FI MU

Faculty of Informatics
Masaryk University Brno

Stochastic Real-Time Games with Qualitative Timed Automata Objectives

by

Tomáš Brázdil
Jan Krčál
Jan Křetínský
Antonín Kučera
Vojtěch Řehák

FI MU Report Series FIMU-RS-2010-05

August 2010

Publications in the FI MU Report Series are in general accessible via WWW:

http://www.fi.muni.cz/reports/

Further information can be obtained by contacting:

Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic

Stochastic Real-Time Games with Qualitative Timed Automata Objectives*

Tomáš Brázdil Jan Krčál Jan Křetínský[†] Antonín Kučera
Vojtěch Řehák
Faculty of Informatics, Masaryk University,
Botanická 68a, 60200 Brno,
Czech Republic
{brazdil, krcal, kucera, rehak}@fi.muni.cz
jan.kretinsky@in.tum.de

September 2, 2010

Abstract

We consider two-player stochastic games over real-time probabilistic processes where the winning objective is specified by a timed automaton. The goal of player □ is to play in such a way that the play (a timed word) is accepted by the timed automaton with probability one. Player ◇ aims at the opposite. We prove that whenever player □ has a winning strategy, then she also has a strategy that can be specified by a timed automaton. The strategy automaton reads the history of a play, and the decisions taken by the strategy depend only on the region of the resulting configuration. We also give an exponential-time algorithm which computes a winning timed automaton strategy if it exists.

1 Introduction

In this paper, we study *stochastic real-time games (SRTGs)* which are obtained as a natural game-theoretic extension of *generalized semi-Markov processes (GSMP)* [1, 2, 3] or *real-time*

*The authors are supported by the Alexander von Humboldt Foundation (T. Brázdil), the Institute for Theoretical Computer Science, project No. 1M0545 (J. Krčál), Brno Municipality (J. Křetínský), and the Czech Science Foundation, grants No. P202/10/1469 (A. Kučera) and No. 201/08/P459 (V. Řehák).

[†]On leave at TU München, Boltzmannstr. 3, Garching, Germany.

probabilistic processes (RTP) [4]. Intuitively, all of these formalisms model systems which react to certain *events*, such as message receipts, subsystem failures, timeouts, etc. A common characteristic of all events is that they are *delayed* (it takes some time before an initiated event actually occurs) and *concurrent* (there can be several previously initiated events that are currently awaited). For example, if two messages e and e' are sent, it takes some (random) time before they arrive, and one can specify, or approximate, the *densities* f_e, $f_{e'}$ of their arrival times. When e arrives (say, after 20 time units), the system reacts to this event by changing its state, and awaits e' in a new state. The arrival time of e' in the new state is measured from zero again, and its density $f_{e'|20}$ is obtained from $f_{e'}$ by incorporating the condition that e' is delayed for at least 20 time units. That is, $f_{e'|20}(x) = f_e(x + 20)/\int_{20}^{\infty} f_e(y)\,dy$. Note that if the delays of all events are exponentially distributed, then $f_e = f_{e|b}$ for every $b \in \mathbb{R}_{\geq 0}$, and thus we obtain continuous-time Markov chains (see, e.g., [5]) and continuous-time stochastic games [6, 7] as restricted forms of RTPs and SRTGs, respectively.

Intuitively, a SRTG is a finite graph (see Fig. 1) with three types of nodes—*states* (drawn as large circles), *controls*, where each control can be either *internal* or *adversarial* (drawn as boxes and diamonds, respectively), and *actions* (drawn as small filled circles). In each state s, there is a finite subset $E(s)$ of events *scheduled* in s (the events scheduled in s are those which are "awaited" in a given state; the other events are disabled. Each state s can react to every event of $E(s)$ by entering a designated control c, where player $\Box$ or player $\Diamond$ chooses some of the available actions. Each action is associated with a fixed probability distribution over states. In general, both players can use randomized strategies, which means that they do not necessarily select just a single action but a probability distribution over the available actions, which is multiplied with the distributions associated to actions. Then, the next state is chosen randomly according to the constructed probability distribution, and the play goes on. Whenever a new state s' is entered from a previous state s along a play, each event scheduled in s' is assigned a new *delay* which is chosen randomly according to the corresponding (conditional) density. The state s' then "reacts" to the event with the least delay (under the assumptions adopted in this paper, the probability of assigning the same delay to different events is zero).

