions on Computational Logic*, 1(1):162–170, 2000.

[BBB+07] Christel Baier, Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Marcus Größer. Probabilistic and topological semantics for timed automata. In *Proc. 27th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'07)*, volume 4855 of *Lecture Notes in Computer Science*, pages 179–191. Springer, 2007.

[BBB+08] Christel Baier, Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Marcus Größer. Almost-sure model checking of infinite paths in one-clock timed automata. In *Proc. 23rd Annual Symposium on Logic in Computer Science (LICS'08)*, pages 217–226. IEEE Computer Society Press, 2008.

[BBBM08] Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Nicolas Markey. Quantitative model-checking of one-clock timed automata under probabilistic semantics. In *Proc. 5th International Conference on Quantitative Evaluation of Systems (QEST'08)*. IEEE Computer Society Press, 2008.

[BBJM12] Patricia Bouyer, Thomas Brihaye, Marcin Jurdzinski, and Quentin Menet. Almost-sure model-checking of reactive timed automata. In *Proc. 9th International Conference on Quantitative Evaluation of Systems (QEST'12)*, pages 138–147. IEEE Computer Society Press, 2012.

[BDE+14] Christel Baier, Marcus Daum, Benjamin Engel, Hermann Härtig, Joachim Klein, Sascha Klüppelholz, Steffen Märcker, Hendrik Tews, and Völp Marcus. Locks: Picking key methods for a scalable quantitative analysis. *Journal of Computer and System Sciences*, 81(1):258–287, 2015.

[BDL+06] Gerd Behrmann, Alexandre David, Kim G. Larsen, John Håkansson, Paul Pettersson, Wang Yi, and Martijn Hendriks. Uppaal 4.0. In *Proc. 3rd International Conference on Quantitative Evaluation of Systems (QEST'06)*, pages 125–126. IEEE Computer Society Press, 2006.

[BDM+98] Marius Bozga, Conrado Daws, Oded Maler, Alfredo Olivero, Stavros Tripakis, and Sergio Yovine. Kronos: a model-checking tool for real-time systems. In *Proc. 10th International Conference on Computer Aided Verification (CAV'98)*, volume 1427 of *Lecture Notes in Computer Science*, pages 546–550. Springer, 1998.

[BF09] Patricia Bouyer and Vojtěch Forejt. Reachability in stochastic timed games. In *Proc. 36th International Colloquium on Automata, Languages and Programming (ICALP'09)*, volume 5556 of *Lecture Notes in Computer Science*, pages 103–114. Springer, 2009.

[BFH+01] Gerd Behrmann, Ansgar Fehnker, Thomas Hune, Kim G. Larsen, Paul Pettersson, Judi Romijn, and Frits Vaandrager. Minimum-cost reachability for priced timed automata. In *Proc. 4th International Workshop on Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *Lecture Notes in Computer Science*, pages 147–161. Springer, 2001.

[BFLM11] Patricia Bouyer, Uli Fahrenberg, Kim G. Larsen, and Nicolas Markey. Quantitative analysis of real-time systems using priced timed automata. *Communication of the ACM*, 54(9):78–87, 2011.

[BHHK03] Christel Baier, Boudewijn Haverkort, Holger Hermanns, and Joost-Pieter Katoen. Model-checking algorithms for continuous-time Markov chains. *IEEE Transactions on Software Engineering*, 29(7):524–541, 2003.

[Bil95] Patrick Billingsley. *Probability and measure*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, 3rd edition, 1995.

[BK98] Christel Baier and Marta Z. Kwiatkowska. On the verification of qualitative properties of probabilistic processes under fairness constraints. *Information Processing Letters*, 66(2):71–79, 1998.

[BS12] Nathalie Bertrand and Sven Schewe. Playing optimally on timed automata with random delays. In *Proc. 10th International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS'12)*, volume 7595 of *Lecture Notes in Computer Science*, pages 43–58. Springer, 2012.

[BvdSHV03] Henrik C. Bohnenkamp, Peter van der Stok, Holger Hermanns, and Frits W. Vaandrager. Cost-optimization of the IPv4 Zeroconf protocol. In *Proc. International Conference on Dependable Systems and Networks (DSN'03)*, pages 531–540. IEEE Computer Society Press, 2003.

[CHKM11] Taolue Chen, Tingting Han, Joost-Pieter Katoen, and Alexandru Mereacre. Model-checking of continuous-time Markov chains against timed automata specifications. *Logical Methods in Computer Science*, 7(1:12):1–34, 2011.

[CHR02] Franck Cassez, Thomas A. Henzinger, and Jean-François Raskin. A comparison of control problems for timed and hybrid systems. In *Proc. 5th International Workshop on Hybrid Systems: Computation and Control (HSCC'02)*, volume 2289 of *Lecture Notes in Computer Science*, pages 134–148. Springer, 2002.

[CLRS09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms*. MIT Press, Cambridge, MA, third edition, 2009.

[CMP92] E. Chang, Z. Manna, and A. Pnueli. Characterization of temporal property classes. In *Proc. 19th Int. Coll. Automata, Languages, and Programming (ICALP'92), Vienna, Austria, July 1992*, volume 623 of *Lecture Notes in Computer Science*, pages 474–486. Springer, 1992.

[DDR04] Martin De Wulf, Laurent Doyen, and Jean-François Raskin. Almost ASAP semantics: From timed models to timed implementations. In *Proc. 7th International Workshop on Hybrid Systems: Computation and Control (HSCC'04)*, volume 2993 of *Lecture Notes in Computer Science*, pages 296–310. Springer, 2004.

[DHS09] Susanna Donatelli, Serge Haddad, and Jeremy Sproston. Model checking timed and stochastic properties with $\mathrm{CSL}^{\mathrm{TA}}$. *IEEE Transactions on Software Engineering*, 35(2):224–240, 2009.

[DP03] Josée Desharnais and Prakash Panangaden. Continuous stochastic logic characterizes bisimulation of continuous-time Markov processes. *Journal of Logic and Algebraic Programming*, 56:99–115, 2003.

[GHJ97] Vineet Gupta, Thomas A. Henzinger, and Radha Jagadeesan. Robust timed automata. In *Proc. International Workshop on Hybrid and Real-Time Systems (HART'97)*, volume 1201 of *Lecture Notes in Computer Science*, pages 331–345. Springer, 1997.

[GThW02] Erich Grdel, Wolfgang Thomas, and Thomas Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*, volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002.

[KNP11] Marta Kwiatkowska, Gethin Norman, and David Parker. PRISM 4.0: verification of probabilistic real-time systems. In *Proc. 23rd International Conference on Computer Aided Verification (CAV'11)*, volume 6806 of *Lecture Notes in Computer Science*, pages 585–591. Springer, 2011.

[KNSS00] Marta Z. Kwiatkowska, Gethin Norman, Roberto Segala, and Jeremy Sproston. Verifying quantitative properties of continuous probabilistic timed automata. In *Proc. 11th International Conference on Concurrency Theory (CONCUR'00)*, volume 1877 of *Lecture Notes in Computer Science*, pages 123–137. Springer, 2000.

[KNSS02] Marta Z. Kwiatkowska, Gethin Norman, Roberto Segala, and Jeremy Sproston. Automatic verification of real-time systems with discrete probability distributions. *Theoretical Computer Science*, 282(1):101–150, 2002.

[KSK76] John G. Kemeny, J. Laurie Snell, and Anthony W. Knapp. *Denumerable Markov Chains*. Graduate Texts in Mathematics. Springer, 1976.

[LMS04] François Laroussinie, Nicolas Markey, and Philippe Schnoebelen. Model checking timed automata with one or two clocks. In *Proc. 15th International Conference on Concurrency Theory (CONCUR'04)*, volume 3170 of *Lecture Notes in Computer Science*, pages 387–401. Springer, 2004.

[Mar11] Nicolas Markey. Robustness in real-time systems. In *Proc. 6th IEEE International Symposium on Industrial Systems (SIES'11)*, pages 28–34. IEEE Computer Society Press, 2011.

[Mun00] James R. Munkres. *Topology*. Prentice Hall, 2nd edition, 2000.

[Oxt57] John C. Oxtoby. The Banach-Mazur game and Banach category theorem. *Annals of Mathematical Studies*, 39:159–163, 1957. Contributions to the Theory of Games, volume 3.

[Pnu77] Amir Pnueli. The temporal logic of programs. In *Proc. 18th Annual Symposium on Foundations of Computer Science (FOCS'77)*, pages 46–57. IEEE Computer Society Press, 1977.

[Pnu83] Amir Pnueli. On the extremely fair treatment of probabilistic algorithms. In *Proc. 15th Ann. Symp. Theory of Computing (STOC'83)*, pages 278–290. ACM Press, 1983.

[PZ93] Amir Pnueli and Lenore D. Zuck. Probabilistic verification. *Information and Computation*, 103(1):1–29, 1993.

[San13] Ocan Sankur. *Robustness in Timed Automata: Analysis, Synthesis, Implementation*. PhD thesis, École Normale Supérieure de Cachan, Cachan, France, 2013.

[SBM11] Ocan Sankur, Patricia Bouyer, and Nicolas Markey. Shrinking timed automata. In *Proc. 31st Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'11)*, volume 13 of *Leibniz International Proceedings in Informatics*, pages 90–102. Leibniz-Zentrum für Informatik, 2011.

[Sto03] Mariëlle Stoelinga. Fun with FireWire: A comparative study of formal verification methods applied to the IEEE 1394 root contention protocol. *Formal Aspects of Computing*, 14(3):328–337, 2003.

[Var85] Moshe Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *Proc. 26th Annual Symposium on Foundations of Computer Science (FOCS'85)*, pages 327–338. IEEE Computer Society Press, 1985.

[VV06] Daniele Varacca and Hagen Völzer. Temporal logics and model checking for fairly correct systems. In *Proc. 21st Annual Symposium on Logic in Computer Science (LICS'06)*, pages 389–398. IEEE Computer Society Press, 2006.

[VW94] Moshe Y. Vardi and Pierre Wolper. Reasoning about infinite computations. *Information and Computation*, 115(1):1–37, 1994.

[ZJNH11] Lijun Zhang, David N. Jansen, Flemming Nielson, and Holger Hermanns. Automata-based CSL model checking. In *Proc. 38th International Colloquium on Automata, Languages and Programming (ICALP'11)*, volume 6756 of *Lecture Notes in Computer Science*, pages 271–282. Springer, 2011.

In this technical appendix, statements in boxes refer to statements given in the core of the paper, and whose proofs were postponed to the appendix. All other statements are local to the appendix.

## Appendix A. Details for Section 3

**Proposition 3.2.** *For every state $s$ of $\mathcal{A}$, $\mathbb{P}_{\mathcal{A}}$ is a probability measure over $(\mathsf{Runs}(\mathcal{A}, s), \Omega^s_{\mathcal{A}})$.*

*Proof.* We first recall a basic property in measure theory [KSK76].

**Proposition A.1.** *Let $\nu$ be a non-negative additive set function defined on some set space $\mathcal{F}$ such that for every $A \in \mathcal{F}$, $\nu(A) < \infty$. The three following properties are equivalent:*

(1) *$\nu$ is $\sigma$-additive,*
(2) *for every sequence $(A_n)_n$ of elements of $\mathcal{F}$ such that $A_0 \subseteq A_1 \subseteq A_2 \subseteq \cdots$ and $A = \bigcup_n A_n \in \mathcal{F}$, $\lim_n \nu(A_n) = \nu(A)$,*
(3) *for every sequence $(B_n)_n$ of elements of $\mathcal{F}$ such that $B_0 \supseteq B_1 \supseteq B_2 \supseteq \cdots$ and $\bigcap_n B_n = \emptyset$, $\lim_n \nu(B_n) = 0$.*

For every $n \in \mathbb{N}$, we write $\mathcal{F}_n(s)$ for the ring[14] generated by the set of (basic) cylinders from $s$ of length $n$, *i.e.*, all $\mathsf{Cyl}(\pi_{\mathcal{C}}(s, e_1 \dots e_n))$. The elements of $\mathcal{F}_n(s)$ are thus finite unions of basic cylinders of length $n$. We then define

$$\mathcal{F}(s) = \bigcup_n \mathcal{F}_n(s)$$

**Lemma A.2.** *For every $n$, $\mathbb{P}_{\mathcal{A}}$ is a probability measure on $\mathcal{F}_n(s)$.*

*Proof.* First, by induction on $n$, it is not difficult to prove that for every $n \in \mathbb{N}$,

$$\sum_{(e_1,\dots,e_n)} \mathbb{P}_{\mathcal{A}}(\pi(s, e_1 \dots e_n)) = \mathbb{P}_{\mathcal{A}}(\pi(s)) = 1 \tag{A.1}$$

We fix $n \in \mathbb{N}$. $\mathbb{P}_{\mathcal{A}}$ is obviously additive, non-negative and finite over $\mathcal{F}_n(s)$. Take a sequence $(A_i)_i$ of elements of $\mathcal{F}_n(s)$ such that $A_0 \subseteq A_1 \subseteq A_2 \subseteq \cdots$ and $A = \bigcup_i A_i \in \mathcal{F}_n(s)$. There are finitely many distinct sequences of edges of length $n$. Hence, intersecting each of the $A_i$'s with each of the symbolic paths $\pi(s, e_1 \dots e_n)$ of length $n$, we can assume w.l.o.g. that each $A_i$ is a single constrained finite symbolic path.

Let $e_1 \dots e_n$ be the sequence of edges underlying all constrained symbolic paths $A_i$, and write $\mathcal{C}_i$ for the tightest constraint defining $A_i$ (*i.e.*, $A_i = \pi_{\mathcal{C}_i}(s, e_1 \dots e_n)$). We have that $\mathcal{C}_i \subseteq \mathcal{C}_{i+1}$, and $(\mathcal{C}_i)_i$ converges to $\mathcal{C}$, which corresponds to the constraint associated with $A$. We can write, if $\mathbb{1}_\alpha$ is the characteristic function of set $\alpha$, that:

$$\lim_i \mathbb{P}_{\mathcal{A}}(A_i) = \lim_i \int_{\tau_1 \in I(s,e_1)} p_{s+\tau_1}(e_1) \int_{\tau_2 \in I(s_{\tau_1}, e_2)} p_{s_{\tau_1}+\tau_2}(e_2) \cdots$$
$$\int_{\tau_n \in I(s_{\tau_1 \cdots \tau_{n-1}}, e_n)} p_{s_{\tau_1 \cdots \tau_{n-1}}+\tau_n}(e_n)\, \mathbb{1}_{\mathcal{C}_i}(\tau_1, \dots, \tau_n)\, \mathrm{d}\mu_{s_{\tau_1 \cdots \tau_{n-1}}}(\tau_n) \cdots \mathrm{d}\mu_s(\tau_1)$$

[14] A *ring* $R \subseteq 2^S$ is such that $\emptyset \in R$, $R$ is closed by finite union and by complement.

$$= \int_{\tau_1 \in I(s,e_1)} p_{s+\tau_1}(e_1) \int_{\tau_2 \in I(s_{\tau_1},e_2)} p_{s_{\tau_1}+\tau_2}(e_2) \cdots$$
$$\int_{\tau_n \in I(s_{\tau_1\cdots\tau_{n-1}},e_n)} p_{s_{\tau_1\cdots\tau_{n-1}}+\tau_n}(e_n) \left( \lim_i \mathbb{1}_{\mathcal{C}_i}(\tau_1,\ldots,\tau_n) \right) \mathrm{d}\mu_{s_{\tau_1\cdots\tau_{n-1}}}(\tau_n) \cdots \mathrm{d}\mu_s(\tau_1)$$

(by dominated convergence and equation (A.1))

$$= \int_{\tau_1 \in I(s,e_1)} p_{s+\tau_1}(e_1) \int_{\tau_2 \in I(s_{\tau_1},e_2)} p_{s_{\tau_1}+\tau_2}(e_2) \cdots$$
$$\int_{\tau_n \in I(s_{\tau_1\cdots\tau_{n-1}},e_n)} p_{s_{\tau_1\cdots\tau_{n-1}}+\tau_n}(e_n)\, \mathbb{1}_{\mathcal{C}}(\tau_1,\ldots,\tau_n)\, \mathrm{d}\mu_{s_{\tau_1\cdots\tau_{n-1}}}(\tau_n) \cdots \mathrm{d}\mu_s(\tau_1)$$

$$= \mathbb{P}_{\mathcal{A}}(A)$$

This shows that $\mathbb{P}_{\mathcal{A}}$ is a measure on $\mathcal{F}_n(s)$, for all $n \in \mathbb{N}$. It is moreover a probability measure since $\mathbb{P}_{\mathcal{A}}(\mathcal{F}_n(s)) = \mathbb{P}_{\mathcal{A}}(\pi(s)) = 1$. □

**Lemma A.3.** *$\mathbb{P}_{\mathcal{A}}$ is a probability measure on $\mathcal{F}(s)$.*

*Proof.* Obviously $\mathbb{P}_{\mathcal{A}}$ is non-negative on $\mathcal{F}(s)$, additive (because $\mathcal{F}_n(s) \subseteq \mathcal{F}_{n+1}(s)$ for every $n \in \mathbb{N}$) and finite over $\mathcal{F}(s)$. It remains to prove that it is $\sigma$-additive. For this, we use Proposition A.1, and consider a sequence $(B_n)_n$ of sets in $\mathcal{F}(s)$ such that $B_0 \supseteq B_1 \supseteq B_2 \supseteq \cdots$ and $\bigcap_n B_n = \emptyset$. W.l.o.g. we assume that for every $n$, $B_n \in \mathcal{F}_n(s)$. We want to prove that $\lim_n \mathbb{P}_{\mathcal{A}}(B_n) = 0$. Applying a reasoning similar to that of [KSK76, Lemmas 2.1, 2.2, 2.3], it is sufficient, thanks to Lemma A.2, to do the proof when $B_n$ is some $\mathsf{Cyl}(\pi_n)$ where $\pi_n$ is a finite (constrained) symbolic path of length $n$. We write $\mathcal{C}_n$ for the tightest constraint over variables $(\tau_i)_{i\leq n}$ corresponding to $\pi_n$. We define $p_i$ the constraint from $\mathbb{R}_+^{i+1}$ onto the $i$ first components (thus in $\mathbb{R}_+^i$). Note that this projection is continuous (for the product topologies). In $\pi_n$, if $i < n$, the $i$ first variables are constrained by $\mathcal{C}_n^i = p_i(\mathcal{C}_n^{i+1})$. Moreover, for every $i \leq n$, we have that

$$\mathcal{C}_{n+1}^i \subseteq \mathcal{C}_n^i \quad \text{and} \quad \mathcal{C}_n^i \subseteq \mathcal{C}_n^{i-1}$$

Fix some $i$, the sequence $(\mathcal{C}_n^i)_n$ is nested, hence converges to $\mathcal{C}^i$, and $\mathcal{C}^i \subseteq \mathcal{C}^{i-1}$. By continuity of the projection over the $i$ first components, we have that $\mathcal{C}^i = p_i(\mathcal{C}^{i+1})$. If none of the $\mathcal{C}^i$ is empty, we can thus construct an element in $\bigcap_i \mathcal{C}^i$ as follows: we take some $\tau_1$ satisfying the constraint $\mathcal{C}^1$; we have that $\mathcal{C}^1 = p_1(\mathcal{C}^2)$ (and $\mathcal{C}^2$ is a constraint over $\tau_1$ and $\tau_2$), hence there exists $\tau_2$ such that $(\tau_1, \tau_2)$ satisfies $\mathcal{C}^2$ (while $\tau_1$ still satisfies $\mathcal{C}^1$); we do the same step-by-step for all $\tau_i$ and construct a sequence $(\tau_i)_i$ which satisfies all constraints $\mathcal{C}^i$. This sequence corresponds to a run in $\bigcap_i \mathsf{Cyl}(\pi_i)$. As we assumed at the beginning of the paragraph that $\bigcap_i \mathsf{Cyl}(\pi_i) = \emptyset$, it thus means that there exists some $i \in \mathbb{N}$ such that $\mathcal{C}^i = \emptyset$.