Our contribution. In this work we consider SRTGs with *deterministic timed automata (DTA)* objectives. Intuitively, a timed automaton "observes" a play of a given SRTG and checks that certain timing constraints are satisfied. A simple example of a property that can be encoded by a DTA is "whenever a new request is generated, it is either serviced within the next 10 time units, or the system eventually enters a safe state". In this case, we want to setup the internal controls so that the above property holds for almost all plays, no matter what deci-

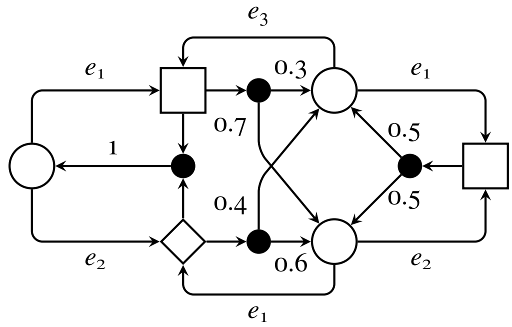

Figure 1: An example of a stochastic real-time game

sions are taken in adversarial controls. Hence, the aim of player $\Box$ is to maximize the probability that a play is accepted by a given timed automaton, while player $\Diamond$ aims at the opposite. By applying the result of [8], we obtain that SRTGs with DTA objectives have a *value*, i.e., $\sup_\sigma \inf_\pi P^{\sigma,\pi} = \inf_\pi \sup_\sigma P^{\sigma,\pi}$, where σ and π range over all strategies of player $\Box$ and player $\Diamond$, and $P^{\sigma,\pi}$ is the probability of all plays satisfying a given DTA objective. This immediately raises the question whether the players have *optimal* strategies which guarantee the equilibrium value against every strategy of the opponent. We show that the answer is *negative*. Then, we concentrate on the *qualitative* variant of the problem, which is perhaps most interesting from the practical point of view. An *almost-sure winning* strategy for player $\Box$ is a strategy such that for every strategy of player $\Diamond$, the probability of all plays satisfying a given DTA objective is equal to one. The main result of this paper is the following: We show that if player $\Box$ has *some* almost-sure winning strategy, then she also has a *DTA* almost-sure winning strategy, which can be encoded by a deterministic timed automaton $\mathcal{A}$ constructable in exponential time. The automaton $\mathcal{A}$ reads the history of a play, and the decision taken by the corresponding DTA strategy depends only on the region of the resulting configuration entered by $\mathcal{A}$.

Our constructions and proofs are combinations of standard techniques (used for timed automata and finite-state games) and some new non-trivial observations that are specific for the considered model of SRTGs. We also adapt some ideas presented in [4] (in particular, we use the concept of δ-separation).

Related work. Continuous-time (semi)Markov chains are a classical and deeply studied model with a mature mathematical theory (see, e.g., [5, 9]). Continuous-time Markov decision processes (CTMDPs) [10, 11, 12] combine probabilistic and non-deterministic choice, but all events are required to be exponentially distributed. Two player games over continuous-time Markov chains were considered only recently [6, 7]. Timed automata [13] were originally introduced as a non-stochastic model with time. Probabilistic semantics of timed automata was proposed in [14, 15], and a more general model of stochastic games over timed automata was considered in [16]. In this paper we build mainly on the previous work about GSMPs [1, 2, 3] and RTPs [4, 17] and interpret timed automata as a model-independent specification language which can express important properties of timed systems. This view is adopted also in [18] where continuous-time Markov chains are checked against timed-automata specifications.

Let us note that our technical treatment of events is somewhat different from the one used for GSMPs and RTPs. Intuitively, in GSMPs (and RTPs), each event is assigned its delay only when it is newly scheduled, and this delay is just updated when moving from state to state (by subtracting the elapsed time) until the event happens or it is disabled. For example, if two messages e and e' are sent, both of them are assigned randomly chosen delays d_e and $d_{e'}$. The smaller of the two delays (say d_e) triggers a transition to the next state, where the delay of $d_{e'}$ is updated by subtracting d_e. Since the current delays of all events are explicitly recorded in the state-space of GSMPs and RTPs, this formalism cannot be directly extended to perfect-information games (the players would "see" the delays assigned to events, i.e., they would know what is going to happen in the future). In our model of SRTGs, we always assign a new random delay to all events that are scheduled in a given control state, but we adjust the corresponding densities (from a "probabilistic" point of view, this approach is equivalent to the one used for GSMPs and RTPs).

2 Definitions

In this paper, the sets of all positive integers, non-negative integers, real numbers, positive real numbers, and non-negative real numbers are denoted by $\mathbb{N}$, $\mathbb{N}_0$, $\mathbb{R}$, $\mathbb{R}_{>0}$, and $\mathbb{R}_{\geq 0}$, respectively.