We will use the fact that $\mathcal{C}^i = \bigcap_{n\geq i} \mathcal{C}_n^i$ is empty to prove that $\lim_n \mathbb{P}_{\mathcal{A}}(\pi_n) = 0$. We write, still with the notation that $\mathbb{1}_\alpha$ is the characteristic function of set $\alpha$:

$$\mathbb{P}_{\mathcal{A}}(\mathsf{Cyl}(\pi_n)) = \int_{\tau_1 \in I(s,e_1)} p_{s+\tau_1}(e_1) \int_{\tau_2 \in I(s_{\tau_1},e_2)} p_{s_{\tau_1}+\tau_2}(e_2) \cdots$$
$$\int_{\tau_n \in I(s_{\tau_1\cdots\tau_{n-1}},e_n)} p_{s_{\tau_1\cdots\tau_{n-1}}+\tau_n}(e_n)\, \mathbb{1}_{\mathcal{C}_n}(\tau_1,\ldots,\tau_n)\, \mathrm{d}\mu_{s_{\tau_1\cdots\tau_{n-1}}}(\tau_n) \cdots \mathrm{d}\mu_s(\tau_1)$$

$$\leq \int_{\tau_1 \in I(s,e_1)} \int_{\tau_2 \in I(s_{\tau_1},e_2)} \cdots \int_{\tau_i \in I(s_{\tau_1\cdots\tau_{i-1}},e_i)} \mathbb{1}_{\mathcal{C}_n^i}(\tau_1,\ldots,\tau_i)\,\mathrm{d}\mu_{s_{\tau_1\cdots\tau_{i-1}}}(\tau_i)\cdots\mathrm{d}\mu_s(\tau_1)$$

Applying the dominated convergence theorem, we get that:

$$\begin{aligned}\lim_n \mathbb{P}_{\mathcal{A}}(\mathsf{Cyl}(\pi_n)) &= \int_{\ldots}\int_{\ldots}\cdots\int_{\ldots}\Big(\lim_n \mathbb{1}_{\mathcal{C}_n^i}(\tau_1,\ldots,\tau_i)\Big)\,\mathrm{d}\mu_{s_{\tau_1\cdots\tau_{i-1}}}(\tau_i)\cdots\mathrm{d}\mu_s(\tau_1)\\ &= 0\end{aligned}$$

This concludes the proof that $\mathbb{P}_{\mathcal{A}}$ is $\sigma$-additive on $\mathcal{F}(s)$, and thus the proof that $\mathbb{P}_{\mathcal{A}}$ is a probability measure on $\mathcal{F}(s)$. □

We conclude the proof using the following classical measure extension theorem:

**Theorem A.4** (Carathéodory's extension theorem)**.** *Let $S$ be a set, and $\nu$ a $\sigma$-finite measure defined on a ring $R \subseteq 2^S$. Then, $\nu$ can be extended in a unique manner to the $\sigma$-algebra generated by $R$.*

We apply Theorem A.4 to the set $S = \mathsf{Runs}(\mathcal{A}, s)$, $R = \mathcal{F}(s)$, and $\nu = \mathbb{P}_{\mathcal{A}}$ which is a $\sigma$-finite measure on $\mathcal{F}(s)$. Hence, there is a unique extension of $\mathbb{P}_{\mathcal{A}}$ on $\Omega^s_{\mathcal{A}}$, the $\sigma$-algebra generated by the cylinders, which is a probability measure. □

---

**Lemma 3.4.** *Let $\langle \mathcal{A}, \mu^{\mathcal{A}}, w^{\mathcal{A}}\rangle$ be a stochastic timed automaton, and let $\langle \mathsf{R}(\mathcal{A}), \mu^{\mathsf{R}(\mathcal{A})}, w^{\mathsf{R}(\mathcal{A})}\rangle$ be the corresponding stochastic timed region automaton as defined above. Then, for every set $S$ of runs in $\mathcal{A}$ we have: $S \in \Omega^s_{\mathcal{A}}$ iff $\iota(S) \in \Omega^{\iota(s)}_{\mathsf{R}(\mathcal{A})}$, and in this case $\mathbb{P}_{\mathcal{A}}(S) = \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(S))$.*

*Proof.* It is sufficient to prove that the measures coincide on finite constrained paths, since it implies that they agree on cylinders and by uniqueness of the extension on any measurable set of infinite runs.

In this proof we will denote transitions of $\mathcal{A}$ by $e_i$ and transition in $\mathsf{R}(\mathcal{A})$ by $f_i$. We prove that $\mathbb{P}_{\mathcal{A}}$ and $\mathbb{P}_{\mathsf{R}(\mathcal{A})}$ coincide on finite paths by induction on the length $n$ of constrained symbolic paths. When $n = 0$, this is obvious as, for every $(\ell, \nu)$, there is a single state $((\ell, r), \nu)$ in $\mathsf{R}(\mathcal{A})$ such that $\nu \in r$, and in that case, $\iota(\pi((\ell,\nu))) = \{\pi(((\ell, r), \nu))\}$. We assume the induction hypothesis holds for all constrained paths of length strictly smaller than $n$.

We will use the following notations (this will be technical, but rather simple): given $s$ a state, we recall that $s + t$ is the state reached from $s$ after a delay $t$, $[s]$ is the region to which $s$ belongs. If $q$ is a state of the region automaton, we write $n_q$ for the number of edges enabled without delay in $q$ in $\mathsf{R}(\mathcal{A})$ (or equivalently in $\mathcal{A}$). If transition $e_1$ can be taken from $q$ without delay, $e_1(q)$ denotes the single image region reached after firing $e_1$ from $q$, and we write $q \models f_1$ if $f_1$ is the unique transition with guard checking that we are in $q$ and corresponding to $e_1$ in $\mathsf{R}(\mathcal{A})$.

Let $\pi = \pi_{\mathcal{C}}(s, e_1, \ldots, e_n)$ be a constrained symbolic path in $\mathcal{A}$. Constraint $\mathcal{C}$ is on $n$ variables $\tau_1 \cdots \tau_n$. We will denote $\mathcal{C}_t$ the constraint obtained from $\mathcal{C}$ by replacing $\tau_1$ by $t$.

$$
\begin{aligned}
\mathbb{P}_{\mathcal{A}}(\pi) &= \int_{t\in I(s,e_1)} p^{\mathcal{A}}_{s+t}(e_1)\,\mathbb{P}_{\mathcal{A}}(\pi_{\mathcal{C}_t}(s_t,e_2\dots e_n))\,\mathrm{d}\mu^{\mathcal{A}}_s(t)\\
&= \int_{t\in I(s,e_1)} p^{\mathcal{A}}_{s+t}(e_1)\,\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(\pi_{\mathcal{C}_t}(s_t,e_2\dots e_n)))\,\mathrm{d}\mu^{\mathcal{A}}_s(t) \quad \text{by induction hypothesis}\\
&= \int_{t\in I(s,e_1)} p^{\mathcal{A}}_{s+t}(e_1) \sum_{\pi'\in\iota(\pi(s_t,e_2\dots e_n))} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\pi'_{\mathcal{C}_t})\,\mathrm{d}\mu^{\mathcal{A}}_s(t)\\
&= \sum_q \int_{\substack{t\in I(s,e_1)\\ s+t\in q}} p^{\mathcal{A}}_{s+t}(e_1) \sum_{\pi'\in\iota(\pi(s_t,e_2\dots e_n))} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\pi'_{\mathcal{C}_t})\,\mathrm{d}\mu^{\mathcal{A}}_s(t)\\
&= \sum_q \int_{\substack{t\in I(s,e_1)\\ s+t\in q}} p^{\mathcal{A}}_{s+t}(e_1) \sum_{(f_2,\dots f_n)\in\iota(e_1(q),e_2,\dots,e_n)} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\pi_{\mathcal{C}_t}(\iota(s_t),f_2\dots f_n))\,\mathrm{d}\mu^{\mathcal{A}}_s(t)\\
&= \sum_q \int_{\substack{t\in I(\iota(s),f_1)\\ s+t\in q\\ q\models f_1\\ [s]\xrightarrow{f_1}e_1(q)}} p^{\mathsf{R}(\mathcal{A})}_{\iota(s)+t}(f_1) \sum_{(f_2,\dots,f_n)\in\iota(e_1(q),e_2,\dots,e_n)} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\pi_{\mathcal{C}_t}(\iota(s)_t,f_2,\dots,f_n))\,\mathrm{d}\mu^{\mathsf{R}(\mathcal{A})}_{\iota(s)}(t)\\
&\qquad\qquad \text{by hypothesis on the measures and weights}\\
&= \sum_{\substack{q\models f_1\\ [s]\xrightarrow{f_1}e_1(q)\\ (f_2,\dots,f_n)\in\iota(e_1(q),e_2,\dots,e_n)}} \int_{t\in I(\iota(s),f_1)} p^{\mathsf{R}(\mathcal{A})}_{\iota(s)+t}(f_1)\,\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\pi_{\mathcal{C}_t}(\iota(s)_t,f_2,\dots,f_n))\,\mathrm{d}\mu^{\mathsf{R}(\mathcal{A})}_{\iota(s)}(t)\\
&= \sum_{\substack{q\models f_1\\ [s]\xrightarrow{f_1}e_1(q)\\ (f_2,\dots,f_n)\in\iota(e_1(q),e_2,\dots,e_n)}} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\pi_{\mathcal{C}}(\iota(s),f_1,\dots,f_n))\\
&= \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(\pi))
\end{aligned}
$$

where $(f_2,\dots,f_n)\in\iota(e_1(q),e_2,\dots,e_n)$ iff $(f_2,\dots,f_n)$ is a finite sequence of transitions corresponding to $(e_2,\dots,e_n)$ and which starts in $(e_1(q))$ (this is a state of $\mathsf{R}(\mathcal{A})$). □

---

**Lemma 3.5.** *$\omega$-regular properties and properties given as* LTL *formulas are measurable. Timed properties given as specification Büchi or Muller timed automata are measurable.*

*Proof.* It is sufficient to do the proof in the case of specifications given as deterministic timed automata. Indeed, it covers also the case of $\omega$-regular and LTL-properties, since they can be turned into a deterministic untimed Muller automaton.

Let $\langle\mathcal{A},\mu,w\rangle$ be a stochastic timed automaton, and $\mathcal{B}$ a specification automaton (that is, a deterministic complete timed automaton). We prove that the set of runs in $\mathcal{A}$ that are accepted by $\mathcal{B}$ is measurable (for the probability measure defined by $\langle\mathcal{A},\mu,w\rangle$). To do so, we consider the product timed automaton $\mathcal{A}\ltimes\mathcal{B}$ (see definition on page 27). Let $\mathsf{R}(\mathcal{A}\ltimes\mathcal{B})$ be its (untimed) region automaton, and $\mathcal{F}$ the accepting condition naturally derived from the one of $\mathcal{B}$. The set of paths in $\mathsf{R}(\mathcal{A}\ltimes\mathcal{B})$ satisfying $\mathcal{F}$ is a Boolean combination of

cylinders $\mathsf{Cyl}(\mathbf{s}_0, \mathbf{e}_1 \dots \mathbf{e}_n)$. Indeed, since $\mathcal{F}$ is an $\omega$-regular condition and seeing $\mathsf{R}(\mathcal{A} \ltimes \mathcal{B})$ as a finite Markov chain (with arbitrary probabilities), this is a consequence of the proof of measurability of $\omega$-regular properties [Var85]. For a fixed finite path $\mathbf{s}_0, \mathbf{e}_1 \dots \mathbf{e}_n$ in $\mathsf{R}(\mathcal{A} \ltimes \mathcal{B})$, we write

$$H(\mathbf{s}_0, \mathbf{e}_1 \dots \mathbf{e}_n) = \{\varrho \in \mathsf{Runs}(\mathcal{A}, s_0) \mid \iota(\varrho^{\mathcal{B}}) \in \mathsf{Cyl}(\pi(\mathbf{s}_0, \mathbf{e}_1 \dots \mathbf{e}_n))\} \ .$$

Roughly speaking, $H(\mathbf{s}_0, \mathbf{e}_1 \dots \mathbf{e}_n)$ is the set of all runs in $\mathcal{A}$ whose natural projection in $\mathsf{R}(\mathcal{A} \ltimes \mathcal{B})$ belongs to $\mathsf{Cyl}(\pi(\mathbf{s}_0, \mathbf{e}_1 \dots \mathbf{e}_n))$. One can be convinced that $H(\mathbf{s}_0, \mathbf{e}_1 \dots \mathbf{e}_n)$ consists of a finite union of cylinders generated by constrained symbolic paths in $\mathcal{A}$. Hence the set of runs in $\mathcal{A}$ satisfying the specification $\mathcal{B}$ can be written as a Boolean combination of cylinders generated by constrained symbolic paths, and is therefore measurable. □

## Appendix B. Details for Section 4

For Definition 4.11 to properly define a topological space, we prove that the intersection of two basic open sets is still a basic open set. This is the object of the following result, whose proof requires several technical intermediary lemmas.

**Lemma B.1.** *Let $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(s, e_1, \dots, e_n)$ and $\pi_{\mathcal{C}'} = \pi_{\mathcal{C}'}(s, e_1, \dots, e_n)$ be two basic open sets of same length. Then $\pi_{\mathcal{C}} \cap \pi_{\mathcal{C}'}$ is an open set.*

For the next lemmas, let us fix $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(s, e_1, \dots, e_n)$ and $\pi_{\mathcal{C}'} = \pi_{\mathcal{C}'}(s, e_1, \dots, e_n)$ be two constrained symbolic paths of same length, where $\mathcal{C}$ and $\mathcal{C}'$ are Borel-measurable. For all $i \leq n$, write $\mathcal{C}_i$ (resp. $\mathcal{C}'_i$) for the projection of $\mathcal{C}$ (resp. $\mathcal{C}'$) on the $i$ first coordinates. Write also $\pi_{\mathcal{C}_i} = \pi_{\mathcal{C}_i}(s, e_1 \dots e_i)$, $\pi_{\mathcal{C}'_i} = \pi_{\mathcal{C}'_i}(s, e_1 \dots e_i)$, and $\pi_i = \pi(s, e_1 \dots e_i)$.

**Lemma B.2.** *Assume $\pi_{\mathcal{C}} \subseteq \pi_{\mathcal{C}'}$ and $\dim(\mathsf{Pol}(\pi_{\mathcal{C}})) = \dim(\mathsf{Pol}(\pi_{\mathcal{C}'}))$. Then for all $i \leq n$, $\dim(\mathsf{Pol}(\pi_{\mathcal{C}_i})) = \dim(\mathsf{Pol}(\pi_{\mathcal{C}'_i}))$*

*Proof.* Assume there exists an index $i \leq n$ such that $\dim(\mathsf{Pol}(\pi_{\mathcal{C}_i})) < \dim(\mathsf{Pol}(\pi_{\mathcal{C}'_i}))$. As $\dim(\mathsf{Pol}(\pi_{\mathcal{C}})) = \dim(\mathsf{Pol}(\pi_{\mathcal{C}'}))$ there must be an index $j$, such that $\mathsf{Pol}(\pi_{\mathcal{C}})$ gains some dimension in the $j$-th direction, whereas $\mathsf{Pol}(\pi_{\mathcal{C}'})$ does not. But this is not possible since $\pi_{\mathcal{C}} \subseteq \pi_{\mathcal{C}'}$ and therefore $\mathsf{Pol}(\pi_{\mathcal{C}}) \subseteq \mathsf{Pol}(\pi_{\mathcal{C}'})$. □

From this basic result, we get the following corollaries.

**Corollary B.3.** *If $\mathsf{Pol}(\pi_{\mathcal{C}})$ is open in $\mathsf{Pol}(\pi)$, then for all $i \leq n$, $\dim(\mathsf{Pol}(\pi_{\mathcal{C}_i})) = \dim(\mathsf{Pol}(\pi_i))$.*

*Proof.* As $\mathsf{Pol}(\pi_{\mathcal{C}})$ is open in $\mathsf{Pol}(\pi)$, there exists an open set $O$ of $\mathbb{R}^n$ such that $\mathsf{Pol}(\pi_{\mathcal{C}}) = O \cap \mathsf{Pol}(\pi)$. This implies that $\dim(\mathsf{Pol}(\pi_{\mathcal{C}})) = \dim(\mathsf{Pol}(\pi))$.[15] Applying Lemma B.2 to $\pi_{\mathcal{C}}$ and $\pi$ yields the expected result. □

[15] We use here the following general topology result: if $X$ is a convex set and $O$ an open set in $\mathbb{R}^n$ such that $X \cap O \neq \emptyset$, then $\dim(X) = \dim(X \cap O)$.

**Corollary B.4.** *Assume $\pi_{\mathcal{C}'}$ is a non-empty open set of $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$ and $\pi_{\mathcal{C}} \subseteq \pi_{\mathcal{C}'}$ and $\mathsf{Pol}(\pi_{\mathcal{C}})$ is open in $\mathsf{Pol}(\pi)$, then $\pi_{\mathcal{C}}$ is thick (that is, $\pi_{\mathcal{C}}$ is a non-empty basic open set of $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$).*

*Proof.* By Corollary B.3 applied to both $\pi_{\mathcal{C}}$ and $\pi$, and $\pi_{\mathcal{C}'}$ and $\pi$, we get for every $1 \leq i \leq n$:

$$\dim(\mathsf{Pol}(\pi_{\mathcal{C}_i})) = \dim(\mathsf{Pol}(\pi_{\mathcal{C}'_i})) = \dim(\mathsf{Pol}(\pi_i)) \,.$$

As $\pi_{\mathcal{C}'}$ is a basic open set it also holds that for every $1 \leq i \leq n$:

$$\dim(\mathsf{Pol}(\pi_{\mathcal{C}'_i})) = \dim(\bigcup_e \mathsf{Pol}(\pi_{\mathcal{C}'_{i-1}}(s, e_1 \dots e_{i-1}e))) \,.$$

By containment of $\pi_{\mathcal{C}_{i-1}}(s, e_1 \dots e_{i-1}e)$ into $\pi_{\mathcal{C}'_{i-1}}(s, e_1 \dots e_{i-1}e)$, we get that

$$\dim(\bigcup_e \mathsf{Pol}(\pi_{\mathcal{C}'_{i-1}}(s, e_1 \dots e_{i-1}e))) \geq \dim(\bigcup_e \mathsf{Pol}(\pi_{\mathcal{C}_{i-1}}(s, e_1 \dots e_{i-1}e))) \,.$$

This shows that $\pi_{\mathcal{C}}$ is thick:

$$\dim(\mathsf{Pol}(\pi_{\mathcal{C}_i})) \geq \dim(\bigcup_e \mathsf{Pol}(\pi_{\mathcal{C}_{i-1}}(s, e_1 \dots e_{i-1}e))) \,.$$

□

We can now come to the proof of Lemma B.1.