Let A be a finite or countably infinite set. A *probability distribution* on A is a function $f : A \to \mathbb{R}_{\geq 0}$ such that $\sum_{a \in A} f(a) = 1$. We say that f is *rational* if $f(a)$ is rational for every $a \in A$. The set of all distributions on A is denoted by $\mathcal{D}(A)$. A *σ-field* over a set Ω is a set $\mathcal{F} \subseteq 2^{\Omega}$ that includes Ω and is closed under complement and countable union. A *measurable space* is a pair $(\Omega, \mathcal{F})$ where Ω is a set called *sample space* and $\mathcal{F}$ is a σ-field over Ω whose elements are called *measurable sets*. A *probability measure* over a measurable space $(\Omega, \mathcal{F})$ is a function $\mathcal{P} : \mathcal{F} \to \mathbb{R}_{\geq 0}$ such that, for each countable collection $\{X_i\}_{i \in I}$ of pairwise disjoint elements of $\mathcal{F}$, $\mathcal{P}(\bigcup_{i \in I} X_i) = \sum_{i \in I} \mathcal{P}(X_i)$, and moreover $\mathcal{P}(\Omega) = 1$. A *probability space* is a triple $(\Omega, \mathcal{F}, \mathcal{P})$, where $(\Omega, \mathcal{F})$ is a measurable space and $\mathcal{P}$ is a probability measure over $(\Omega, \mathcal{F})$. We say that a property $A \subseteq \Omega$ holds *for almost all* elements of a measurable set Y if $\mathcal{P}(Y) > 0$, $A \cap Y \in \mathcal{F}$, and $\mathcal{P}(A \mid Y) = 1$.

Let us note that all of the integrals used in this paper should be understood as Lebesgue integrals, although we use Riemann-like notation.

2.1 Stochastic real-time games

Let $\mathcal{E}$ be a finite set of *events*, which are independent of each other. To every $e \in \mathcal{E}$ we associate its *lower bound* $\ell_e \in \mathbb{N}_0$, *upper bound* $u_e \in \mathbb{N} \cup \{\infty\}$, and a *density function* $f_e : \mathbb{R} \to \mathbb{R}$ which is positive on (ℓ_e, u_e) such that $\int_{\ell_e}^{u_e} f_e(x)\,dx = 1$. Further, for every $b \in \mathbb{R}_{\geq 0}$ we also define the *conditional density function* $f_{e|b} : \mathbb{R} \to \mathbb{R}$ as follows:

$$f_{e|b}(x) = \frac{f_e(x+b)}{\left[\int_b^{u_e} f_e(y)\,dy\right]_{\neq 0}}$$

Here $[\cdot]_{\neq 0} : \mathbb{R} \to \mathbb{R}$ is a function which for a given x returns x if $x \neq 0$, and 1 otherwise. The function f_e defines the density of delaying the event e, i.e., for every time $t \in \mathbb{R}_{\geq 0}$, the probability of delaying e for at most t is equal to $\int_0^t f_e(x)\,dx$. Note that the integral $\int_0^t f_{e|b}(x)\,dx$ is equal to the conditional probability of delaying e for at most $b+t$ under the condition that e is delayed for at least b. Since all events are mutually independent, for every subset $E' \subseteq E$ we have that the conditional probability of delaying all events in E' for at least $b+t$ under the condition that all events in E' are delayed for at least b is equal to $\prod_{e \in E'} \int_t^\infty f_{e|b}(x)\,dx$.

Definition 2.1. *A* stochastic real-time game (SRTG) *is a tuple* $\mathcal{G} = (S, E, C_\Box, C_\Diamond, Act, F, A, \mu_0)$ *where* S *is a finite set of* states, $E : S \to 2^{\mathcal{E}}$ *assigns to each* $s \in S$ *the set of events scheduled to occur in* s, $C_\Box$ *and* $C_\Diamond$ *are finite disjoint sets of* controls *of player* $\Box$ *and player* $\Diamond$, $Act \subseteq \mathcal{D}(S)$ *is a finite set of* actions, F *is a* flow function *which to every pair* (s, e), *where* $s \in S$ *and* $e \in E(s)$, *assigns a control of* $C_\Box \cup C_\Diamond$, $A : C_\Box \cup C_\Diamond \to 2^{Act}$ *assigns to each control* c *a non-empty finite set of actions enabled at* c, *and* $\mu_0 \in \mathcal{D}(S)$ *is an* initial *distribution.*

A *stamp* is an element (s, t, e) of $S \times \mathbb{R}_{>0} \times \mathcal{E}$ where $e \in E(s)$. A (computational) *history* of $\mathcal{G}$ is a finite sequence $\mathfrak{h} = (s_0, t_0, e_0) \cdots (s_n, t_n, e_n)$ of stamps. Intuitively, t_i is the time spent in s_i while waiting for some of the events scheduled in s_i, and e_i is the event that triggered a transition to the next state s_{i+1}. A *strategy* of player $\odot$, where $\odot \in \{\Box, \Diamond\}$, is a measurable function which to every history $(s_0, t_0, e_0) \cdots (s_n, t_n, e_n)$ such that $F(s_n, e_n) = c \in C_\odot$ assigns a probability distribution over the set $A(c)$ of actions that are enabled at c. The set of all strategies of player $\Box$ and player $\Diamond$ are denoted by Σ and Π, respectively.

Let $(\sigma, \pi) \in \Sigma \times \Pi$. The corresponding *play* of $\mathcal{G}$ is initiated in some $s_0 \in S$ (with probability $\mu_0(s_0)$). Then, each event $e \in E(s_0)$ is assigned a randomly chosen delay $d_e^0 \in \mathbb{R}_{>0}$

according to the density f_e (note that $f_e = f_{e|0}$). Let $t_0 = \min\{d^0_e \mid e \in E(s_0)\}$ be the minimal delay of all events scheduled in s_0, and let $\mathit{trigger}_0$ be the set of all $e \in E(s_0)$ such that $d^0_e = t_0$. The event e_0 which "triggers" a transition to the next state is the least element of $\mathit{trigger}_0$ w.r.t. some fixed linear ordering $\leq$ (note that the probability of assigning the same delay to different events is zero, and hence the choice of $\leq$ is irrelevant; we need this ordering just to make our semantics well defined). The event e_0 determines a control $c = F(s_0, e_0)$, where the responsible player makes a decision according to her strategy τ, i.e., selects a distribution $\tau(\mathfrak{h})$ over $A(c)$ where $\mathfrak{h} = (s_0, t_0, e_0)$ is the current history. Hence, the next state s_1 is chosen with probability $\sum_{\mu \in A(c)} \tau(\mathfrak{h})(\mu) \cdot \mu(s_1)$. In s_1, we assign a randomly chosen delay d^1_e to every $e \in E(s_1)$ according to the conditional density $f_{e|b}$, where b is determined as follows: If e was scheduled in the previous state s_0 and $e \neq e_0$, then $b = t_0$; otherwise $b = 0$. The event e_1 is the least event (w.r.t. $\leq$) with the minimal delay $t_1 = \min\{d^1_e \mid e \in E(s_1)\}$. The next state s_2 is chosen randomly by combining the strategy of the respective player with the corresponding actions. In general, after entering a state s_i, every $e \in E(s_i)$ is assigned a randomly chosen delay d^i_e according to the conditional density $f_{e|b}$ where b is the total waiting time for e accumulated in the history of the play.

To formalize the intuition given above, we define a suitable probability space $(\mathit{Play}, \mathcal{F}, \mathcal{P}^{\sigma,\pi}_{\mathfrak{h}})$ over the set Play of all infinite sequences of stamps, where $\mathfrak{h}$ is a history of steps "performed previously" (the technical convenience of $\mathfrak{h}$ becomes apparent later in Section 3; the definition given below is perhaps easier to understand in the special case when $\mathfrak{h}$ is empty). For the rest of this section, we fix a history $\mathfrak{h} = (s_0, t_0, e_0) \cdots (s_n, t_n, e_n)$ where $n \in \mathbb{N}_0 \cup \{-1\}$. If $n = -1$, then $\mathfrak{h}$ is empty. A *template* is a finite sequence of the form $B = (s_{n+1}, I_{n+1}, e_{n+1}) \cdots (s_{n+m}, I_{n+m}, e_{n+m})$ such that $m \geq 1$, $e_i \in E(s_i)$, and I_i is an interval in $\mathbb{R}_{>0}$ for every $n+1 \leq i \leq n+m$. Each such B determines the corresponding *cylinder* $\mathit{Play}(B) \subseteq \mathit{Play}$ consisting of all sequences of the form $(s_{n+1}, t_{n+1}, e_{n+1}) \cdots (s_{n+m}, t_{n+m}, e_{n+m}) \cdots$ where $t_i \in I_i$ for all $n+1 \leq i \leq n+m$. The σ-field $\mathcal{F}$ is the Borel σ-field generated by all cylinders. For each cylinder $\mathit{Play}(B)$, the probability $\mathcal{P}^{\sigma,\pi}_{\mathfrak{h}}(\mathit{Play}(B))$ is defined in the way described below. Then, $\mathcal{P}^{\sigma,\pi}_{\mathfrak{h}}$ is extended to $\mathcal{F}$ (in the unique way) by applying the extension theorem (see, e.g., [19]).