*of Lemma B.1.* Let us denote in this proof $\mathcal{C}'' = \mathcal{C} \cap \mathcal{C}'$, and $\pi$ the unconstrained symbolic path $\pi(s, e_1, \dots, e_n)$. Write $\pi_{\mathcal{C}''}$ for $\pi_{\mathcal{C}} \cap \pi_{\mathcal{C}'} = \pi_{\mathcal{C}''}(s, e_1, \dots, e_n)$. If $\pi_{\mathcal{C}''}$ is empty, we are done since the empty set is an open set. We therefore assume that $\pi_{\mathcal{C}''}$ is non-empty.

We first show that $\mathsf{Pol}(\pi_{\mathcal{C}} \cap \pi_{\mathcal{C}'})$ is open in $\mathsf{Pol}(\pi)$, which is the second condition for $\pi_{\mathcal{C}} \cap \pi_{\mathcal{C}'}$ to be an open set. We have that $\mathsf{Pol}(\pi_{\mathcal{C}''}) = \mathsf{Pol}(\pi_{\mathcal{C}}) \cap \mathsf{Pol}(\pi_{\mathcal{C}'})$. By assumption both $\mathsf{Pol}(\pi_{\mathcal{C}})$ and $\mathsf{Pol}(\pi_{\mathcal{C}'})$ are open in $\mathsf{Pol}(\pi)$, hence their intersection too.

The fact that $\pi_{\mathcal{C}''}$ is thick is a consequence of Corollary B.4. We conclude that $\pi_{\mathcal{C}''}$ is an open set for our topology. □

---

**Proposition 4.12.** *For every state $s$ of $\mathcal{A}$, the topological space $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$ is a Baire space.*

*Proof.* To prove that $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$ is a Baire space, we prove that every non-empty basic open set in $\mathcal{T}^s_{\mathcal{A}}$ is not meagre. Let $\mathsf{Cyl}(\pi_{\mathcal{C}}(s, e_1 \dots e_n))$ be a basic open set. Using Banach-Mazur games (see page 15 or [Oxt57]), we prove that $\mathsf{Cyl}(\pi_{\mathcal{C}}(s, e_1 \dots e_n))$ is not meagre by proving that Player 2 does not have a winning strategy for the Banach-Mazur game played with basic open sets and where the goal set is $C = \mathsf{Cyl}(\pi_{\mathcal{C}}(s, e_1 \dots e_n))$.

Player 1 starts by choosing a set $B_1 = \mathsf{Cyl}(\pi_{\mathcal{C}}(s, e_1 \dots e_n))$. Then Player 2 picks some basic open set $B_2 = \mathsf{Cyl}(\pi_{\mathcal{C}^2}(s, e_1 \dots e_n \dots e_{n_1}))$ such that $B_1 \supseteq B_2$.

Let us now explain how Player 1 can build her move in order to avoid to reach the empty set. Since $B_2$ is an open set, we have that (i) $\pi_{\mathcal{C}^2}$ is thick and (ii) $\mathsf{Pol}(\pi_{\mathcal{C}^2}(s, e_1 \dots e_{n_1}))$ is open in $\mathsf{Pol}(\pi(s, e_1 \dots e_{n_1})) \subseteq \mathbb{R}^{n_1}_+$. The topology on $\mathsf{Pol}(\pi(s, e_1 \dots e_{n_1}))$ is induced from a distance, hence there exists a closed, bounded and convex set denoted $K_1$ such that $\mathring{K}_1 \neq \emptyset$ and $K_1 \subseteq \mathsf{Pol}(\pi_{\mathcal{C}^2}(s, e_1 \dots e_{n_1}))$. Let $\mathcal{D}^1$ be the constraint associated with $K_1$; clearly cylinder $\mathsf{Cyl}(\pi_{\mathcal{D}^1}(s, e_1 \dots e_{n_1}))$ is included in $B_2$. Let $O$ be an open set included in $K_1$ and $\mathcal{C}^3$ be

the constraint associated with $O$. It is Borel-measurable since it is open. Applying Corollary B.4, we know that $\pi_{\mathcal{C}^3}(s, e_1 \dots e_{n_1})$ is thick. Hence clearly enough, $\mathsf{Cyl}(\pi_{\mathcal{C}^3}(s, e_1 \dots e_{n_1}))$ is an open set. Player 1's move will be to take $B_3 = \mathsf{Cyl}(\pi_{\mathcal{C}^3}(s, e_1 \dots e_{n_1}))$. By iterating this process, we define a strategy for Player 1 which satisfies:

$$B_1 \supseteq B_2 \supseteq \mathsf{Cyl}(\pi_{\mathcal{D}^1}) \supseteq B_3 \supseteq B_4 \supseteq \mathsf{Cyl}(\pi_{\mathcal{D}^2}) \supseteq \ldots \supseteq B_{2i-1} \supseteq B_{2i} \supseteq \mathsf{Cyl}(\pi_{\mathcal{D}^i}) \supseteq \cdots$$

where for each $i$, $K_i = \mathsf{Pol}(\pi_{\mathcal{D}^i})$ is a closed and bounded subset of $\mathsf{Pol}(\pi(e_1, \dots, e_{n_i})) \subseteq \mathbb{R}_+^{n_i}$ (where the $n_i$'s form a non-decreasing sequence of $\mathbb{N}$). We then have that:

$$\bigcap_{i=1}^{\infty} B_i \;=\; \bigcap_{i=1}^{\infty} \mathsf{Cyl}(\pi_{\mathcal{D}^i})\,.$$

We would like to guarantee that the above intersection is non-empty. This is not completely straightforward since the polyhedra $K_i = \mathsf{Pol}(\pi_{\mathcal{D}^i})$ belong to different powers of $\mathbb{R}_+$. We distinguish between two cases:

- either the sequence $(n_i)_{i\geq 1}$ diverges to $+\infty$. In that case, we will embed $\bigcap_{i=1}^{\infty} K_i$ into a compact set of $\mathbb{R}_+^{\mathbb{N}}$. We first define

$$\widetilde{K_j} = \mathsf{Proj}_{\{n_{j-1}+1,\dots,n_j\}} K_j \quad \text{and} \quad \widetilde{K} = \prod_{j\geq 1} \widetilde{K_j}\,,$$

  where $\mathsf{Proj}_I(K_J)$ for $I \subseteq \{1 \cdots n_j\}$ is the natural projection from $\mathbb{R}_+^{n_j}$ to the coordinates specified by $I$. Note that $\widetilde{K_j}$ is a compact set, since it is the projection of a compact set. Each $K_i$ can naturally be embedded in $\widetilde{K}$ by considering the sets $K_i'$ defined by

$$K_i' = K_i \times \prod_{j>i} \widetilde{K_j}\,.$$

  The decomposition is illustrated on Figure 12. The $K_i'$'s form a nested chain of closed sets of $\widetilde{K}$. By Tychonoff's theorem, $\widetilde{K}$ is compact. Hence we can ensure that $\bigcap_{i=1}^{\infty} K_i'$ is non-empty (Heine-Borel theorem). Take a sequence $(\tau_j)_{j\geq 1}$ in $\bigcap_{i=1}^{\infty} K_i'$. Each subsequence $(\tau_j)_{1\leq j\leq n_i}$ straightforwardly belongs to $K_i$. Hence, the run $s \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} s_2 \dots$ is in $\bigcap_{i=1}^{\infty} B_i$, which completes the proof in this case.

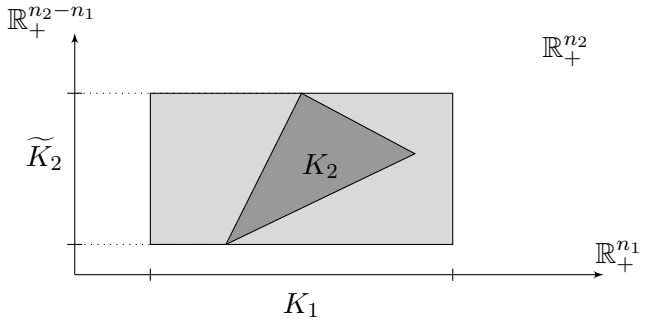



Figure 12: The decomposition of the $K_i$'s.

- either the sequence $(n_i)_{i\geq 1}$ is upper bounded. In that case, we embed $\bigcap_{i=1}^{\infty} K_i$ into a compact set of $\mathbb{R}_+^N$ where $N = \lim_{i\to+\infty} n_i$. We let the details to the reader, as they are very similar to (and easier than) the previous case.

□

---

**Lemma 4.14.** *In the topological space* $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$*, a finite symbolic path* $\pi$ *defines a basic open set if and only if there exist an open constraint* $\mathcal{C}$ *of* $\mathbb{R}^n$ *and thick edges* $e_1, \dots, e_n$ *such that* $\pi = \pi_{\mathcal{C}}(s, e_1 \dots e_n)$*.*

*Proof.* First notice that if $\mathcal{C}$ is an open constraint of $\mathbb{R}^n$, then $\mathcal{C}$ is Borel-measurable, and $\mathsf{Pol}(\pi_{\mathcal{C}}(s, e_1 \dots e_n)) = \mathsf{Pol}(\pi(s, e_1 \dots e_n)) \cap \mathcal{C}$ is open in $\mathsf{Pol}(\pi(s, e_1 \dots e_n))$. Furthermore, applying Corollary B.4, $\pi_{\mathcal{C}}(s, e_1 \dots e_n)$ is thick if $\pi(s, e_1 \dots e_n)$ is thick as well.

Now, assume that $\pi = \pi_{\mathcal{C}}(s, e_1 \dots e_n)$ is a basic open set. This means in particular that $\mathsf{Pol}(\pi)$ is open in $\mathsf{Pol}(\pi(s, e_1 \dots e_n))$: there exists an open set $\delta$ of $\mathbb{R}^n$ such that $\mathsf{Pol}(\pi) = \mathsf{Pol}(\pi(s, e_1 \dots e_n)) \cap \delta$. As it is open, $\delta$ is Borel-measurable, and we get that $\mathsf{Pol}(\pi) = \mathsf{Pol}(\pi_\delta(s, e_1 \dots e_n))$, and therefore $\pi = \pi_\delta(s, e_1 \dots e_n)$, which is the expected result. □

---

**Lemma 4.17.** *Let* $\iota : \mathsf{Runs}(\mathcal{A}, s) \to \mathsf{Runs}(\mathsf{R}(\mathcal{A}), \iota(s))$ *be the projection of runs in* $\mathcal{A}$ *onto the region automaton* $\mathsf{R}(\mathcal{A})$*. Then* $\iota$ *is continuous, and for every non-empty open set* $O \in \mathcal{T}^s_{\mathcal{A}}$*,* $\widehat{\iota(\overset{\circ}{O})} \neq \emptyset$*.*

*Proof. We first prove that* $\iota$ *is continuous.* Let $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(\iota(s), f_1 \dots f_n)$ be a symbolic path in $\mathsf{R}(\mathcal{A})$ such that $\mathsf{Cyl}(\pi_{\mathcal{C}})$ is a basic open set of $(\mathsf{Runs}(\iota(s), \mathsf{R}(\mathcal{A})), \mathcal{T}^{\iota(s)}_{\mathsf{R}(\mathcal{A})})$. We need to prove that $\iota^{-1}(\mathsf{Cyl}(\pi_{\mathcal{C}}))$ is an open set of $\mathcal{T}^s_{\mathcal{A}}$. One can easily be convinced that $\iota^{-1}(\mathsf{Cyl}(\pi_{\mathcal{C}})) = \mathsf{Cyl}(\iota^{-1}(\pi_{\mathcal{C}}))$. Thus proving the continuity of $\iota$ consists in proving that $\iota^{-1}(\pi_{\mathcal{C}})$ is a thick finite symbolic path whose polyhedron is open in its ambient space.

First notice that there are unique edges $e_1, \dots, e_n$ such that $\iota^{-1}(\pi_{\mathcal{C}}) \subseteq \pi(s, e_1 \dots e_n)$, we can then set $\pi' \stackrel{\text{def}}{=} \iota^{-1}(\pi_{\mathcal{C}})$. Then obviously, $\mathsf{Pol}(\pi_{\mathcal{C}}) = \mathsf{Pol}(\pi')$.

Let $\gamma$ be the tightest constraint which defines $\mathsf{Pol}(\pi(\iota(s), f_1 \dots f_n))$. We have for every $i \leq n$:

$$\begin{cases} \iota^{-1}(\pi_{\mathcal{C}_i}(\iota(s), f_1 \dots f_i)) & = & \pi_{\mathcal{C}_i \wedge \gamma_i}(s, e_1 \dots e_i) \\ \bigcup_f \iota^{-1}(\pi_{\mathcal{C}_i}(\iota(s), f_1 \dots f_{i-1} f)) & = & \bigcup_e \pi_{\mathcal{C}_{i-1} \wedge \gamma_{i-1}}(s, e_1 \dots e_{i-1} e) \end{cases}$$

where $\mathcal{C}_i$ and $\gamma_i$ are the projection of $\mathcal{C}$ and $\gamma$ on the $i$ first coordinates (in particular, $\gamma_i$ is the tightest constraint defining $\pi(\iota(s), f_1 \dots f_i)$ since this is in $\mathsf{R}(\mathcal{A})$). As $\pi_{\mathcal{C}}$ is thick, the two dimensions on the left are equal, and therefore so are the two dimensions on the right. We deduce that $\pi'$ is thick.

Now, as $\pi(\iota(s), f_1 \dots f_n)$ is thick (since $\pi_{\mathcal{C}}$ is thick), we can prove by induction on its length that there is some open constraint $\delta$ such that

$$\mathsf{Pol}(\pi(\iota(s), f_1 \dots f_n)) = \mathsf{Pol}(\pi_\delta(s, e_1 \dots e_n)) = \mathsf{Pol}(\pi(s, e_1 \dots e_n)) \cap \delta$$

Indeed:

- if $\delta_i$ is an open constraint such that $\mathsf{Pol}(\pi_{\delta_i}(s, e_1 \dots e_i)) = \mathsf{Pol}(\pi(\iota(s), f_1 \dots f_i))$, we have that $\dim(\mathsf{Pol}(\pi_{\delta_i}(s, e_1 \dots e_i e_{i+1}))) = \dim(\mathsf{Pol}(\pi(\iota(s), f_1 \dots f_i f_{i+1})))$ (by thickness of $\pi(\iota(s), f_1 \dots f_n)$);

- furthermore this value is either equal to $\dim(\mathsf{Pol}(\pi(\iota(s), f_1 \dots f_i)))$ or to $\dim(\mathsf{Pol}(\pi(\iota(s), f_1 \dots f_i))) + 1$;
- in the first case, we set $\delta_{i+1} = \delta_i$, whereas in the second case $\delta_{i+1}$ is obtained by adding to $\delta_i$ the open constraint derived from the last transition $f_{i+1}$ (which is then open);
- we can easily check that this concludes the induction.

As $\pi_{\mathcal{C}}$ is thick, $\mathsf{Pol}(\pi_{\mathcal{C}})$ is open in $\mathsf{Pol}(\pi(\iota(s), f_1 \dots f_n))$: there exists an open set $O$ of $\mathbb{R}^n$ such that $\mathsf{Pol}(\pi_{\mathcal{C}}) = \mathsf{Pol}(\pi(\iota(s), f_1 \dots f_n)) \cap O$.

We infer that $\mathsf{Pol}(\pi') = \mathsf{Pol}(\pi_{\mathcal{C}}) = \mathsf{Pol}(\pi(s, e_1 \dots e_n)) \cap \delta \cap O$, and $\delta \cap O$ is open in $\mathbb{R}^n$: $\mathsf{Pol}(\pi')$ is open in $\mathsf{Pol}(\pi(s, e_1 \dots e_n))$.

This concludes the proof: $\mathsf{Cyl}(\pi')$ is a basic open set in $(\mathsf{Runs}(\mathcal{A}, s), \mathcal{T}^s_{\mathcal{A}})$.

*We now prove that for every non-empty open set $O \in \mathcal{T}^s_{\mathcal{A}}$, $\widehat{\iota(O)} \neq \emptyset$.* Again it is sufficient to prove that for each basic open set $\mathsf{Cyl}(\pi_{\mathcal{C}})$ of $\mathcal{T}^s_{\mathcal{A}}$, $\iota(\mathsf{Cyl}(\pi_{\mathcal{C}}))$ contains a basic open set $\mathsf{Cyl}(\pi')$ of $\mathcal{T}^{\iota(s)}_{\mathsf{R}(\mathcal{A})}$, that is, there is a thick symbolic path $\pi'$ whose polyhedron is open in its ambient space and such that $\mathsf{Cyl}(\pi') \subseteq \iota(\mathsf{Cyl}(\pi_{\mathcal{C}}))$.

Let $\pi_{\mathcal{C}} = \pi_{\mathcal{C}}(s, e_1 \dots e_n)$ be a constrained symbolic path such that $\mathsf{Cyl}(\pi_{\mathcal{C}})$ is a basic open set of $(\mathsf{Runs}(s, \mathcal{A}), \mathcal{T}^s_{\mathcal{A}})$. We have that

$$\iota(\pi_{\mathcal{C}}) = \bigcup_{f_1, \dots, f_n} \pi_{\mathcal{C}}(\iota(s), f_1 \dots f_n)$$

where the (finite) union is taken over all sequences of edges $f_1, \dots, f_n$ corresponding to $e_1, \dots, e_n$. There exist thus edges $f_1, \dots, f_n$ such that

$$\dim(\mathsf{Pol}(\pi_{\mathcal{C}})) = \dim\Big(\mathsf{Pol}\big(\pi_{\mathcal{C}}(\iota(s), f_1 \dots f_n)\big)\Big)$$

and we write $\pi'_{\mathcal{C}} = \pi_{\mathcal{C}}(\iota(s), f_1 \dots f_n)$. We will prove that $\pi'_{\mathcal{C}}$ is an open set. Note that as $\mathsf{Pol}(\pi_{\mathcal{C}})$ is open in $\mathsf{Pol}(\pi(s, e_1 \dots e_n))$, we can assume w.l.o.g. that $\mathcal{C}$ defines an open set of $\mathbb{R}^n$. Hence $\pi'_{\mathcal{C}}$ is open in $\pi(\iota(s), f_1 \dots f_n)$. Assume that it is thin. Then, there exists some $i$ such that

$$\dim\Big(\mathsf{Pol}\big(\pi_{\mathcal{C}'_i}(\iota(s), f_1 \dots f_i)\big)\Big) < \dim\Big(\bigcup_f \mathsf{Pol}\big(\pi_{\mathcal{C}'_{i-1}}(\iota(s), f_1 \dots f_{i-1}, f)\big)\Big)$$

where $\mathcal{C}'_i$ corresponds to the projection on the $i$ first coordinates of the tightest constraint defining $\pi'_{\mathcal{C}}$. Moreover, as $\mathsf{Pol}(\pi'_{\mathcal{C}}) \subseteq \mathsf{Pol}\big(\pi_{\mathcal{C}}\big)$ and $\dim(\mathsf{Pol}(\pi'_{\mathcal{C}})) = \dim(\mathsf{Pol}(\pi_{\mathcal{C}}))$, applying Lemma B.2, we get that for all $i$'s, $\dim(\mathsf{Pol}(\pi'_{\mathcal{C}'_i})) = \dim(\mathsf{Pol}(\pi_{\mathcal{C}_i}))$. Furthermore, $\bigcup_f \mathsf{Pol}\big(\pi_{\mathcal{C}'_{i-1}}(\iota(s), f_1 \dots f_{i-1} f)\big) \subseteq \bigcup_e \mathsf{Pol}\big(\pi_{\mathcal{C}_{i-1}}(s, e_1 \dots e_{i-1} e)\big)$ (this is a property of the region automaton). Finally, we get that

$$\dim\Big(\mathsf{Pol}\big(\pi_{\mathcal{C}_i}\big)\Big) < \dim\Big(\bigcup_e \mathsf{Pol}\big(\pi_{\mathcal{C}_{i-1}}(s, e_1 \dots e_{i-1} e)\big)\Big)$$

which contradicts the hypothesis that $\pi$ is thick. We deduce that $\mathsf{Cyl}(\pi')$ is a basic open set of $(\mathsf{Runs}(\mathsf{R}(\mathcal{A}), \iota(s)), \mathcal{T}^{\iota(s)}_{\mathsf{R}(\mathcal{A})})$, hence the result. □

**Proposition 4.19.** *Let $s$ be a state of $\mathcal{A}$, and $P$ be a timed property over $\mathsf{AP}$. Then,*

$$\mathcal{A}, s \not\approx_{\mathcal{T}} P \;\Leftrightarrow\; \mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathcal{T}} P\,.$$

*Proof.* We prove both implications using characterisation of meagre sets by Banach-Mazur games and Lemma 4.17. To play this game, we choose as basis all open sets.