It remains to show how to define the probability $\mathcal{P}^{\sigma,\pi}_{\mathfrak{h}}(\mathit{Play}(B))$ of a given cylinder $\mathit{Play}(B)$, where $B = (s_{n+1}, I_{n+1}, e_{n+1}) \cdots (s_{n+m}, I_{n+m}, e_{n+m})$. We put $\mathcal{P}^{\sigma,\pi}_{\mathfrak{h}}(\mathit{Play}(B)) = T_{n+1}$, where the expression T_i is defined inductively for all $n+1 \leq i \leq n+m+1$ as follows:

$$T_i = \begin{cases} \int_{I_i} \mathit{State}_i \cdot \mathit{Win}_i \cdot T_{i+1} \, dt_i & \text{if } n+1 \leq i \leq n+m; \\ 1 & \text{if } i = n+m+1. \end{cases}$$

Observe that T_{n+1} is an expression with m nested integrals. Further, note that when constructing T_{i+1}, we already have $t_0, \ldots, t_i$ at our disposal (each t_i is either fixed in $\mathfrak{h}$, or it is a variable used in some of the preceding integrals).

The subterm $State_i$ corresponds to the probability that s_i is chosen as the next state, assuming that the current history is $(s_0, t_0, e_0) \cdots (s_{i-1}, t_{i-1}, e_{i-1})$. Hence, we define

- $State_{n+1} = \mu_0(s_{n+1})$ if $\mathfrak{h}$ is empty, otherwise $State_{n+1} = \sum_{\mu \in A(c)} \tau(\mathfrak{h})(\mu) \cdot \mu(s_{n+1})$, where $c = F(s_n, e_n)$, and τ is either σ or π, depending on whether $c \in C_\Box$ or $c \in C_\Diamond$, respectively.
- $State_i = \sum_{\mu \in A(c)} \tau(\mathfrak{h}')(\mu) \cdot \mu(s_i)$, where $n+1 < i \leq n+m$, $c = F(s_{i-1}, e_{i-1})$, $\mathfrak{h}' = (s_0, t_0, e_0) \cdots (s_{i-1}, t_{i-1}, e_{i-1})$, and τ is either σ or π, depending on whether $c \in C_\Box$ or $c \in C_\Diamond$, respectively.

The most complicated part is the definition of Win_i which intuitively corresponds to the probability that the event e_i "wins" the competition among the events scheduled in s_i.

In order to define Win_i, we have to overcome a technical obstacle that the events scheduled in s_i might have been scheduled also in the preceding states. For each $e \in E(s_i)$, let $K(e, i)$ be the minimal index such that $0 \leq K(e, i) \leq i$ and for all $K(e, i) \leq j < i$ we have that $e \in E(s_j)$ and $e \neq e_j$. We put $b(e, i) = t_{K(e,i)} + \cdots + t_{i-1}$. Intuitively, $b(e, i)$ is the total waiting time for e accumulated in the history of the play. Note that if $K(e, i) = i$, then the defining sum of $b(e, i)$ is empty and hence equal to zero. We put

$$Win_i \quad = \quad f_{e_i|b(e_i,i)}(t_i) \quad \cdot \prod_{\substack{e \in E(s_i) \\ e \neq e_i}} \int_{t_i}^{\infty} f_{e|b(e,i)}(x)\, dx.$$

2.2 Deterministic timed automata

Let $\mathcal{X}$ be a finite set of *clocks*. A *valuation* is a function $\nu : \mathcal{X} \to \mathbb{R}_{\geq 0}$. For every valuation ν and every subset $X \subseteq \mathcal{X}$ of clocks, we use $\nu[X := \vec{0}]$ to denote the unique valuation such that $\nu[X := \vec{0}](x) = 0$ for all $x \in X$, and $\nu[X := \vec{0}](x) = \nu(x)$ for all $x \in \mathcal{X} \setminus X$. Further, for every valuation ν and every $\delta \in \mathbb{R}_{\geq 0}$, the symbol $\nu + \delta$ denotes the unique valuation such that $(\nu + \delta)(x) = \nu(x) + \delta$ for all $x \in \mathcal{X}$.

A *clock constraint* (or *guard*) is a finite conjunction of basic constraints of the form $x \bowtie c$, where $x \in \mathcal{X}$, $\bowtie \in \{<, \leq, >, \geq\}$, and $c \in \mathbb{N}_0$. For every valuation ν and every clock constraint g we have that ν either does or does not satisfy g, written $\nu \models g$ or $\nu \not\models g$, respectively (the satisfaction relation is defined in the expected way). Sometimes we slightly abuse our notation and identify a guard g with the set of all valuations that satisfy g (for example, we write $g \cap g'$). The set of all guards over $\mathcal{X}$ is denoted by $\mathcal{B}(\mathcal{X})$.

Definition 2.2. *A* deterministic timed automaton (DTA) *is a tuple* $\mathcal{A} = (Q, \Sigma, \mathcal{X}, \longrightarrow, q_0, T)$, *where* Q *is a nonempty finite set of* locations, Σ *is a finite* alphabet, $\mathcal{X}$ *is a finite set of* clocks, $q_0 \in Q$ *is an* initial location, $T \subseteq Q$ *is a set of* target locations, *and* $\longrightarrow \subseteq Q \times \Sigma \times \mathcal{B}(\mathcal{X}) \times 2^{\mathcal{X}} \times Q$ *is an* edge relation *such that for all* $q \in Q$ *and* $a \in \Sigma$ *we have the following:*

1. *the guards are deterministic, i.e., for all edges of the form* (q, a, g_1, X_1, q_1) *and* (q, a, g_2, X_2, q_2) *such that* $g_1 \cap g_2 \neq \emptyset$ *we have that* $g_1 = g_2$, $X_1 = X_2$, *and* $q_1 = q_2$;

2. *the guards are total, i.e., for all* $q \in Q$, $a \in \Sigma$, *and every valuation* ν *there is an edge* (q, a, g, X, q') *such that* $\nu \models g$.

A *configuration* of $\mathcal{A}$ is a pair (q, ν), where $q \in Q$ and ν is a valuation. An *infinite timed word* is an infinite sequence $w = c_0 c_1 c_2 \cdots$ where each c_i is either a letter of Σ or a positive real number denoting a time stamp (note that letters and time stamps are not required to alternate in w). The *run* of $\mathcal{A}$ on w is the unique infinite sequence $(q_0, \nu_0)\, c_0\, (q_1, \nu_1)\, c_1 \cdots$ such that q_0 is the initial location of $\mathcal{A}$, $\nu_0 = \vec{0}$, and for each $i \in \mathbb{N}_0$ we have that

- if c_i is a time stamp $t \in \mathbb{R}_{\geq 0}$, then $q_{i+1} = q_i$ and $\nu_{i+1} = \nu_i + t$,
- if c_i is an input letter $a \in \Sigma$, then there is a unique edge (q_i, a, g, X, q) such that $\nu_i \models g$, and we require that $q_{i+1} = q$ and $\nu_{i+1} = \nu_i[X := \vec{0}]$.

We say that w is *accepted* by $\mathcal{A}$ if the run of $\mathcal{A}$ on w visits a configuration (q, ν) where $q \in T$. Without restrictions, we may assume that each $q \in T$ is *absorbing*, i.e., all of the outgoing edges of q lead back to q.

In this paper, we use DTA for two different purposes. Firstly, DTA are used as a generic *specification language* for properties of timed systems. In this case, a given DTA is constructed so that it accepts the set of all "correct" runs (timed words) of a given timed system. Formally, for a fixed SRTG $\mathcal{G}$ with a set of states S, a finite set Ap of atomic propositions and a labeling $L : S \to 2^{Ap}$, every play $\rho = (s_0, t_0, e_0)(s_1, t_1, e_1) \cdots$ of $\mathcal{G}$ determines a unique infinite timed word $Ap(\rho) = L(s_0)\, t_0\, L(s_1)\, t_1 \cdots$. A DTA $\mathcal{A}$ with alphabet 2^{Ap} then either accepts $Ap(\rho)$ or not. Intuitively, the automaton $\mathcal{A}$ encodes some desirable property of plays, and the aim of player $\Box$ and player $\Diamond$ is to maximize and minimize the probability of all plays accepted by $\mathcal{A}$, respectively. We denote $Play(\mathcal{A}) \subseteq Play$ the set of all plays ρ such that $Ap(\rho)$ is accepted by $\mathcal{A}$. Note that the DTA does not read any information about the events that occurred. However, one can easily encode the information about the last event into the subsequent state by considering copies s_e of each state s for every event e.

Secondly, we use DTA to encode strategies in stochastic real-time games. Here, the constructed DTA "observes" the history of a play, and the decisions taken by the corresponding strategy depend only on the resulting configuration (q, ν). Actually, we require that the decision depends only on the *region* of (q, ν) (see [13] or Section 3.1), which makes DTA strategies finitely representable. Formally, every history $\mathfrak{h} = (s_0, t_0, e_0) \cdots (s_n, t_n, e_n)$ of $\mathcal{G}$ can be seen as a (finite) timed word $s_0\, t_0\, e_0\, \cdots\, s_n\, t_n\, e_n$, where the states and events are seen as letters, and the delays are seen as time stamps. We define DTA strategies as follows.

Definition 2.3. *A* DTA strategy *is a strategy τ such that there is a DTA $\mathcal{A}$ with alphabet $S \cup \mathcal{E}$ satisfying the following: for every history $\mathfrak{h}$ we have that $\tau(\mathfrak{h})$ is a rational distribution which depends only on the region of (q, ν), where (q, ν) is the configuration entered by $\mathcal{A}$ after reading the word $\mathfrak{h}$.*

3 Results

For the rest of the paper, we fix an SRTG $\mathcal{G} = (S, E, C_\Box, C_\Diamond, Act, F, A, \mu_0)$, a finite set Ap of atomic propositions, a labeling $L : S \to 2^{Ap}$, and a DTA $\mathcal{A} = (Q, 2^{Ap}, \mathcal{X}, \longrightarrow, q_0, T)$.