Assume Player 2 has a winning strategy in $\mathcal{A}$ to avoid $[\![P]\!]_{\mathcal{A},s}$. We will show that Player 2 also has a winning strategy in $\mathsf{R}(\mathcal{A})$ to avoid $[\![P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$. Before starting the simulation, we recall that $[\![P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)} = \iota([\![P]\!]_{\mathcal{A},s})$.

The first move of Player 1 in $\mathsf{R}(\mathcal{A})$ is some open set $B_1$ (in $\mathcal{T}^{\iota(s)}_{\mathsf{R}(\mathcal{A})}$). That move can be transported in $\mathcal{A}$: thanks to Lemma 4.17, $B'_1 \stackrel{\text{def}}{=} \iota^{-1}(B_1)$ is a legal move of the game in $\mathcal{A}$. Then, Player 2 plays according to her strategy in $\mathcal{A}$ with move $B'_2$. This move cannot directly be transported to $\mathsf{R}(\mathcal{A})$ (since $\iota(B'_2)$ may not be an open set), but thanks to Lemma 4.17, there is a non-empty open set $B_2$ such that $B_2 \subseteq \iota(B'_2)$. We continue the simulation that way. Finally we get that $\bigcap_i B_i \subseteq \bigcap_i \iota(B'_i) = \iota(\bigcap_i B'_i)$. As Player 2 plays with a winning strategy in $\mathcal{A}$, we get that $\bigcap_i B'_i \cap [\![P]\!]_{\mathcal{A},s} = \emptyset$, which implies that $\bigcap_i B_i \cap [\![P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)} = \emptyset$.

On the contrary, assume that Player 2 has a winning strategy in $\mathsf{R}(\mathcal{A})$ to avoid $[\![P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$. We will show that Player 2 also has a winning strategy in $\mathcal{A}$ to avoid $[\![P]\!]_{\mathcal{A},s}$. Assume that Player 1 plays $\pi_\gamma(s, e_1 \dots e_n)$, then applying Lemma 4.17, Player 2 can play as if it was $\pi_\gamma(\iota(s), f_1 \dots f_n)$ in $\mathsf{R}(\mathcal{A})$ for some $f_1, \dots, f_n$. The game then plays as in $\mathsf{R}(\mathcal{A})$, and all moves are legal thanks to Lemma 4.17. We conclude that this strategy avoids $[\![P]\!]_{\mathcal{A},s}$ as well, which concludes the proof. □

## Appendix C. Details for Section 5

**Lemma 5.3.** *Let $\pi(s, e_1 \dots e_n)$ be a symbolic path of $\mathsf{R}(\mathcal{A})$. Assuming $\mathsf{Pol}(\pi(s, e_1 \dots e_n)) \neq \emptyset$ and letting $q$ be the target region of $\pi(s, e_1 \dots e_{n-1})$,*

$$\dim(\mathsf{Pol}(\pi(s, e_1 \dots e_n))) = \dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1}))) + \dim(I(q, e_n)).$$

*Proof.* Let $q$ be the target region of $\pi(s, e_1 \dots e_{n-1})$. There are two possible cases:

- $\dim(I(q, e_n)) = 0$: for every $s' \in q$, there is a unique delay $\tau_n(s')$ such that $s' \xrightarrow{\tau_n(s'), e_n}$. Also, for all delays $\tau_1, \dots, \tau_{n-1}$ such that $s \xrightarrow{\tau_1, e_1} \dots \xrightarrow{\tau_{n-1}, e_{n-1}}$ there is a unique $s' \in q$ such that $s \xrightarrow{\tau_1, e_1} \dots \xrightarrow{\tau_{n-1}, e_{n-1}} s'$. We can therefore define the function $g$ with $g(\tau_1, \dots, \tau_{n-1}) = \tau_n(s')$. We then write:

$$\begin{aligned}
\mathsf{Pol}(\pi(s, e_1 \dots e_n)) &= \{(\tau_1, \dots, \tau_n) \mid \tau_n = \tau_n(s') \text{ where } s \xrightarrow{\tau_1, e_1} \dots \xrightarrow{\tau_{n-1}, e_{n-1}} s'\} \\
&= \{(\tau_1, \dots, \tau_n) \mid (\tau_1, \dots, \tau_{n-1}) \in \mathsf{Pol}(\pi(s, e_1 \dots e_{n-1})) \\
&\qquad\qquad \text{and } \tau_n = g(\tau_1, \dots, \tau_{n-1})\} \\
&= \{(\tau_1, \dots, \tau_{n-1}, g(\tau_1, \dots, \tau_{n-1})) \mid (\tau_1, \dots, \tau_{n-1}) \in \mathsf{Pol}(\pi(s, e_1 \dots e_{n-1}))\}
\end{aligned}$$

The second equality holds because $\pi(s, e_1 \dots e_n)$ is a symbolic path in the region automaton $\mathsf{R}(\mathcal{A})$. We deduce that $\dim(\mathsf{Pol}(\pi(s, e_1 \dots e_n))) = \dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1})))$, which is the expected value.

- $\dim(I(s', e_n)) = 1$ for every $s' \in q$: there is non-punctual open interval $(\mu_n(s'), \nu_n(s'))$ such that $s' \xrightarrow{\tau_n, e_n}$ iff $\tau_n \in (\mu_n(s'), \nu_n(s'))$. We can then rewrite $\mathsf{Pol}(\pi(s, e_1, \dots, e_n))$ as follows:

$$\begin{aligned}\mathsf{Pol}(\pi(s, e_1 \dots e_n)) \quad = \quad & \{(\tau_1, \dots, \tau_{n-1}, \tau_n) \mid (\tau_1, \dots, \tau_{n-1}) \in \mathsf{Pol}(\pi(s, e_1 \dots e_{n-1})) \\ & \qquad \text{and } \tau_n \in g(\tau_1, \dots, \tau_{n-1})\}\end{aligned}$$

where $g(\tau_1, \dots, \tau_{n-1})$ defines an open interval. We then get

$$\dim(\mathsf{Pol}(\pi(s, e_1 \dots e_n))) = \dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1}))) + 1.$$

□

**Proposition 5.4.** *Let $\pi = \pi(s, e_1 \dots e_n)$ be a symbolic path in $\mathsf{R}(\mathcal{A})$. Then, $\pi$ is thin in $\mathsf{R}(\mathcal{A})$ iff there exists $1 \leq i \leq n$ such that $e_i$ is thin.*

*Proof.* The proof is done by induction on the length $n$ of $\pi$.

The case $n = 0$ is obvious since $\pi(s)$ is thick and $\pi(s)$ surely contains no thin edge. Assume $n > 0$ and the result holds for any $0 \leq j \leq n-1$, and let $\pi = \pi(s, e_1 \dots e_n)$ be a symbolic path of length $n$. For every $1 \leq j \leq n$, write $\pi_{\leq j}$ for $\pi(s, e_1 \dots e_j)$.

- Let us first assume that $\pi$ is thin in $\mathsf{R}(\mathcal{A})$. In case there exists $j < n$ such that $\pi_{\leq j}$ is thin, then we are done by applying the induction hypothesis to $\pi_{\leq j}$. We therefore assume that $\pi_{\leq j}$ is thick for every $j < n$. Let $k = \dim(\mathsf{Pol}(\pi_{\leq n-1}))$. Applying Lemma 5.3, $\dim(\mathsf{Pol}(\pi)) \in \{k, k+1\}$, and if $\dim(\mathsf{Pol}(\pi)) = k+1$, then $\pi$ would be thick ($k+1$ is the maximal dimension of any possible $\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1} e))$). We thus get that $\dim(\mathsf{Pol}(\pi)) = k$, and there exists $e$ such that $\dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1} e))) = k+1$ (this witnesses the fact that $\pi$ is thin). By Lemma 5.3, we therefore infer that $\dim(I(q, e_n)) = 0$ whereas $\dim(I(q, e)) = 1$ where $q$ is the target region of $\pi(s, e_1 \dots e_{n-1})$: edge $e_n$ is thin, which concludes the left-to-right implication.
- Assume now that $\pi$ is thick. By definition of thickness, for every $j \leq n$, $\pi_{\leq j}$ is thick. By induction hypothesis, all edges $e_1, \dots, e_{n-1}$ are then thick. Assume towards a contradiction that $e_n$ is thin. Let $q$ be the source-region of $e_n$, there exists an edge $e$ such that for every $s' \in q$, $\dim(I(s', e_n)) < \dim(I(s', e))$. Now we know that for every $s'$ with $s \xrightarrow{\tau_1, e_1} \dots \xrightarrow{\tau_{n-1}, e_{n-1}} s'$ for some $\tau_1, \dots, \tau_{n-1}$, $s' \in q$ (this is a property of region automata). Applying Lemma 5.3, we get that

$$\dim(\mathsf{Pol}(\pi(s, e_1 \dots e_n))) = \dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1}))) + \dim(I(q, e_n))$$

whereas

$$\dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1} e))) = \dim(\mathsf{Pol}(\pi(s, e_1 \dots e_{n-1}))) + \dim(I(q, e)).$$

We deduce that $\pi$ is thin, contradicting the assumption. This concludes the proof of the right-to-left implication. □

## Appendix D. Details for Section 6

### D.1. **Safety properties.**

**Proposition 6.1.** *Consider a finite symbolic path $\pi = \pi(s, e_1 \dots e_n)$ in $\mathsf{R}(\mathcal{A})$. Then, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi)) > 0$ iff $\pi$ is thick. Equivalently, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi)) = 0$ iff $\pi$ is thin.*

*Proof.* We assume that the probability distributions in $\mathsf{R}(\mathcal{A})$ are those used in Lemma 3.4, and we write $\mu$ for the distributions over delays.

We first prove that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi)) > 0$ implies that $\pi$ is thick. Towards a contradiction, assume that $\pi$ is thin. Following Proposition 5.4, there exists $1 \leq i \leq n$ such that $e_i$ is thin. Let $q$ be the target set of $\pi(s, e_1 \dots e_{i-1})$: $\dim(I(q, e_i)) < \dim(I(q))$. By hypothesis on the measures $\mu$ (condition $(\star)$, *cf* page 11), for every $s' \in q$, $\mu_{s'}(I(s', e_i)) = 0$. Hence, for every $s' \in q$, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi(s', e_i \dots e_n))) = 0$. This implies that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi)) = 0$.

Assume now that $\pi$ is thick. For every $1 \leq i \leq n$, we write $q_i$ for the target region of $\pi(s, e_1 \dots e_i)$. Following Proposition 5.4, for every $1 \leq i \leq n$, $e_i$ is thick, which means that $\dim(I(q_i, e_{i+1})) = \dim(I(q_i))$. As in the first implication, by assumption $(\star)$ on the measure $\mu$, for every $s_i \in q_i$, $\mu_{s_i}(I(s_i, e_{i+1})) > 0$. We then use the definition of the probability inductively on suffixes of $\pi$ starting in some $s_i$ to obtain a sequence of integral computation over non negligible set of a positive function, hence $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi)) > 0$. □

---

**Theorem 6.2.** *Let $s$ be a state of $\mathcal{A}$, and $P$ be a(n untimed) safety property over $\mathsf{AP}$. Then the four following properties are equivalent:*

(a) $\mathcal{A}, s \not\approx_{\mathbb{P}} P$;
(b) $\mathcal{A}, s \not\approx_{\mathcal{T}} P$;
(c) *every infinite thick symbolic path $\pi$ from $\iota(s)$ in $\mathsf{R}(\mathcal{A})$ satisfies $P$;*
(d) *every infinite path $\pi$ from $\iota(s)$ in $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$ satisfies $P$;*
(e) $\mathsf{MC}(\mathcal{A}), \iota(s) \not\approx P$.

*Proof.* Equivalence between $(c)$ and $(d)$ is by construction of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$.

Thanks to Lemmas 3.4 and 4.19, to prove the equivalence between $(a)$ and $(b)$, it is equivalent (and hence sufficient) to prove that $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathcal{T}} P$ iff $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathbb{P}} P$. Since $P$ is a property over $\mathsf{AP}$, for every infinite symbolic path $\pi = \pi(s, e_1 \dots e_i \dots)$, either all realisations $\varrho \in \pi$ satisfy $P$, or none of them satisfies $P$. Now, using the fact that $P$ is a safety property, we can write $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}), \iota(s)}$ as a denumerable union of cylinders:

$$[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}), \iota(s)} = \bigcup_{i \in I} \mathsf{Cyl}(\pi_i)$$

where $\pi_i$ is a finite (unconstrained) symbolic path from $\iota(s)$, and $I$ is a denumerable set. Thus, $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathbb{P}} P$ is equivalent to $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\bigcup_{i \in I} \mathsf{Cyl}(\pi_i)) = 0$. Since $I$ is denumerable, we obtain $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathbb{P}} P$ iff for all $i \in I$, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi_i)) = 0$. By Proposition 6.1, we have that $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathbb{P}} P$ iff for all $i \in I$, $\pi_i$ is thin (this by-the-way proves equivalence between $(a)$ and $(c)$). Thus proving the theorem amounts to proving that $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}), \iota(s)}$ is meagre iff for all $i \in I$, $\pi_i$ is thin.

Let us assume first that for all $i \in I$, $\pi_i$ is thin. To prove that , $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$ is meagre we use a Banach-Mazur game and Theorem 4.3, playing with the set $\mathcal{B}$ of basic open sets of $\mathcal{T}^{\iota(s)}_{\mathsf{R}(\mathcal{A})}$. The objective of the game is set to be $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$. By hypothesis, $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)} = \bigcup_{i\in I} \mathsf{Cyl}(\pi_i)$ and all $\pi_i$ are thin. As every basic open set is of the form $\mathsf{Cyl}(\pi)$ with $\pi$ thick, it holds that for every $B \in \mathcal{B}$ such that $B \neq \mathsf{Runs}(\mathsf{R}(\mathcal{A}), \iota(s))$, we have $B \cap [\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)} = \emptyset$. Thus, if the first move of Player 1 is $\mathsf{Runs}(\mathsf{R}(\mathcal{A}), \iota(s))$, Player 2 picks some cylinder of a finite thick path. If the first move of Player 1 is a cylinder $\mathsf{Cyl}(\pi)$, then Player 2 just chooses the same set. Then, Player 2 wins the game by mimicking at each round the choices of Player 1, *i.e.*, whatever set $B_{2j-1}$ Player 1 chooses in the $j$-th round, Player 2 answers with the same choice $B_{2j} = B_{2j-1}$. For such a play we clearly have $\bigcap_{i=1}^{\infty} B_i \cap [\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)} = \emptyset$. Thus Player 2 has a winning strategy and Theorem 4.3 implies that $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$ is meagre.

Let us now prove the other implication. For a contradiction we assume that $\pi_i$ is thick for some $i \in I$. In particular $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$ would contain the open set $\mathsf{Cyl}(\pi_i)$, which is not meagre because our topological space is Baire (see Proposition 4.12). Since the notion of being meagre is closed under subset, the set $[\![\neg P]\!]_{\mathsf{R}(\mathcal{A}),\iota(s)}$ would not be meagre, which is a contradiction. □

## D.2. Details for the counter-example $\mathcal{A}_{\mathsf{unfair}}$ of Figure 9.

**Proposition D.1.** $\mathbb{P}_{\mathcal{A}_{\mathsf{unfair}}}\big(\pi(s_0, (e_3e_4e_5)^\omega)\big) > 0$, *where* $s_0 = (\ell_0, (0,0))$.

*Proof.* In this proof, we write $\mathbb{P}$ for $\mathbb{P}_{\mathcal{A}_{\mathsf{unfair}}}$, and we first show that given $0 < t_0 < 1$, $\mathbb{P}\big(\pi(s_{t_0}, (e_3e_4e_5)^\omega)\big) > 0$ where $s_{t_0} = (\ell_0, (0, t_0))$. Obviously, we have that

$$\mathbb{P}\big(\pi(s_{t_0}, (e_3e_4e_5)^\omega)\big) = \lim_{N\to+\infty} \mathbb{P}\big(\pi(s_{t_0}, (e_3e_4e_5)^N)\big)$$

where we write $\mathbb{P}\big(\pi(s_{t_0}, (e_3e_4e_5)^N)\big)$ for $\mathbb{P}\big(\mathsf{Cyl}(\pi(s_{t_0}, (e_3e_4e_5)^N))\big)$.

We now would like to express $\mathbb{P}(\pi(s_{t_0}, (e_3e_4e_5)^N))$ as a multiple integral. In order to take the leftmost loop, we need to choose a first delay ensuring that the valuation of the clock $y$ satisfies the guard $1 < y < 2$. The location $\ell_4$ is then reached with the clock valuation $(2 - t_0, 0)$. From there a second positive time delay has to be chosen in order to reach location $\ell_0$. We thus have that:

$$\begin{aligned}\mathbb{P}\big(\pi(s_{t_0}, (e_3e_4e_5)^N)\big) &= \frac{1}{2-t_0}\int_{\tau=1-t_0}^{2-t_0} \frac{1}{1-t_0}\int_{t_1=t_0}^{1} \mathbb{P}\big(\pi(s_1, (e_3e_4e_5)^{N-1})\big)\mathrm{d}t_1\mathrm{d}\tau \\ &= \frac{1}{2-t_0}\cdot\frac{1}{1-t_0}\int_{t_1=t_0}^{1} \mathbb{P}\big(\pi(s_1, (e_3e_4e_5)^{N-1})\big)\mathrm{d}t_1\end{aligned}$$

where $s_1 = (\ell_0, (0, t_1))$. By iterating this process, we obtain that:

$$\mathbb{P}\big(\pi(s_{t_0}, (e_3e_4e_5)^N)\big) = \frac{1}{2-t_0}.\frac{1}{1-t_0}\int_{t_1=t_0}^{1}\frac{1}{2-t_1}\cdot\frac{1}{1-t_1}\int_{t_2=t_1}^{1}\cdots\frac{1}{2-t_{N-1}}\cdot\frac{1}{1-t_{N-1}}\int_{t_N=t_{N-1}}^{1}\mathrm{d}t_N\ldots\mathrm{d}t_1.$$

We write

$$\gamma_i^N = \frac{1}{1-t_{i-1}}\int_{t_i=t_{i-1}}^{1}\frac{1}{2-t_i}\cdot\frac{1}{1-t_i}\int_{t_{i+1}=t_i}^{1}\cdots\frac{1}{2-t_{N-1}}\cdot\frac{1}{1-t_{N-1}}\int_{t_N=t_{N-1}}^{1}\mathrm{d}t_N\ldots\mathrm{d}t_i$$

and we can prove by a descending induction on $i$ (see Lemma D.2 below) that

$$\gamma_i^N \geq t_{i-1}.$$

Thus, we deduce that

$$\mathbb{P}\big(\pi(s_{t_0}, (e_3e_4e_5)^N)\big) = \frac{1}{2-t_0} \cdot \gamma_1^N \geq \frac{t_0}{2-t_0} > 0.$$

It remains to show that $\mathbb{P}\big(\pi(s_0, (e_3e_4e_5)^\omega)\big) > 0$, *i.e.* that the above result extends to the case $t_0 = 0$. Roughly speaking, after one loop, we will have $t_1 > 0$, hence we can use the above inequality from the second loop on by writing:

$$\begin{aligned}
\mathbb{P}(\pi(s_0, (e_3e_4e_5)^\omega)) &= \frac{1}{2-0} \cdot \frac{1}{1-0} . \int_0^1 \mathbb{P}(\pi(s_{t_1}, (e_3e_4e_5)^\omega))\, \mathrm{d}t_1 \\
&\geq \frac{1}{2} . \int_0^1 \frac{t_1}{2-t_1}\, \mathrm{d}t_1 \\
&\geq \frac{1}{2} . \int_0^1 \left(-1 + \frac{2}{2-t_1}\right) \mathrm{d}t_1 \\
&\geq \frac{1}{2} . \left[-t_1 - 2\log(2-t_1)\right]_{t_1=0}^1 \\
&\geq \log(2) - \frac{1}{2} > 0.
\end{aligned}$$

This concludes the proof. □

**Lemma D.2.** *If* $\gamma_i^N = \frac{1}{1-t_{i-1}} \int_{t_i=t_{i-1}}^1 \frac{1}{2-t_i} \cdot \frac{1}{1-t_i} \int_{t_{i+1}=t_i}^1 \cdots \frac{1}{2-t_{N-1}} \cdot \frac{1}{1-t_{N-1}} \int_{t_N=t_{N-1}}^1 \mathrm{d}t_N \ldots \mathrm{d}t_i$ *with* $0 < t_{i-1} < 1$, *then:*

$$\gamma_i^N \geq t_{i-1}.$$

*Proof.* The base case is when $i = N$. In that case,

$$\gamma_N^N = \frac{1}{1-t_{N-1}} \int_{t_N=t_{N-1}}^1 \mathrm{d}t_N = 1$$

which proves the desired property.