As observed in [8], the determinacy result for Blackwell games [20] implies determinacy of a large class of stochastic games. This abstract class includes the games studied in this paper, and thus we obtain the following:

Proposition 3.1. *Let $\mathfrak{h}$ be a history of $\mathcal{G}$. Then*

$$\sup_{\sigma \in \Sigma} \inf_{\pi \in \Pi} \mathcal{P}_{\mathfrak{h}}^{\sigma,\pi}(Play(\mathcal{A})) \quad = \quad \inf_{\pi \in \Pi} \sup_{\sigma \in \Sigma} \mathcal{P}_{\mathfrak{h}}^{\sigma,\pi}(Play(\mathcal{A}))$$

The value *of $\mathcal{G}$ (with respect to $\mathfrak{h}$), denoted by $val_{\mathfrak{h}}$, is defined by the above equality.*

The existence of $val_{\mathfrak{h}}$ implies the existence of ε-optimal strategies for both players. However, note that player $\Box$ does not necessarily have an *optimal* strategy which would achieve the outcome $val_{\mathfrak{h}}$ or better against every strategy of player $\Diamond$, even if $val_{\mathfrak{h}} = 1$ and $C_\Diamond = \emptyset$. A simple counterexample is given in Fig. 2. Here f_e is the uniform density on $(0, 1)$ (i.e., $f_e(x) = 1$ for all $x \in (0, 1)$), $Ap = \{p_0, p_1\}$, $L(s_0) = \{p_0\}$, $L(s_1) = \{p_1\}$, and the only target location is gray. All of the "missing" edges in the depicted DTA (which are needed to satisfy the requirement that the guards are total) lead to a "garbage" location. The initial distribution μ_0 assigns 1 to s_0. Now observe that $val_{\mathfrak{h}} = 1$ (where $\mathfrak{h}$ is the empty history), because for every $\varepsilon > 0$, player $\Box$ can "wait" in s_0 until e is fired so that its delay is smaller than ε (this eventually happens with probability 1), and then she moves to s_1. The probability that e is assigned a delay at most $1 - \varepsilon$

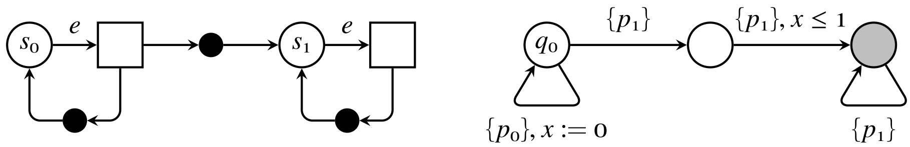

Figure 2: Player □ does not have an optimal strategy.

in s_1 is $1 - \varepsilon$, and hence the constructed DFA accepts a play with probability $1 - \varepsilon$. However, player □ has no optimal strategy.

In this paper we consider the existence and effective constructability of *almost-sure* winning strategies for player □. Formally, a strategy $\sigma \in \Sigma$ is almost-sure winning for a history $\mathfrak{h}$ if for every strategy $\pi \in \Pi$ we have that $\mathcal{P}^{\sigma,\pi}_{\mathfrak{h}}(Play(\mathcal{A})) = 1$. We show the following:

Theorem 3.2. *Let $\mathfrak{h}$ be a history. If player □ has (some) almost-sure winning strategy for $\mathfrak{h}$, then she also has a DTA almost-sure winning strategy for $\mathfrak{h}$. The problem whether player □ has a (DTA) almost-sure winning strategy for $\mathfrak{h}$ is solvable in exponential time. A DTA almost-sure winning strategy is computable in exponential time if it exists.*

A proof of Theorem 3.2 is not immediate and requires several steps. First, in Section 3.1 we construct a *product game* $\mathcal{G}_{\mathcal{A}}$ of $\mathcal{G}$ and $\mathcal{A}$ and show that $\mathcal{G}_{\mathcal{A}}$ can be examined instead of $\mathcal{G}$ and $\mathcal{A}$. The existence of a DTA almost-sure winning strategy in $\mathcal{G}_{\mathcal{A}}$ is analyzed in Section 3.2. Finally, in Section 3.3 we present an algorithm which computes a DTA almost-sure winning strategy if it exists.