We assume we have proved the property for $i+1$, and want to prove it for $i$.

$$\begin{aligned}
\gamma_i^N &= \frac{1}{1-t_{i-1}} \int_{t_i=t_{i-1}}^1 \frac{1}{2-t_i} \cdot \gamma_{i+1}^N\, \mathrm{d}t_i \\
&\geq \frac{1}{1-t_{i-1}} \int_{t_i=t_{i-1}}^1 \frac{t_i}{2-t_i} \mathrm{d}t_i \qquad \text{(by i.h.)} \\
&\geq \frac{1}{1-t_{i-1}} \int_{t_i=t_{i-1}}^1 \left(-1 + \frac{2}{2-t_i}\right) \mathrm{d}t_i \\
&\geq \frac{1}{1-t_{i-1}} \left[-t_i - 2\log(2-t_i)\right]_{t_i=t_{i-1}}^1 \\
&\geq \frac{1}{1-t_{i-1}} \left(-1 + t_{i-1} + 2\log(2-t_{i-1})\right)
\end{aligned}$$

Now, when $0 \leq x \leq 1$, we know that $\log(1+x) \geq x - \frac{x^2}{2}$ (see Lemma D.3). Applying this inequality to $x = 1 - t_{i-1}$, we get the following inequality:

$$\begin{aligned} \gamma_i^N &\geq \frac{1}{1-t_{i-1}} \left(-1 + t_{i-1} + 2(1-t_{i-1}) - (1-t_{i-1})^2\right) \\ &\geq 1 - (1-t_{i-1}) = t_{i-1}. \end{aligned}$$

This concludes the inductive case. □

**Lemma D.3.** *Let $0 \leq x \leq 1$. Then $\log(1+x) \geq x - \frac{x^2}{2}$.*

*Proof.* Let $f(x) = \log(1+x)$ and $g(x) = x - \frac{x^2}{2}$. Since $f'(x) = \frac{1}{1+x}$ and $g'(x) = 1 - x$, we remark that for any $0 \leq x \leq 1$, we have $f'(x) \geq g'(x)$. Indeed, for any $x \geq 0$, $\frac{1}{1+x} \geq 1 - x$ if and only if $1 \geq 1 - x^2$. Since $f(0) = 0 = g(0)$, the result follows. □

D.3. **Extension to specification timed automata.** In this part, we aim at proving Theorem 6.9 below.

**Theorem 6.9.** *Let $s$ be a state of $\mathcal{A}$, and $\mathcal{B}$ be a specification Büchi or Muller timed automaton. Assuming $\mathbb{P}_{\mathcal{A}\ltimes\mathcal{B}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s) \models \mathsf{fair}) = 1$, the following holds:*

$$\mathcal{A}, s \not\approx_{\mathbb{P}} \mathcal{B} \;\Leftrightarrow\; \mathcal{A}, s \not\approx_{\mathcal{T}} \mathcal{B} \;\Leftrightarrow\; \S(\mathcal{A} \ltimes \mathcal{B}, \mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), P_{\mathcal{A}\ltimes\mathcal{B}})\,.$$

The proof of this theorem will require several lemmas that we present below. Before that, we define $\iota_{\mathcal{B}}$ the application (bijection) which assigns to every run $\varrho$ in $\mathcal{A}$ its unique image $\varrho^{\mathcal{B}}$ in $\mathcal{A} \ltimes \mathcal{B}$.

**Lemma D.4.** *Let $s$ be a state of $\mathcal{A}$, and $\mathcal{B}$ be a specification Büchi or Muller timed automaton. Assume measures and weights in $\mathcal{A} \ltimes \mathcal{B}$ are properly set. Then:*

$$\mathbb{P}_{\mathcal{A}}(s \models \mathcal{B}) = \mathbb{P}_{\mathcal{A}\ltimes\mathcal{B}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s) \models P_{\mathcal{A}\ltimes\mathcal{B}})\,.$$

*Proof.* This lemma is a consequence of the two following properties:

(1) for every measurable set $E$ of $\mathsf{Runs}(\mathcal{A}, s)$,

$$\mathbb{P}_{\mathcal{A}}(E) = \mathbb{P}_{\mathcal{A}\ltimes\mathcal{B}}\{\iota_{\mathcal{B}}(\varrho) \mid \varrho \in E\}$$

(2) for every run $\varrho$ in $\mathcal{A}$, $\varrho \models \mathcal{B}$ iff $\iota_{\mathcal{B}}(\varrho) \models P_{\mathcal{A}\ltimes\mathcal{B}}$.

The second item is a direct consequence of the definition of $P_{\mathcal{A}\ltimes\mathcal{B}}$. The proof for the first item is similar to that for region automata, that is, to the proof of Lemma 3.4. We therefore skip it. □

A similar result holds for the topological semantics:

**Lemma D.5.** *Let $\mathcal{A}$ be a timed automaton, let $s$ be a state of $\mathcal{A}$, and let $\mathcal{B}$ be a specification Büchi or Muller timed automaton. Then, for every set $S \subseteq \mathsf{Runs}(\mathcal{A}, s)$:*

$$S \text{ is large in } \mathcal{T}^s_{\mathcal{A}} \;\Leftrightarrow\; \{\varrho^{\mathcal{B}} \mid \varrho \in S\} \text{ is large in } \mathcal{T}^{\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s)}_{\mathcal{A}\ltimes\mathcal{B}}$$

*Proof.* The proof of this lemma is similar to that of Proposition 4.19, even though projection $\iota_{\mathcal{B}}$ might not be continuous. The next lemma (Lemma D.6) is sufficient to make the simulation between the two Banach-Mazur games (as in the proof of Proposition 4.19), and to prove the expected result. □

**Lemma D.6.**

(1) *If $O$ is a non-empty open set of $\mathsf{Runs}(\mathcal{A} \ltimes \mathcal{B}, \mathit{init}_{\mathcal{A}\ltimes\mathcal{B}}(s))$, then $\iota_{\mathcal{B}}^{-1}(O)$ has a non-empty interior.*

(2) *If $O$ is a non-empty open set of $\mathsf{Runs}(\mathcal{A}, s)$, then $\iota_{\mathcal{B}}(O)$ has a non-empty interior.*

*Proof.* We show the first property. Let $\pi$ be a basic open set of $\mathsf{Runs}(\mathcal{A} \ltimes \mathcal{B}, \mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s))$. Following the notations of Section 6.3 and applying Lemma 4.14, we can write the basic open set $\pi$ as $\pi_{\mathcal{C}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), e^1_{\mathsf{e}^1} \dots e^n_{\mathsf{e}^n})$, where $\mathcal{C}$ is an open constraint of $\mathbb{R}^n$. Note that as $\mathcal{B}$ is complete, none of the transitions $\mathsf{e}^i$ have equality constraints (otherwise $\pi$ would not be an open set). Also notice that all $e^i$'s are also thick (for the same reason). Let $\gamma$ be the constraint of $\mathbb{R}^n$ generated by the transitions $\mathsf{e}^1, \dots, \mathsf{e}^n$. Due to the previous remark, the interior of $\gamma$ is non-empty. Also, as $\mathcal{C} \wedge \gamma$ is non-empty, this implies that $\mathcal{C} \wedge \gamma$ has a non-empty interior. Now, we should just notice that $\iota_{\mathcal{B}}^{-1}(\pi) = \pi_{\mathcal{C}\wedge\gamma}(s, e^1 \dots e^n)$, which contains an open set, since $\mathcal{C} \wedge \gamma$ has a non-empty interior, and all the $e^i$'s are thick.

Fix now a non-empty open set $O$ in $\mathsf{Runs}(\mathcal{A}, s)$, and pick $\pi = \pi_{\mathcal{C}}(s, e^1 \dots e^n)$ a basic open set included in $O$. The image of $\pi$ by $\iota_{\mathcal{B}}$ can be decomposed as the following finite union over all sequences of edges in $\mathcal{B}$ of length $n$:

$$\iota_{\mathcal{B}}(\pi) = \bigcup_{(\mathsf{e}^1,\dots,\mathsf{e}^n)} \pi_{\mathcal{C}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), e^1_{\mathsf{e}^1} \dots e^n_{\mathsf{e}^n})$$

Also note that we don't lose any behaviour, in the sense that

$$\begin{aligned} \mathsf{Pol}(\pi) &= \mathsf{Pol}\left( \bigcup_{(\mathsf{e}^1,\dots,\mathsf{e}^n)} \pi_{\mathcal{C}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), e^1_{\mathsf{e}^1} \dots e^n_{\mathsf{e}^n}) \right) \\ &= \bigcup_{(\mathsf{e}^1,\dots,\mathsf{e}^n)} \mathsf{Pol}\left( \pi_{\mathcal{C}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), e^1_{\mathsf{e}^1} \dots e^n_{\mathsf{e}^n}) \right) \end{aligned}$$

In particular, there is some $(\mathsf{e}^1, \dots, \mathsf{e}^n)$ such that $\dim(\mathsf{Pol}\left(\pi_{\mathcal{C}_i}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), e^1_{\mathsf{e}^1} \dots e^i_{\mathsf{e}^i})\right)) = \dim(\mathsf{Pol}(\pi_{|\leq i}))$ for every $i$, where we consider here the projections over the $i$ first components. In particular, we can easily prove now that $\pi_{\mathcal{C}}(\mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s), e^1_{\mathsf{e}^1} \dots e^n_{\mathsf{e}^n})$ is a basic open set in $\mathsf{Runs}(\mathcal{A} \ltimes \mathcal{B}, \mathsf{init}_{\mathcal{A}\ltimes\mathcal{B}}(s))$. □

*Proof of Theorem 6.9.* We can now prove Theorem 6.9. The property $P_{\mathcal{A}\ltimes\mathcal{B}}$ is prefix-independent. We can therefore apply Corollary 6.7 to $\mathcal{A} \ltimes \mathcal{B}$. We can now combine with the previous technical lemmas, and we get the expected equivalence. □

## Appendix E. Details for single-clock timed automata

**Lemma 7.3.**
(1) *For every subregion $(q, J)$ of $q$ such that (i) $J$ is non-empty and open in $q$ (for the induced topology), and (ii) $\overline{J} \subseteq q$ is compact,*
(2) *for every thick edge $e$ enabled in $q$,*
(3) *for every subregion $(q', J')$ of $q'$ such that for every $s \in (q, J)$, $e(s) \cap J'$ is non-empty and open in $q'$ (for the induced topology), where $e(s) = \{s' \mid \exists \tau \in \mathbb{R}_+ \text{ s.t. } s \xrightarrow{\tau,e} s'\}$,*
(4) *for every state $s$ of $\mathsf{R}(\mathcal{A})$ such that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q, J)) > 0$,*[a]

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q, J) \xrightarrow{e} (q', J') \mid \mathbf{G}\,\mathbf{F}\,(q, J)) = 1\,.$$

[a]This is for the next conditional probability to be defined.

*Proof.* We write $\mathbb{P}_{\mathcal{A}}(s \xrightarrow{e} (q', J'))$ for the probability of the set of runs starting from $s$ with a move $s \xrightarrow{\tau,e} s'$ with $s' \in (q', J')$ and for some $\tau \in \mathbb{R}_+$.[16]

Let $\lambda \stackrel{\text{def}}{=} \inf_{s \in (q,J)} \mathbb{P}_{\mathcal{A}}(s \xrightarrow{e} (q', J'))$. Since $\overline{J} \subseteq q$ is compact and $\forall s \in q$, $\mathbb{P}_{\mathcal{A}}(s \xrightarrow{e} (q', J')) > 0$ (because $e$ is thick and $e(s) \cap J'$ is non-empty and open), $\lambda > 0$. Indeed we have supposed that for all $\ell \in L$, for all $[a, b] \subseteq \mathbb{R}_+$, the function $v \mapsto \mu_{(\ell,v)}([a, b])$ is continuous, see hypothesis $(H3)$ in (†) (page 28), hence $s \mapsto \mathbb{P}_{\mathcal{A}}(s \xrightarrow{e} (q', J'))$ is continuous.

Denote $E_k$ the set of paths in $\mathcal{A}$ that visit $(q, J)$ infinitely often, but from the $k$-th passage in $(q, J)$ on never fire $(q, J) \xrightarrow{e} (q', J')$ anymore. Note that the set $E_k$ is $\mathbb{P}_{\mathcal{A}}$-measurable, and that $\mathbb{P}_{\mathcal{A}}(E_k) \leq \prod_k^{\infty}(1 - \lambda) = 0$. Then note that the set $\bigcup_{k \geq 1} E_k$ can be equivalently defined by $B \wedge \neg A$ where $B$ is '$\mathbf{G}\,\mathbf{F}\,(q, J)$' and $A$ is '$\mathbf{G}\,\mathbf{F}\,(q, J) \xrightarrow{e} (q', J')$'. Hence, we get that $\mathbb{P}_{\mathcal{A}}(s, B \wedge \neg A) \leq \lim_{k \to +\infty} \mathbb{P}_{\mathcal{A}}(E_k) = 0$, and thus

$$\begin{aligned}
\mathbb{P}_{\mathcal{A}}(s, A \mid B) &= \tfrac{\mathbb{P}_{\mathcal{A}}(s, A \wedge B)}{\mathbb{P}_{\mathcal{A}}(s, B)} \quad \text{(by definition)} \\
&= \tfrac{\mathbb{P}_{\mathcal{A}}(s, A \wedge B)}{\mathbb{P}_{\mathcal{A}}(s, A \wedge B) + \mathbb{P}_{\mathcal{A}}(s, \neg A \wedge B)} \\
&\qquad \text{(by Bayes formulas)} \\
&= 1 \quad \text{(because } \mathbb{P}_{\mathcal{A}}(s, B \wedge \neg A) = 0\text{)}
\end{aligned}$$

which is exactly $\mathbb{P}_{\mathcal{A}}(s, \mathbf{G}\,\mathbf{F}\,(q, J) \xrightarrow{e} (q', J') \mid \mathbf{G}\,\mathbf{F}\,(q, J)) = 1$. □

---

**Theorem 7.2.** *Assuming $\mathcal{A}$ satisfies (†), if $s$ is a state of $\mathcal{A}$, $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$.*

*Proof.* Let $s$ be a state in $\mathcal{A}$. We want to prove that $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{fair}) = 1$. We will equivalently prove $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{fair}) = 1$. To that purpose, we decompose the set of infinite runs in $\mathsf{R}(\mathcal{A})$ from $\iota(s)$ into:

($F_1$) the set of runs with infinitely many resets,
($F_2$) the set of runs with finitely many resets, and which are ultimately in the unbounded region $(c_k, +\infty)$,

[16]Note that this set is $\mathbb{P}_{\mathcal{A}}$-measurable because it can be seen as $\mathsf{Cyl}(\pi_{\mathcal{C}_{J'}}(s, e))$ for some constraint $\mathcal{C}_{J'}$ enforcing the first move to lead to $J'$.

($F_3$) the set of runs with finitely many resets, and which ultimately stay forever in a bounded region, either $\{c_i\}$ with $0 \leq i \leq k$, or $(c_i, c_{i+1})$ with $0 \leq i < k$. We write $(F_3^{(c_i,c_{i+1})})$ (resp. $(F_3^{c_i})$) for condition $F_3$ restricted to $(c_i, c_{i+1})$ (resp. $\{c_i\}$).

We write $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j)$ for the probability of the runs starting in $s$ and satisfying condition $F_j$. The three sets of runs above are measurable and partition the set of all runs. Hence $\sum_{j=1,2,3} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j) = 1$, and applying Bayes formula:

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair}) = \sum_{j=1,2,3} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_j) \cdot \mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j)\,. \qquad (\bullet)$$

We now distinguish between the three cases to prove that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_j) = 1$ (in case $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, F_j) = 0$ we remove the corresponding term from $(\bullet)$).

**Case $F_1$:** We consider the set of runs with infinitely many resets. Let $\varrho = s_0 \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} \dots$ be such a run. There exists $q$ such that for infinitely many $i$ with $i \in \mathbb{N}$, $s_i = (q, 0)$ (since $\mathcal{A}$ is single-clock). Now, fix a state $(q, 0)$ and assume that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q, 0)) > 0$ (otherwise the set of runs visiting infinitely often $(q, 0)$ will be negligible). For every sequence $\sigma$ of edges and compact sets (as in the statement of Lemma 7.5), we get that

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,\sigma \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) = 1\,.$$

Hence, for sequences of edges $(e_i)_{1 \leq i \leq p}$ such that such a $\sigma$ exists, we get that

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q, 0) \xrightarrow{e_1} q_1 \dots \xrightarrow{e_p} q_p \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) = 1\,. \qquad (\star)$$

Now notice that such a $\sigma$ always exists whenever these edges are thick, hence $(\star)$ holds for every sequence of consecutive thick edges.

Now, fix a thick edge $e$, and assume that the set of paths passing through $(q, 0)$ infinitely often and enabling $e$ infinitely often, has a positive probability. We will then prove that

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, (\mathbf{G}\,\mathbf{F}\; e \text{ enabled}) \Rightarrow (\mathbf{G}\,\mathbf{F}\; \xrightarrow{e}) \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) = 1\,,$$

which will imply that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_1) = 1$.

- Assume that $e$ is reachable from $(q, 0)$ following thick edges, say $(e_i)_{1 \leq i \leq p}$ with $e_p = e$. Then, applying $(\star)$, we get that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\,(q, 0) \xrightarrow{e_1} q_1 \dots \xrightarrow{e_p} q_p \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) = 1$, hence that $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\; \xrightarrow{e}| \mathbf{G}\,\mathbf{F}\,(q, 0)) = 1$.
- Assume on the contrary that $e$ is not reachable from $(q, 0)$ following thick edges. If $e$ is not reachable from $(q, 0)$, then $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F}\; e \text{ enabled} \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) = 0$. Let $W$ be the set of finite sequences of edges $(e_i)_{1 \leq i \leq p}$ leading from $(q, 0)$ to a state where $e$ is enabled. Then:

$$\begin{aligned}
&\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathbf{G}\,\mathbf{F}\; e \text{ enabled} \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) \\
&\quad = \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathbf{G}\,\mathbf{F}\; \textstyle\bigcup_{w \in W} w \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) \\
&\quad \leq \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathbf{F}\; \textstyle\bigcup_{w \in W} w \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) \\
&\quad = 0 \qquad \text{because one of the edges in } w \text{ is thin.}
\end{aligned}$$

In both cases, we get the expected property.

**Case $F_2$:** We consider the set of runs with finitely many resets and which end up in the unbounded region $(c_k, +\infty)$. Let $\varrho = s \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} \dots$ be such a run, and assume that from $s_n$ on, all states are in the unbounded region. From that state on, all edges which are enabled are thick and have guard $x > c_k$. Let $e$ be such an edge, and $q$

be the region-state source of $e$: the probability $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\mathsf{Cyl}(\pi(s,e)))$ for every $s \in q$ is independent of the choice of $s$ (as there is no restriction on delays, it is equal to $w_e/(\sum_{e' \text{ enabled at } q} w_{e'})$). Hence, ultimately, after having reached the unbounded region (and never leave it anymore), it will behave like a finite Markov chain.

Assume now that a resetting edge $e$ is enabled infinitely often along $\varrho$. Then, by a similar argument to the one in the proof of Lemma 7.3 with the $E_k$, as the probability distribution of taking an edge is lower-bounded (because we are now in a finite Markov chain), then any edge will be almost surely taken infinitely often. Hence,

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \mathbf{G}\,\mathbf{F} \text{ resetting edge enabled} \mid F_2) = 0\,,$$

and thus

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s, \neg(\mathbf{G}\,\mathbf{F} \text{ resetting edge enabled}) \mid F_2) = 1\,.$$

Once more, due to the distribution over edges (which is a finite Markov chain), when there is no more resetting edges, we get

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_2) = 1\,.$$

**Case $F_3$:** We consider the set of runs with finitely many resets and which end up in a bounded region. We assume the region $r \stackrel{\text{def}}{=} (c_i, c_{i+1})$. Let $\varrho = s \xrightarrow{e_1} s_1 \xrightarrow{e_2} \dots$ be a witness run, and we assume that from $s_n$ on, we are in region $r$. If $s_{j_1}$ and $s_{j_2}$ with $n \leq j_1 < j_2$ correspond to the same location, then the clock value of $s_{j_1}$ is less than (or equal to) that of $s_{j_2}$. Hence, if a thick edge $e$ and whose guard is included in $[c_{i+1}, +\infty)$ is enabled in $s_{j_1}$ (and thus also in $s_{j_2}$), the probability of taking $e$ from $s_{j_2}$ is greater than (or equal to) the probability of taking $e$ from $s_{j_1}$ (due to $(H4)$ in (†) (page 28) on $\mu$'s and to the fact that the discrete probability over edges is constant by regions). Hence, there is a positive lower bound for the probability of taking $e$, and if $e$ is enabled infinitely often, it will be taken infinitely often. Such an enabled edge is thus only possible with probability 0 under the assumption made in this case. Hence, with probability 1, only edges with guard $x \in r$ are enabled. For these edges, as previously, the system behaves like a finite Markov chain. We thus get that

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_3^{(c_i, c_{i+1})}) = 1\,.$$

If we now assume the region $r = \{c_i\}$, the reasoning is very similar to the previous one. Given a location $\ell$ along the suffix of the path where $x = c_i$ always holds, the edges enabled in $(\ell, x = c)$ are equipped with a distribution defining a finite Markov chain. Hence any edge enabled infinitely often will be taken infinitely often almost surely, which implies that

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair} \mid F_3^{c_i}) = 1\,.$$

Gathering all cases, we get the desired property, *i.e.*, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(s \models \mathsf{fair}) = 1$. □

E.1. **Details for Zenoness in single-clock timed automata.**

**Lemma 7.6.** *Assuming $\mathcal{A}$ satisfies (†), if $s$ is a state of $\mathcal{A}$, then:*

$$\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = \sum_{B \text{ Zeno BSCC of } \mathcal{G}_t(\mathcal{A})} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathbf{F}\, B)$$

*where a BSCC of $\mathcal{G}_t(\mathcal{A})$ is said Zeno whenever it is bounded and the clock is never reset.*

*Proof.* From Lemma 3.4 (resp. Lemma 4.19), we know that $(a)$ (resp. $(b)$) is equivalent to $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathbb{P}} \neg\mathsf{Zeno}$ (resp. $\mathsf{R}(\mathcal{A}), \iota(s) \not\approx_{\mathcal{T}} \neg\mathsf{Zeno}$).

We first remove syntactically all resets from edges of $\mathsf{R}(\mathcal{A})$ labelled by $x = 0$ since they are useless. We borrow the notations used in the proof of Theorem 7.2, and following that proof, we decompose the set of infinite runs from $s$ into:

($F_1$) the set of runs with infinitely many resets,
($F_2$) the set of runs with finitely many resets, and which are ultimately in the unbounded region $(c_k, +\infty)$,
($F_3$) the set of runs with finitely many resets, and which ultimately stay forever in a bounded region, either $\{c_i\}$ with $0 \leq i \leq k$, or $(c_i, c_{i+1})$ with $0 \leq i < k$.

We then also have:

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{Zeno}) = \sum_{i=1,2,3} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{Zeno} \mid F_i) \cdot \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s), F_i) \tag{E.1}$$

when these conditional probabilities are well-defined (otherwise it is correct to remove the term from the sum).

The proof of Lemma 7.6 is then decomposed into two parts, first we prove that the two first terms of the above sum are always equal to 0, and then that we can decide whether the last term is equal to 0.

**Lemma E.1.** $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{Zeno} \mid F_1) = 0$ *and* $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{Zeno} \mid F_2) = 0$.

*Proof.* We distinguish two cases.

**Case $F_1$:** We consider the set of runs with infinitely many resets. This set can be decomposed according to the states $(q, 0)$ (where $q \in Q$ is a region) that are visited infinitely often. We show that $\mathbb{P}_{\mathcal{A}}(\iota(s) \models \mathsf{Zeno} \mid \mathbf{G}\,\mathbf{F}\,(q, 0)) = 0$. In order to prove this, we distinguish the four following subcases depending on the set $I((q, 0))$: either (i) $I((q, 0)) \cap [0, 1) = \emptyset$, or (ii) $(0, 1) \subseteq I((q, 0))$, or (iii) $\{0\} \subsetneq I((q, 0))$, or (iv) $\{0\} = I((q, 0))$.

Let us first treat the easy case (i). If $I((q, 0)) \cap [0, 1) = \emptyset$, since the timed automaton is non-blocking, this means that each time the automaton arrives in state $(q, 0)$ at least 1 time unit elapses before the next transition. Hence a run visiting infinitely often such state $(q, 0)$ is necessarily non-Zeno.

Let us now consider case (ii), *i.e.*, we assume that $(0, 1) \subseteq I((q, 0))$. Since the probability distribution over the delays is then equivalent to the Lebesgue measure (see hypothesis $(\star)$), the probability of waiting a time delay $\tau \leq \frac{1}{2}$ in $(q, 0)$ is positive and strictly smaller than 1 (we write $\lambda_{(q,0)}$ for this value: $0 < \lambda_{(q,0)} < 1$). Let $E_k$ be the set of runs starting from $\iota(s)$, visiting $(q, 0)$ infinitely often, and such that from the $k$-th passage on, the time elapsed from state $(q, 0)$ (before taking an action) is less than $\frac{1}{2}$. We have $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(E_k) \leq \prod_k^\infty \lambda_{(q,0)} = 0$, and as a consequence

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}\big(\iota(s) \models \mathsf{Zeno} \mid \mathbf{G}\,\mathbf{F}\,(q, 0) \wedge (ii)\big) \leq \sum_{k=0}^{\infty} \mathbb{P}(E_k) = 0\,.$$

In case (iii), we assume that $\{0\} \subsetneq I((q, 0))$. If $(0, 1) \subseteq I((q, 0))$, we are done by case (ii). We can thus suppose that if $0 \neq \tau \in I((q, 0))$, we have that $\tau \geq 1$. If $I((q, 0))$ reduces to a finite union of points, the probability $\lambda_0$ of waiting a delay greater than or equal to 1 is positive and strictly smaller than 1 (because the measure is then equivalent to the uniform measure over those points, see hypothesis $(\star)$). When going infinitely

often through $(q,0)$, we will thus wait infinitely often a time greater than or equal to 1. If $I((q,0))$ contains an open interval, the probability of waiting a delay greater or equal than 1 from $(q,0)$ is 1 (by hypothesis $(\star)$). From this we can easily derive that:

$$\mathbb{P}_{\mathsf{R}(\mathcal{A})}\big(\iota(s) \models \mathsf{Zeno} \mid \mathbf{G}\,\mathbf{F}\,(q,0) \wedge (iii)\big) = 0\,.$$

Let us conclude with case (iv) where $I((q,0)) = \{0\}$. Since no positive delay can elapse from $(q,0)$, the probability of taking any edge enabled in $(q,0)$ is positive (the distribution over edges indeed becomes uniform). Hence, any state $(q_e,0)$ reachable from $(q,0)$ taking edge $e$, is almost surely infinitely often visited (as soon as $(q,0)$ is). From $(q_e,0)$, again two situations are possible: either $I((q_e,0)) = \{0\}$ or not. In the first case, note that it is necessarily the case that such a chain $(q,0) \xrightarrow{0,e_1} (q_1,0) \xrightarrow{0,e_2} (q_2,0)\cdots$ is finite, otherwise the run would contain only finitely many resets[17]. Thus we surely reach infinitely often a state $(q',0)$ such that $I((q',0)) \neq \{0\}$ allowing us to rely on the previous cases to obtain the desired results.

Gathering the four cases, we conclude that $\mathbb{P}_{\mathcal{A}}(\iota(s) \models \mathsf{Zeno} \mid \mathbf{G}\,\mathbf{F}\,(q,0)) = 0$. Hence

$$\mathbb{P}_{\mathcal{A}}(\iota(s) \models \mathsf{Zeno} \mid F_1) = 0\,.$$

**Case $F_2$:** We consider the set of runs with finitely many resets and which end up in the unbounded region. From any state in the unbounded region, the set of potential delays is necessarily of the form $[0,+\infty)$[18]. From hypothesis $(H5)$ in $(\dagger)$ on the distributions over delays, the probability of waiting a time delay $\tau \leq \frac{1}{2}$ from $s$, denoted $\lambda_s$, can be bounded by a constant: $0 < \lambda_s \leq \lambda_0 < 1$. Let $E_k$ denote the set of executions which, at the $k$-th step, are in the unbounded region without leaving it afterwards, and such that all delays afterwards are less than $\frac{1}{2}$. The probability of being Zeno when in $E_k$ satisfies: $\mathbb{P}(E_k) \leq \prod_{i>k} \lambda_0 = 0$, from which we derive:

$$\mathbb{P}(\iota(s) \models \mathsf{Zeno} \mid F_2) \leq \sum_{k=0}^{\infty} \mathbb{P}(E_k) = 0\,.$$

This concludes the proof of the Lemma 7.6. □

The case of condition $F_3$ is not similar to the two previous cases. Indeed, it is worth noticing that every execution satisfying the condition $F_3$ is Zeno. Hence, if $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models F_3) \neq 0$ (otherwise the term $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{Zeno} \mid F_3) \cdot \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models F_3)$ does not appear in the sum E.1), then $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathsf{Zeno} \mid F_3) = 1$. It remains to compute or characterise the value $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models F_3)$.

A BSCC $B$ in $\mathcal{G}_{\mathsf{b}}(\mathcal{A})$ is called a *Zeno BSCC* if it is bounded and contains no resetting edges. Note that in a Zeno BSCC the value of the clock lies in a unique interval $(c, c+1)$ (with $0 \leq c < M$) or $\{c\}$ (with $0 \leq c \leq M$).

**Lemma E.2.** $\mathbb{P}_{\mathsf{R}}(\mathcal{A})(\iota(s) \models F_3) = \sum_{B \text{ Zeno BSCC of } \mathcal{G}_{\mathsf{t}}(\mathcal{A})} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathbf{F}\,B)$.

*Proof.* Runs in $\mathsf{R}(\mathcal{A})$ are almost surely fair (thanks to Theorem 7.2), hence $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models F_3) = \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models F_3 \wedge \mathsf{fair})$. Now, a fair run in $\mathsf{R}(\mathcal{A})$ actually ends up in a BSCC of $\mathcal{G}_{\mathsf{t}}(\mathcal{A})$. It is now sufficient to remark that fair runs in $F_3$ end up in a BSCC that is bounded and does not reset the clock. Indeed, if one of these condition does not hold, the run would not

[17] Recall that edges labelled with $x = 0$ are not labelled with a reset.

[18] Otherwise the clock would be compared to a constant greater than the maximal one

be in $F_3$ (either it would end up in an unbounded region, or have infinitely many resets). Conversely, any run ending up in a Zeno BSCC is in $F_3$. Hence, the mentioned equality holds. □

This concludes the proof of the lemma. □

---

**Theorem 7.7.** *Assuming $\mathcal{A}$ satisfies (†), if $s$ is a state of $\mathcal{A}$, then the three following properties are equivalent:*
(a) $\mathcal{A}, s \not\approx_{\mathbb{P}} \neg\mathsf{Zeno}$;
(b) $\mathcal{A}, s \not\approx_{\mathcal{T}} \neg\mathsf{Zeno}$;
(c) *no Zeno BSCC is reachable in $\mathcal{G}_t(\mathcal{A})$ from $\iota(s)$.*

*Proof.* The equivalence of $(a)$ and $(c)$ is a consequence of Lemma 7.6 and of Theorem 6.2: $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = \sum_{B \text{ Zeno BSCC of } \mathcal{G}_t(\mathcal{A})} \mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathbf{F}\, B)$. Therefore, $\mathcal{A}, s \not\approx_{\mathbb{P}} \neg\mathsf{Zeno}$ iff $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = 0$, which is then equivalent to "for every Zeno BSCC $B$ of $\mathcal{G}_t(\mathcal{A})$, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathbf{F}\, B) = 0$", which is itself equivalent to "for every Zeno BSCC $B$ of $\mathcal{G}_t(\mathcal{A})$, $\mathbb{P}_{\mathsf{R}(\mathcal{A})}(\iota(s) \models \mathbf{G}\, \neg B) = 1$"; it remains to realise that $\mathbf{G}\, \neg B$ is a simple safety property, and to apply Theorem 6.2.

We now show the equivalence with $(b)$. We remove syntactically all resets from edges of $\mathsf{R}(\mathcal{A})$ labelled by $x = 0$ since they are useless. We also borrow the notations used in the proof of Theorem 7.2. Assume first that $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = 0$. Then no BSCC of $\mathcal{G}_t(\mathcal{A})$ is Zeno. We once more play a Banach-Mazur game using the basic open sets. Player 1 plays some move $B_1$, and player 2 then plays a move $B_2$ leading to a BSCC $B$ of $\mathcal{G}_t(\mathcal{A})$. By hypothesis, $B$ is not a Zeno BSCC, hence either it is not bounded, or it contains resetting edges.

- We first consider the case where $B$ contains no resetting edges. In that case, it means that the clock value when in $B$ is always above the maximal constant. Hence, the game can keep going on, and each time Player 2 chooses a move, she first chooses a move which constrains the cylinder saying that the delay has to be larger than 1. This is always possible, due to the form of the constraints, which all include $(c_k, +\infty)$. In that case, it is not difficult to check that the resulting runs are all non-Zeno.
- We now consider the case where $B$ has resetting edges. Note that the clock can then become larger than 0. In that case, Player 2 can always choose a move so that it terminates with a resetting edge, but has visited a positive region, and has enforced that the value of the clock in that precise region was larger than 1/2. In that case also, all runs resulting from that play are non-Zeno.

Hence, we get that Player 2 has a strategy to avoid the set of Zeno runs, hence this set is meagre.

Conversely assume that the set of Zeno runs is meagre, but assume also that $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) > 0$. Once more, let's play the Banach-Mazur game. Player 2 has a strategy to avoid Zeno behaviours. However, as $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) > 0$, Player 1 can play a first move leading to a Zeno BSCC $B$ of $\mathcal{G}_t(\mathcal{A})$. Then $B$ has no resetting edges and lies within an interval $(c_i; c_{i+1})$ or $\{c_i\}$. Then whatever move Player 2 chooses, the resulting runs will all be Zeno, hence contradicting the assumption that the set of Zeno runs is meagre. The claim follows. □

## Appendix F. Details for Subsection 7.2 (reactive timed automata)

Let $s$ be a state of $\mathsf{R}(\mathcal{A})$, that we will take as initial. If $e$ is a thick edge in $T$, and $q \in Q$, we write $\mathfrak{R}^e(s)$ for the set of runs in $\mathsf{R}(\mathcal{A})$ that start in $s$ and take $e$ infinitely often, and $\mathfrak{R}^q(s)$ for the set of runs of $\mathsf{R}(\mathcal{A})$ that start in $s$ and visit $q$ infinitely often. In particular, we write $\mathfrak{R}^{\mathsf{source}(e)}(s)$ for the set of runs that start in $s$ and visit $\mathsf{source}(e)$ infinitely often (hence along which $e$ is enabled infinitely often).

We fix a thick edge $e$ in $T$, and we let $\mathcal{Q}$ be the set of pairs $q = (\ell, r)$ where $r$ is memoryless and $\mathcal{Q}'$ the set of elements $q = (\ell, r) \in \mathcal{Q}$ such that

$$\mathbb{P}(\mathfrak{R}^q(s)) > 0 \quad \text{and} \quad \mathbb{P}(\mathfrak{R}_0^{q,e}(s_q)) > 0$$

where $\mathfrak{R}_0^{q,e}(s_q)$ is the set of runs that start from $s_q$ and take $e$ before any other visit to $q$.

**Lemma 7.13.** *Assuming the above notations,*

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s)\right) = 1\,.$$

*Proof.* We let $\mathfrak{D}_n^{>M}(s)$ be the set of runs from $s$ that delay more than $M$ time units before taking the $n$-th transition (i.e. $\mathfrak{D}_n^{>M}(s) = \{\varrho \in \mathsf{Runs}(\mathcal{A}, s) \mid \varrho = s \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} \cdots \mid \tau_n > M\}$), and $\mathfrak{K}_n^\ell(s)$ (resp. $\mathfrak{K}_n^q(s)$ if $q = (\ell, r) \in \mathcal{Q}$) the set of runs from $s$ such that the $n$-th configuration is of the form $(\ell, v)$ (resp. $(\ell, v)$ with $v \in r$).

If we denote $\mathfrak{D}_{n,N}^{\leq M}(s) := \bigcap_{n\leq k\leq N}(\mathfrak{D}_k^{>M}(s))^c$ (where $\mathfrak{A}^c$ is the complement of any set $\mathfrak{A}$) and $S_N$ the set of locations $l \in L$ such that

$$\mathbb{P}\left(\mathfrak{D}_{n,N}^{\leq M}(s) \cap \mathfrak{K}_N^\ell(s)\right) > 0$$

then for any $\ell \in S_{N-1}$, we have

$$\begin{aligned}\mathbb{P}\left(\mathfrak{D}_{n,N}^{\leq M}(s) \mid \mathfrak{D}_{n,N-1}^{\leq M}(s) \cap \mathfrak{K}_{N-1}^\ell(s)\right) &= \mathbb{P}\left(\mathfrak{D}_{N,N}^{\leq M}(s) \mid \mathfrak{D}_{n,N-1}^{\leq M}(s) \cap \mathfrak{K}_{N-1}^\ell(s)\right)\\ &= \mathbb{P}\left(\mathfrak{D}_{N,N}^{\leq M}(s) \mid \mathfrak{K}_{N-1}^\ell(s)\right)\\ &= \mu_\ell([0, M])\end{aligned}$$

and thus

$$\begin{aligned}\mathbb{P}\left(\mathfrak{D}_{n,N}^{\leq M}(s)\right) &= \sum_{\ell\in S_{N-1}} \mathbb{P}\left(\mathfrak{D}_{n,N}^{\leq M}(s) \mid \mathfrak{D}_{n,N-1}^{\leq M}(s) \cap \mathfrak{K}_{N-1}^\ell(s)\right)\\ &\qquad \cdot \mathbb{P}\left(\mathfrak{D}_{n,N-1}^{\leq M}(s) \cap \mathfrak{K}_{N-1}^\ell(s)\right)\\ &= \sum_{\ell\in S_{N-1}} \mu_\ell([0, M])\mathbb{P}\left(\mathfrak{D}_{n,N-1}^{\leq M}(s) \cap \mathfrak{K}_{N-1}^\ell(s)\right)\\ &\leq \max_{\ell\in L} \mu_\ell([0, M]) \cdot \sum_{\ell\in S_{N-1}} \mathbb{P}\left(\mathfrak{D}_{n,N-1}^{\leq M}(s) \cap \mathfrak{K}_{N-1}^\ell(s)\right)\\ &= \max_{\ell\in L} \mu_\ell([0, M]) \cdot \mathbb{P}\left(\mathfrak{D}_{n,N-1}^{\leq M}(s)\right)\,.\end{aligned}$$

As for any $\ell \in L$, we have assumed $\mu_\ell([0, M]) < 1$, we conclude that

$$\mathbb{P}\left(\mathfrak{D}_{n,N}^{\leq M}(s)\right) \leq (\max_{\ell\in L} \mu_\ell([0, M]))^{N+1-n} \underset{N\to\infty}{\longrightarrow} 0$$

and thus that

$$\mathbb{P}\left(\bigcup_{n\in\mathbb{N}}\bigcap_{k\geq n}(\mathfrak{D}_k^{>M}(s))^c\right)=0\,.$$

We get that

$$\mathbb{P}\left(\bigcap_{n\in\mathbb{N}}\bigcup_{k\geq n}\mathfrak{D}_k^{>M}(s)\right)=1.$$

Now it is just a matter of noticing that if $\varrho = s \xrightarrow{t_1,a_1} s_1 \cdots \xrightarrow{t_n,a_n} s_n \cdots$ is in $\mathfrak{D}_n^{>M}(s)$, then $s_n$ is of the form $(\ell_n, v_n)$ with $(\ell_n, [v_n]_{\mathcal{A}}) \in \mathcal{Q}$ (a clock is either reset on the $n$-transition (hence its value is 0), or it is above $M$) and thus $\varrho \in \mathfrak{K}_n^{(\ell_n,[v_n]_{\mathcal{A}})}(s)$. Hence, we have $\mathfrak{D}_k^{>M}(s) \subset \bigcup_{q\in\mathcal{Q}} \mathfrak{K}_k^q(s)$ and

$$\mathbb{P}\left(\bigcap_{n\in\mathbb{N}}\bigcup_{k\geq n}\left(\bigcup_{q\in\mathcal{Q}}\mathfrak{K}_k^q(s)\right)\right)=1\,.$$

As $\mathcal{Q}$ is a finite set, we deduce

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}}\left(\bigcap_{n\in\mathbb{N}}\bigcup_{k\geq n}\mathfrak{K}_k^q(s)\right)\right)=1$$

and as $\bigcap_{n\in\mathbb{N}}\bigcup_{k\geq n}\mathfrak{K}_k^q(s)=\mathfrak{R}^q(s)$, we get

$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}}\mathfrak{R}^q(s)\right)=1\,.$$ □

---

Now we give the proof of Lemma 7.14. It will require quite long developments that we give in details.

**Lemma 7.14.** *Assuming the above notations, for all $q \in \mathcal{Q}'$*

$$\mathbb{P}\left(\mathfrak{R}^e(s) \mid \mathfrak{R}^q(s)\right)=1.$$

*Proof.* Let $q \in \mathcal{Q}'$. We want prove that

$$\mathbb{P}\left(\mathfrak{R}^e(s) \mid \mathfrak{R}^q(s)\right)=1$$

or equivalently that

$$\mathbb{P}\left(\mathfrak{R}^e(s)\cap\mathfrak{R}^q(s) \mid \mathfrak{R}^q(s)\right)=1.$$

We notice that the event $\mathfrak{R}^e(s)\cap\mathfrak{R}^q(s)$ coincides with

$$\bigcap_{n\in\mathbb{N}}\bigcup_{k\geq n}\mathfrak{R}_k^{q,e}(s)$$

where $\mathfrak{R}_k^{q,e}(s)$ is the set of runs starting in $s$ along which an occurrence of edge $e$ is preceded by precisely $k$ visits to $q$, i.e. $\mathfrak{R}_k^{q,e}(s) = \{\varrho \in \mathsf{Runs}(\mathcal{A}, s) \mid \varrho = s \xrightarrow{\tau_1,e_1} s_1 \ldots \xrightarrow{\tau_m,e_m} s_m \ldots$ and there exists $j$ s.t. $e_j = e$ and $\#\{1 \leq i < j \mid \mathsf{loc}(s_i) = q\} = k\}$, where $\mathsf{loc}(s_i)$ is the location of state $s_i$. We recall the following lemma, which is well-known in probability theory (see for example [Bil95]):

**Lemma F.1** (Borel-Cantelli)**.** *Assume $(\mathcal{E}, \mathbb{P})$ is a probabilistic space, and that the measurable events $(E_k)_{k\in\mathbb{N}}$ are independent. If $\sum_{k\in\mathbb{N}} \mathbb{P}(E_k) = +\infty$, then*

$$\mathbb{P}\left(\bigcap_{n\in\mathbb{N}} \bigcup_{k\geq n} E_k\right) = 1\,.$$

With the aim to apply this lemma, we will prove that the events $\mathfrak{R}_k^{q,e}(s)$ are independent in the $\mathfrak{R}^q(s)$-conditional $\sigma$-algebra, and that $\sum_{k\in\mathbb{N}} \mathbb{P}(\mathfrak{R}_k^{q,e}(s) \mid \mathfrak{R}^q(s)) = +\infty$, which will imply Lemma 7.14. This is non-trivial and will require several technical lemmas that we present now. The following arguments rely on result that will be given as Corollary F.7 (which is technical, and therefore postponed).

Independence of events.

**Lemma F.2.** *The events $\mathfrak{R}_k^{q,e}(s)$ are conditionally independent given $\mathfrak{R}^q(s)$.*

*Proof.* Defining $\mathfrak{R}_{\geq n}^q(s)$ as the set of runs starting in $s$ and visiting $q$ at least $n$ times, we compute:

$$\begin{aligned}
\mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \mid \mathfrak{R}^q(s)\right) &= \mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \mid \bigcap_{n>k} \mathfrak{R}_{\geq n}^q(s)\right) \\
&= \frac{\mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \cap \bigcap_{n>k} \mathfrak{R}_{\geq n}^q(s)\right)}{\mathbb{P}\left(\bigcap_{n>k} \mathfrak{R}_{\geq n}^q(s)\right)} \\
&= \frac{\mathbb{P}\left(\bigcap_{n>k} \left(\mathfrak{R}_k^{q,e}(s) \cap \mathfrak{R}_{\geq n}^q(s)\right)\right)}{\mathbb{P}\left(\bigcap_{n>k} \mathfrak{R}_{\geq n}^q(s)\right)}\,.
\end{aligned}$$

The two sequences $\left(\mathfrak{R}_k^{q,e}(s) \cap \mathfrak{R}_{\geq n}^q(s)\right)_{n>k}$ and $\left(\mathfrak{R}_{\geq n}^q(s)\right)_{n>k}$ are non-increasing, hence:

$$\begin{cases}
\mathbb{P}\left(\bigcap_{n>k} \left(\mathfrak{R}_k^{q,e}(s) \cap \mathfrak{R}_{\geq n}^q(s)\right)\right) = \lim_{n\to\infty} \mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \cap \mathfrak{R}_{\geq n}^q(s)\right) \\
\mathbb{P}\left(\bigcap_{n>k} \mathfrak{R}_{\geq n}^q(s)\right) = \lim_{n\to\infty} \mathbb{P}\left(\mathfrak{R}_{\geq n}^q(s)\right)\,.
\end{cases}$$

Thus, by Corollary F.7,

$$\begin{aligned}\mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \mid \mathfrak{R}^q(s)\right) &= \frac{\lim_{n\to\infty} \mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \cap \mathfrak{R}_{\geq n}^{q}(s)\right)}{\lim_{m\to\infty} \mathbb{P}\left(\mathfrak{R}_{\geq m}^{q}(s)\right)} \\ &= \frac{\lim_{n\to\infty} \mathbb{P}\left(\mathfrak{E}_q(s)\right) \cdot \mathbb{P}\left(\mathfrak{R}_0^{q,e}(s_q)\right)}{\lim_{m\to\infty} \mathbb{P}\left(\mathfrak{E}_q(s)\right)} \\ &= \mathbb{P}\left(\mathfrak{R}_0^{q,e}(s_q)\right)\end{aligned}$$

where $\mathfrak{E}_q(s)$ is the set of runs starting in $s$ and visiting $q$ at least once, and $s_q$ is the canonical configuration $(\ell, v_r)$ for $q = (\ell, r)$.

Similarly we prove, using Corollary F.7, that for $k \neq k'$:

$$\begin{aligned}\mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \cap \mathfrak{R}_{k'}^{q,e}(s) \mid \mathfrak{R}^q(s)\right) &= \mathbb{P}\left(\mathfrak{R}_0^{q,e}(s_q)\right)^2 \\ &= \mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \mid \mathfrak{R}^q(s)\right) \cdot \mathbb{P}\left(\mathfrak{R}_{k'}^{q,e}(s) \mid \mathfrak{R}^q(s)\right).\end{aligned}$$

We conclude that the two events $\mathfrak{R}_k^{q,e}(s)$ and $\mathfrak{R}_{k'}^{q,e}(s)$ are conditionally independent given $\mathfrak{R}^q(s)$. □

Divergence of the series.

**Lemma F.3.** $\sum_{k\in\mathbb{N}} \mathbb{P}(\mathfrak{R}_k^{q,e}(s) \mid \mathfrak{R}^q(s)) = +\infty$.

*Proof.* As in the previous lemma, we can deduce from equalities of Corollary F.7 that

$$\mathbb{P}\left(\mathfrak{R}_k^{q,e}(s) \mid \mathfrak{R}^q(s)\right) = \mathbb{P}\left(\mathfrak{R}_0^{q,e}(s_q)\right).$$

However, as $q \in \mathcal{Q}'$, we know by definition that we have

$$\mathbb{P}(\mathfrak{R}_0^{q,e}(s_q)) > 0.$$

The result follows. □

Decomposition using basic sets. This section aims to prove Corollary F.7 and so to complete the previous proofs.

**Lemma F.4.** *Let $r$ be a memoryless region, $v \in r$, and $s' = (\ell, v)$ a configuration of $\mathcal{A}$. Writing $q$ for region-state $(\ell, r)$, we have for every sequence $(e_1, \ldots, e_n) \in E^n$,*

$$\mathbb{P}(\pi(s', e_1 \ldots e_n)) = \mathbb{P}(\pi(s_q, e_1 \ldots e_n)).$$

*Proof.* We will prove a stronger result. We show that for every pair $s' = (\ell, v)$, $s'' = (\ell, v')$ satisfying for every $x \in X$, $v(x) = v'(x)$ or $\min(v(x), v'(x)) > M$, we have for every sequence $(e_1, \ldots, e_n) \in E^n$,

$$\mathbb{P}(\pi(s', e_1 \ldots e_n)) = \mathbb{P}(\pi(s'', e_1 \ldots e_n)).$$

We prove this result by induction on the length $n$ of the sequence of edges. The result trivially holds for $n = 0$. Assume that $n > 0$, and that the lemma holds for sequences of edges of length $n-1$. The constraint labelling the edge $e_1$ is equivalent to some $\bigwedge_{x\in X} (x \in I_x)$

where $I_x$ is an interval of the form $[c;c]$ for some integer $0 \leq c \leq M$, or $(c;c+1)$ for some integer $0 \leq c < M$, or $(M;+\infty)$. We have that:

$$\mathbb{P}(\pi(s', e_1 \dots e_n)) = \int_{t \in I(s', e_1)} p_{s'+t}(e_1)\, \mathbb{P}(\pi(s'^{e_1}_t, e_2 \dots e_n))\, \mathrm{d}\mu_\ell(t)$$

where $s' \xrightarrow{t,e_1} s'^{e_1}_t$. Now, it is not difficult to check that $p_{s'+t}(e_1) = p_{s''+t}(e_1)$ and $I(s', e_1) = I(s'', e_1)$ by hypothesis on $s'$ and $s''$. Also, writing $s'' \xrightarrow{t,e_1} s''^{e_1}_t$, we easily get that if $v'^{e_1}_t$ and $v''^{e_1}_t$ are the valuations of $s'^{e_1}_t$ and $s''^{e_1}_t$ then for every $x \in X$, we have $v'^{e_1}_t(x) = v''^{e_1}_t(x)$ or $\min(v'^{e_1}_t(x), v''^{e_1}_t(x)) > M$. We deduce by induction hypothesis:

$$\mathbb{P}(\pi(s'^{e_1}_t, e_2 \dots e_n)) = \mathbb{P}(\pi(s''^{e_1}_t, e_2 \dots e_n)).$$

Hence, we have

$$\begin{aligned}\mathbb{P}(\pi(s', e_1 \dots e_n)) &= \int_{t \in I(s'', e_1)} p_{s''+t}(e_1)\, \mathbb{P}(\pi(s''^{e_1}_t, e_2 \dots e_n))\, \mathrm{d}\mu_\ell(t) \\ &= \mathbb{P}(\pi(s'', e_1 \dots e_n)).\end{aligned}$$

This concludes the proof. □

We define $\mathfrak{E}_q(s)$ the set of runs starting in $s$ and visiting $q$ at least once.

**Proposition F.5.** *Let $q \in \mathcal{Q}'$. The following equalities hold true:*

(1) *For every $n \geq 1$,*

$$\mathbb{P}\left(\mathfrak{R}^q_{\geq n}(s)\right) = \mathbb{P}\left(\mathfrak{E}_q(s)\right) \cdot \mathbb{P}\left(\mathfrak{R}^q_{\geq 1}(s_q)\right)^{n-1}.$$

(2) *For every $1 \leq k < n$,*

$$\mathbb{P}\left(\mathfrak{R}^{q,e}_k(s) \cap \mathfrak{R}^q_{\geq n}(s)\right) = \mathbb{P}\left(\mathfrak{E}_q(s)\right) \cdot \mathbb{P}\left(\mathfrak{R}^q_{\geq 1}(s_q)\right)^{n-2} \cdot \mathbb{P}\left(\mathfrak{R}^q_{\geq 1}(s_q) \cap \mathfrak{R}^{q,e}_0(s_q)\right).$$

(3) *For every $1 \leq k < k' < n$,*

$$\begin{aligned}\mathbb{P}\left(\mathfrak{R}^{q,e}_k(s) \cap \mathfrak{R}^{q,e}_{k'}(s) \cap \mathfrak{R}^q_{\geq n}(s)\right) &= \mathbb{P}\left(\mathfrak{E}_q(s)\right) \\ &\quad \cdot \mathbb{P}\left(\mathfrak{R}^q_{\geq 1}(s_q)\right)^{n-3} \cdot \mathbb{P}\left(\mathfrak{R}^q_{\geq 1}(s_q) \cap \mathfrak{R}^{q,e}_0(s_q)\right)^2.\end{aligned}$$

*Proof.* We will only prove the first equality. The other equalities can be handled, using decompositions, similarly to the first equality.

We define the set $\varepsilon^q(s)$ as:

$$\begin{aligned}\bigcup_{h \in \mathbb{N}} \big\{(e_1, \dots, e_h) \in E^h \mid \mathsf{source}(e_1) = [s]_{\mathcal{A}}, \\ \mathsf{target}(e_h) = q, \text{ and } \mathsf{target}(e_i) \neq q \text{ for any } i < h\big\}.\end{aligned}$$

Note that a run $\varrho$ is in $\mathfrak{E}_q(s)$ iff there is some $(e_1, \dots, e_h) \in \varepsilon^q(s)$ with $\varrho \in \mathsf{Cyl}\left(\pi(s, e_1 \dots e_h)\right)$. The set $\varepsilon^q(s)$ is a *basic set* for $\mathfrak{E}_q(s')$. For every state $s'$, we define the set $\gamma^q_n(s')$ as:

$$\begin{aligned}\bigcup_{h \in \mathbb{N}} \big\{(e_1, \dots, e_h) \in E^h \mid \mathsf{source}(e_1) = [s']_{\mathcal{A}}, \\ \mathsf{target}(e_h) = q, \text{ and } \#\{1 \leq i \leq h \mid \mathsf{target}(e_i) = q\} = n\big\}.\end{aligned}$$

Note that a run $\varrho$ is in $\mathfrak{R}^q_{\geq n}(s')$ iff there is some $(e_1, \dots, e_h) \in \gamma^q_n(s')$ such that $\varrho \in \mathsf{Cyl}\left(\pi(s', e_1 \dots e_h)\right)$. The set $\gamma^q_n(s')$ is a *basic set* for $\mathfrak{R}^q_{\geq n}(s')$. Note also that if $s'$ and

$s''$ are region-equivalent, then the two sets $\gamma_n^q(s')$ and $\gamma_n^q(s'')$ coincide. We will thus write $\gamma_n^q(q)$, where $q$ is the region-state of $s'$ and $s''$.

We will decompose runs in $\mathfrak{R}^q_{\geq n}(s)$ using the basic sets $\varepsilon^q(s)$ and $\gamma_n^q(s_q)$. Indeed, if $\varrho \in \mathfrak{R}^q_{\geq n}(s)$, then we can decompose $\varrho$ into $\varrho' \cdot \varrho''$ such that there exist $(e_1, \dots, e_h) \in \varepsilon^q(s)$ with $\varrho' \in \pi(s, e_1 \dots e_h)$, and $(f_1, \dots, f_k) \in \gamma^q_{n-1}(\mathsf{last}(\varrho'))$ with $\varrho'' \in \mathsf{Cyl}\left(\pi(\mathsf{last}(\varrho'), f_1, \dots, f_k)\right)$. By definition of $\varepsilon^q(s)$, $\mathsf{last}(\varrho') \in q$, and by applying Lemma F.4, we get

$$\mathbb{P}\left(\pi(\mathsf{last}(\varrho'), f_1 \dots f_k)\right) = \mathbb{P}\left(\pi(s_q, f_1, \dots, f_k)\right).$$

If we denote $\sum_{\varepsilon^q(s)}$ instead of $\sum_{(e_1,\dots,e_h)\in\varepsilon^q(s)}$ and $\sum_{\gamma^q_{n-1}(q)}$ instead of $\sum_{(f_1,\dots,f_k)\in\gamma^q_{n-1}(q)}$, we have the following simplification:

$$\begin{aligned}
\mathbb{P}\left(\mathfrak{R}^q_{\geq n}(s)\right) &= \sum_{\varepsilon^q(s)} \sum_{\gamma^q_{n-1}(q)} \mathbb{P}\left\{\varrho = \varrho' \cdot \varrho'' \mid \varrho' \in \pi(s, e_1 \dots e_h),\right. \\
&\qquad\qquad \left. \varrho'' \in \mathsf{Cyl}\left(\pi(\mathsf{last}(\varrho'), f_1 \dots f_k)\right)\right\} \\
&= \sum_{\varepsilon^q(s)} \sum_{\gamma^q_{n-1}(q)} \left( \int_{t_1} \int_{t_2} \cdots \int_{t_h} \Big( \prod_{i=0}^{h-1} p_{s_i + t_{i+1}}(e_{i+1}) \Big) \right. \\
&\qquad\qquad \left. \cdot\, \mathbb{P}\left(\pi(s', f_1 \dots f_k)\right)\, \mathrm{d}\mu_{s_{h-1}}(t_h) \dots \mathrm{d}\mu_s(t_1) \right) \\
&\quad \left(\text{where } s' = \mathsf{last}(s \xrightarrow{t_1, e_1} s_1 \dots s_{h-1} \xrightarrow{t_h, e_h}) \in q\right) \\
&= \sum_{\varepsilon^q(s)} \sum_{\gamma^q_{n-1}(q)} \left( \int_{t_1} \int_{t_2} \cdots \int_{t_h} \Big( \prod_{i=0}^{h-1} p_{s_i + t_{i+1}}(e_{i+1}) \Big) \right. \\
&\qquad\qquad \left. \cdot\, \mathbb{P}\left(\pi(s_q, f_1 \dots f_k)\right)\, \mathrm{d}\mu_{s_{h-1}}(t_h) \dots \mathrm{d}\mu_s(t_1) \right) \\
&= \left( \sum_{\gamma^q_{n-1}(q)} \mathbb{P}\left(\pi(s_q, f_1 \dots f_k)\right) \right) \\
&\qquad \cdot \left( \sum_{\varepsilon^q(s)} \int_{t_1} \int_{t_2} \cdots \int_{t_h} \Big( \prod_{i=0}^{h-1} p_{s_i + t_{i+1}}(e_{i+1}) \Big) \mathrm{d}\mu_{s_{h-1}}(t_h) \dots \mathrm{d}\mu_{s_0}(t_1) \right) \\
&= \mathbb{P}\left(\mathfrak{R}^q_{\geq n-1}(s_q)\right) \cdot \mathbb{P}\left(\mathfrak{E}_q(s)\right).
\end{aligned}$$

By induction on $n$, using a similar decomposition, we can prove that:

$$\mathbb{P}\left(\mathfrak{R}^q_{\geq n}(s_q)\right) = \mathbb{P}\left(\mathfrak{R}^q_{\geq 1}(s_q)\right)^n$$

which concludes the proof for the first equality. □

We can simplify equalities of the previous proposition thanks to following lemma:

**Lemma F.6.** *Let $q \in \mathcal{Q}'$. We have $\mathbb{P}(\mathfrak{R}^q_{\geq 1}(s_q)) = 1$.*

*Proof.* For a contradiction, we assume that $\mathbb{P}(\mathfrak{R}^q_{\geq 1}(s_q)) = \alpha_0 < 1$. By Proposition F.5, we thus have that:

$$\begin{aligned}\mathbb{P}(\mathfrak{R}^q(s)) &= \mathbb{P}\left(\bigcap_n \mathfrak{R}^q_{\geq n}(s)\right) = \lim_{n\to\infty} \mathbb{P}\left(\mathfrak{R}^q_{\geq n}(s)\right) \\ &= \lim_{n\to\infty} \mathbb{P}\left(\mathfrak{E}_q(s)\right) \cdot \mathbb{P}\left(\mathfrak{R}^q_{\geq 1}(s_q)\right)^{n-1} \\ &= \mathbb{P}\left(\mathfrak{E}_q(s)\right) \lim_{n\to\infty} (\alpha_0)^{n-1} = 0\end{aligned}$$

which contradicts the fact that $q \in \mathcal{Q}'$. □

As an immediate corollary we get the following result.

**Corollary F.7.** *Let $q \in \mathcal{Q}'$. The following equalities hold true:*

(1) *For every $n \geq 1$,*
$$\mathbb{P}\left(\mathfrak{R}^q_{\geq n}(s)\right) = \mathbb{P}\left(\mathfrak{E}_q(s)\right).$$
(2) *For every $1 \leq k < n$,*
$$\mathbb{P}\left(\mathfrak{R}^{q,e}_k(s) \cap \mathfrak{R}^q_{\geq n}(s)\right) = \mathbb{P}\left(\mathfrak{E}_q(s)\right) \cdot \mathbb{P}\left(\mathfrak{R}^{q,e}_0(s_q)\right).$$
(3) *For every $1 \leq k < k' < n$,*
$$\mathbb{P}\left(\mathfrak{R}^{q,e}_k(s) \cap \mathfrak{R}^{q,e}_{k'}(s) \cap \mathfrak{R}^q_{\geq n}(s)\right) = \mathbb{P}\left(\mathfrak{E}_q(s)\right) \cdot \mathbb{P}\left(\mathfrak{R}^{q,e}_0(s_q)\right)^2.$$

This concludes the proof of Lemma 7.14. □

---

We now extend the previous study to weak reactive stochastic timed automata.

**Lemma 7.16.** *Let $\mathcal{A}$ be a weak reactive stochastic timed automaton and $s$ be a state of $\mathcal{A}$. Then*
$$\mathbb{P}\left(\bigcup_{q\in\mathcal{Q}} \mathfrak{R}^q(s)\right) = 1.$$

*Proof.* We seek to show that, with probability 1, we delay infinitely many times more than $M$ time units before taking a transition. Let $\mathfrak{D}^{>M}_n(s)$ be the set of runs from state $s$ that delays more than $M$ time units before taking the $n$-th transition (i.e. $\mathfrak{D}^{>M}_n(s) = \{\varrho \in \mathsf{Runs}(\mathcal{A}, s) \mid \varrho = s \xrightarrow{\tau_1,e_1} s_1 \xrightarrow{\tau_2,e_2} \cdots \mid \tau_n > M\}$). We denote $\mathfrak{D}^{\leq M}_{k,n}(s) := \bigcap_{k\leq j\leq n} (\mathfrak{D}^{>M}_j(s))^c$.

Assume $s = (\ell, v)$ with $\ell \in L_u$. Since $\mathcal{A}$ is a weak reactive stochastic timed automaton, there exists $0 < \lambda_0 \leq 1$ such that for every $\ell \in L_u$, for every $v \models \mathcal{I}(\ell)$, we have
$$\mu_{(\ell,v)}\left([M, +\infty[\,\right) \geq \lambda_0,$$
and there exist $0 < \lambda_1 \leq 1$ and $N \geq 1$ such that for any $\ell \in L_b$, any $v \models \mathcal{I}(\ell)$, we have
$$\mathbb{P}_{\mathcal{A}}\Big(\bigcup_{(e_1,\ldots,e_N)\in E_u} \pi((\ell, v), e_1, \ldots, e_N)\Big) \geq \lambda_1$$
where $E_u = \{(e_1, \ldots, e_N) \mid \mathsf{target}(e_i) \in L_u \text{ for some } 1 \leq i < N\}$.

Let $s = (\ell, v)$ be a state of $\mathcal{A}$. If $\ell \in L_u$, we have

$$\mathbb{P}\Big(\bigcup_{1\leq j\leq N} \mathfrak{D}_j^{>M}(s)\Big) \geq \mathbb{P}(\mathfrak{D}_1^{>M}(s)) \geq \lambda_0 \geq \lambda_0\lambda_1\,.$$

If $\ell \in L_b$, we first remark that for every $v \models \mathcal{I}(\ell)$, we have

$$\mathbb{P}_{\mathcal{A}}\Big(\bigcup_{(e_1,\dots,e_N)\in E_u} \pi(s, e_1, \dots, e_N)\Big) = \sum_{(e_1,\dots,e_k)\in F_u} \mathbb{P}_{\mathcal{A}}(\pi(s, e_1, \dots, e_k))$$

where

$$F_u = \{(e_1, \dots, e_k) \mid 1 \leq k < N, \mathsf{target}(e_k) \in L_u \text{ and for any } 1 \leq i < k,\ \mathsf{target}(e_i) \notin L_u\}.$$

Therefore, if $\ell \in L_b$, we have

$$\begin{aligned}
&\mathbb{P}(\bigcup_{1\leq j\leq N} \mathfrak{D}_j^{>M}(s))) \\
&\geq \sum_{(e_1,\dots,e_k)\in F_u} \sum_{e\in E} \mathbb{P}(\pi_{t_{k+1}>M}(s, e_1, \dots, e_k, e) \\
&\geq \sum_{(e_1,\dots,e_k)\in F_u} \sum_{e\in E} \int_{t_1\in I(s,e_1)} p_{s+t_1}(e_1)\cdots \\
&\qquad\qquad \cdots \int_{t_{k+1}\in I(s_k,e)\cap ]M,+\infty[} p_{s_k+t_{k+1}}(e)\mathrm{d}\mu_{s_k}(t_{k+1})\dots\mathrm{d}\mu_s(t_1) \\
&\geq \sum_{(e_1,\dots,e_k)\in F_u} \int_{t_1\in I(s,e_1)} p_{s+t_1}(e_1)\cdots \\
&\qquad\qquad \cdots \int_{t_{k+1}\in ]M,+\infty[} \sum_{e\in E} p_{s_k+t_{k+1}}(e)\mathrm{d}\mu_{s_k}(t_{k+1})\dots\mathrm{d}\mu_s(t_1) \\
&\geq \sum_{(e_1,\dots,e_k)\in F_u} \int_{t_1\in I(s,e_1)} p_{s+t_1}(e_1)\cdots \int_{t_{k+1}\in ]M,+\infty[} \mathrm{d}\mu_{s_k}(t_{k+1})\dots\mathrm{d}\mu_s(t_1) \\
&\geq \sum_{(e_1,\dots,e_k)\in F_u} \int_{t_1\in I(s,e_1)} p_{s+t_1}(e_1)\cdots \\
&\qquad\qquad \cdots \int_{t_k\in I(s_{k-1},e_k)} p_{s_{k-1}+t_k}(e_k)\mu_{s_k}(]M,+\infty[)\mathrm{d}\mu_{s_{k-1}}(t_k)\dots\mathrm{d}\mu_s(t_1) \\
&\geq \lambda_0 \sum_{(e_1,\dots,e_k)\in F_u} \mathbb{P}_{\mathcal{A}}(\pi(s, e_1, \dots, e_k)) \\
&\geq \lambda_0\lambda_1\,.
\end{aligned}$$

We deduce that for every state $s$, $\mathbb{P}(\mathfrak{D}_{1,N}^{\leq M}(s)) \leq 1-\lambda_0\lambda_1$. Therefore for any state $s$, for any $k \geq 2$, we get

$$\begin{aligned}
&\mathbb{P}(\mathfrak{D}_{1,kN}^{\leq M}(s)) \\
&\quad\leq \sum_{(e_1,\dots,e_{(k-1)N})} \int_{t_1 \in I(s,e_1)\cap[0,M]} p_{s+t_1}(e_1) \cdots \int_{t_{(k-1)N} \in I(s_{(k-1)N-1},e_{(k-1)N})\cap[0,M]} \\
&\qquad \Big(p_{s_{(k-1)N-1}+t_{(k-1)N}}(e_{(k-1)N})\mathbb{P}(\mathfrak{D}_{1,N}^{\leq M}(s_{(k-1)N}))\Big) \mathrm{d}\mu_{s_{(k-1)N-1}}(t_{(k-1)N}) \dots \mathrm{d}\mu_{(\ell,v)}(t_1) \\
&\quad\leq (1-\lambda_0\lambda_1)\mathbb{P}(\mathfrak{D}_{1,(k-1)N}^{\leq M}(s)) \\
&\quad\leq (1-\lambda_0\lambda_1)^k.
\end{aligned}$$

In the same way, we deduce that for any $j, k \geq 1$,

$$\mathbb{P}(\mathfrak{D}_{j,j+kN}^{\leq M}(s)) \leq (1-\lambda_0\lambda_1)^k .$$

We conclude that for any $k \geq 1$, $\mathbb{P}(\mathfrak{D}_{k,n}^{\leq M}(s))$ converges to 0 when $n$ tends to infinity and we finish the proof with the same arguments as in the proof of Lemma 7.13. □

**Proposition 7.19.** *Let $\mathcal{A}$ be a (weak) reactive stochastic timed automaton, and $s$ be a state of $\mathcal{A}$. Then $\mathbb{P}_{\mathcal{A}}(s \models \mathsf{Zeno}) = 0$.*

*Proof.* This is a consequence of the proofs of Lemma 7.13 (for reactive automata) and of Lemma 7.16 (for weak reactive automata).

We should just notice that, writing $\mathsf{Zeno}(s)$ for the set of Zeno runs from $s$,

$$\mathsf{Zeno}(s)^c \supseteq \bigcap_{n\in\mathbb{N}} \bigcup_{k\geq n} \mathfrak{D}_k^{>M}(s) .$$

And in the two proofs that are mentioned, it is proven that $\mathbb{P}(\mathfrak{D}_{k,n}^{\leq M}(s))$ tends to 0 when $n$ tends to $\infty$, which implies that $\mathbb{P}(\mathsf{Zeno}(s)) = 0$. □

## Appendix G. Details for Section 8

**Theorem 8.4.** (i) *The almost-sure model-checking of stochastic timed automata for simple safety properties is* PSPACE*-complete.*

(ii) *The almost-sure model-checking of single-clock stochastic timed automata for Büchi or Muller properties, or for properties given as specification (untimed) automata, is* NLOGSPACE*-complete.*[a]

(iii) *The almost-sure model-checking of single-clock stochastic timed automata for properties given as* LTL *formulas is* PSPACE*-complete.*

(iv) *The almost-sure model-checking of (weak) reactive stochastic timed automata for Büchi or Muller properties or properties given as specification timed automata is* PSPACE*-complete.*

[a] Note that simple safety or simple reachability properties can be expressed as small specification untimed automata, which yield an NLOGSPACE upper bound in those cases as well.

The upper bounds have already been explained in the core of the paper (they are obvious consequences of the previous developments). We will now explain several lower bounds.

**Hardness in reactive timed automata.** We prove that the almost-sure model-checking problem in reactive timed automata against simple safety and simple reachability properties is PSPACE-hard. To that aim we simulate a linearly-bounded Turing machine $\mathcal{M}$ on an input word $w_0$. The general reduction is rather standard [AL02] but it is required to work out the details so that there is no equality constraints in the constructed timed automaton, so that $\mathsf{R}(\mathcal{A})$ and $\mathcal{G}_{\mathfrak{t}}(\mathcal{A})$ coincide.

Let $N$ be the bound on the tape of $\mathcal{M}$ when simulating on input word $w_0$. We assume the alphabet is $\{a, b\}$ and we encode the content of $j$-th cell $C_j$ using a clock $x_j$ with the following convention: when we enter a module, cell $C_j$ contains an $a$ whenever $x_j < 1$ and it contains a $b$ whenever $x_j > 2$. To simulate a transition $q' = \delta(q, \alpha, \beta, \mathsf{dir})$ where $\alpha, \beta \in \{a, b\}$, we construct a module as in Fig. 13 for every index $i$ such that $1 \leq i \leq N$ and $1 \leq \mathsf{dir}(i) \leq N$, where $\mathsf{dir}(i)$ is either $i + 1$ (if the head goes to the left), or $i - 1$ (if the head goes to the right).

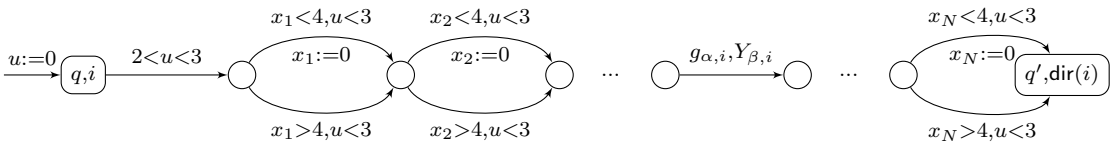



Figure 13: Simulation of $\mathcal{M}$-transition $q' = \delta(q, \alpha, \beta, \mathsf{dir})$ where $\alpha, \beta \in \{a, b\}$. Index $i$ is such that $1 \leq i \leq N$ and $1 \leq \mathsf{dir}(i) \leq N$. Guard $g_{a,i}$ is $x_i < 4, u < 3$ and guard $g_{b,i}$ is $x_i > 4, u < 3$. Set $Y_{a,i}$ is $\{x_i\}$ and set $Y_{b,i}$ is $\emptyset$.

We complete the construction with an initialisation module (with input $w_0$ on the tape), and we complete the automaton so that it is reactive (by adding transitions to sink location). We note $\mathcal{A}$ for this timed automaton and write halt for the halting location (which can be made a sink). Let $P_{\mathsf{safety}}$ be the safety property that halt is not visited, and $P_{\mathsf{reach}}$ be the property of reaching sink. We attach the exponential distribution with parameter, say 1, to every state, and assume weight 1 for every edge. Let $s_0$ be the initial state of $\mathcal{A}$ where all clocks are set to 0.

**Lemma G.1.** *The following equivalences hold:*

$$\begin{aligned} \mathcal{M} \text{ does not halt on input } w_0 \quad &\Leftrightarrow \quad \mathbb{P}_{\mathcal{A}}(s_0 \models P_{\mathsf{safety}}) = 1 \\ &\Leftrightarrow \quad \mathbb{P}_{\mathcal{A}}(s_0 \models P_{\mathsf{reach}}) = 1 \end{aligned}$$

*Proof.* Assume $\mathcal{M}$ halts on input $w_0$. Then there is a finite run $\varrho$ in $\mathcal{A}$ leading to halt. Due to the special form of $\mathcal{A}$, the probability of the cylinder generated by $\pi_\varrho$ is positive ($\mathcal{A}$ has only strict guards). This implies that $\mathbb{P}_{\mathcal{A}}(s_0 \models P_{\mathsf{reach}}) < 1$ (since state sink is not reached by that cylinder), and $\mathbb{P}_{\mathcal{A}}(s_0 \models P_{\mathsf{safety}}) < 1$.

If $\mathcal{M}$ does not halt on input $w_0$, then location halt is never visited, and therefore sink is visited with probability 1. This implies that $\mathbb{P}_{\mathcal{A}}(s_0 \models P_{\mathsf{reach}}) = 1$, and that $\mathbb{P}_{\mathcal{A}}(s_0 \models P_{\mathsf{safety}}) = 1$. □

This shows hardness results for (i) and (iv).

**Hardness in single-clock timed automata.** The NLOGSPACE-hardness result of (ii) already holds for finite Markov chains (since checking reachability properties in finite graphs is NLOGSPACE-hard).

Similarly, the result of (iii) concerning PSPACE-hardness already holds for finite Markov chains [Var85].