3.1 The product game

Intuitively, the product game of $\mathcal{G}$ and $\mathcal{A}$, denoted by $\mathcal{G}_{\mathcal{A}}$, is constructed by simulating the execution of $\mathcal{A}$ on-the-fly in $\mathcal{G}$. Waiting times for events and clock valuations are represented explicitly in the states of $\mathcal{G}_{\mathcal{A}}$, and hence the state-space of $\mathcal{G}_{\mathcal{A}}$ is uncountable. Still, $\mathcal{G}_{\mathcal{A}}$ is in many aspects similar to $\mathcal{G}$, and therefore we use a suggestive notation compatible with the one used for $\mathcal{G}$. To distinguish among the notions related to game $\mathcal{G}$ and product $\mathcal{G}_{\mathcal{A}}$, we consistently use the "p-" prefix. Hence, $\mathcal{G}$ has stamps, states, histories, etc., while $\mathcal{G}_{\mathcal{A}}$ has p-stamps, p-states, p-histories, etc.

Let $n = |\mathcal{E}| + |X|$. The clock values of $\mathcal{A}$ and the delays of currently scheduled events are represented by a *p-vector* $\xi \in \mathbb{R}^n_{\geq 0}$. The set of *p-states* is $S \times Q \times \mathbb{R}^n_{\geq 0}$, and the sets of *p-controls* of player □ and player ◇ are $C_\Box \times Q \times \mathbb{R}^n_{\geq 0}$ and $C_\Diamond \times Q \times \mathbb{R}^n_{\geq 0}$, respectively.

A *p-stamp* is an element (s, q, ξ, t, e) of $S \times Q \times \mathbb{R}^n_{\geq 0} \times \mathbb{R}_{>0} \times \mathcal{E}$. The dynamics of $\mathcal{G}_{\mathcal{A}}$ is determined as follows. First, we define a *p-flow* function $F_{\mathcal{A}}$, which to a given p-stamp (s, q, ξ, t, e)

assigns the p-control (c, q', ξ'), where $c = F(s, e)$, and q', ξ' are determined as follows. Let $(q, L(s), g, X, q')$ be the unique edge of $\mathcal{A}$ such that the guard g is satisfied by the clock valuation stored in $\xi + t$. We put $\xi' = (\xi +_s t)[(e \cup X) := \vec{0}]$. The operator "$+_s t$" adds t to all clocks stored in ξ and to all events scheduled in s, and $(e \cup X) := \vec{0}$ resets all clocks of X to zero and assigns zero delay to e. Second, we define the set of p-actions. For every p-control (c, q, ξ) and an action $a \in A(c)$, there is a corresponding p-action which to a given p-state (s', q, ξ'), where $\xi' = \xi[(\mathcal{E} \setminus E(s')) := \vec{0}]$, assigns the probability $a(s')$.

Now we define *p-histories* and *p-plays* as sequences of p-stamps. In the game $\mathcal{G}$ we allowed arbitrary sequences of stamps, whereas in the product game we need the automaton part of the product to be consistent with the game part. We say that a p-stamp $x_0 = (s_0, q_0, \xi_0, t_0, e_0)$ is consistent with a p-stamp $x_1 = (s_1, q_1, \xi_1, t_1, e_1)$ if the image of x_0 under the p-flow function is a p-control (c, q_1, ξ') such that $\xi_1 = \xi'[A := \vec{0}]$ where A is the set of actions not enabled in s_1.

A *p-history* is a finite sequence of p-stamps $\mathfrak{p} = x_0 \cdots x_n$ such that x_i is consistent with x_{i+1} for all $0 \le i < n$. A *p-play* is an infinite sequence of p-stamps $x_0 x_1 \cdots$ where each finite prefix $x_0 \cdots x_i$ is a p-history. Each p-history $\mathfrak{p} = (s_0, q_0, \xi_0, t_0, e_0) \cdots (s_n, q_n, \xi_n, t_n, e_n)$ can be mapped to a unique history $H(\mathfrak{p}) = (s_0, t_0, e_0) \cdots (s_n, t_n, e_n)$. Note that H is in fact a bijection, because each history induces a unique finite execution of the DTA $\mathcal{A}$ and the consistency condition reflects this unique execution. By the *last p-control* of a p-history $\mathfrak{p}$ we denote the image of the last p-stamp of $\mathfrak{p}$ under the p-flow function.

3.1.1 Probability space in the product game

Note that every pair of strategies $(\sigma, \pi) \in \Sigma \times \Pi$ defined for the original game $\mathcal{G}$ can also be applied in the constructed product game $\mathcal{G}_{\mathcal{A}}$ (we just ignore the extra components of p-stamps). By re-using the construction of Section 2.1, we define a probability space $(Play, \mathcal{F}, \mathcal{P}^{\sigma,\pi}_{\mathfrak{p}})$ for every pair of strategies $(\sigma, \pi) \in \Sigma \times \Pi$ and every p-history $\mathfrak{p}$. In the following, we first define a probability measure on a semialgebra of p-cylinders. Then, the probability measure $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}}$ is extended to $\mathcal{F}$ by applying the extension theorem.

For the following, we fix a p-history $\mathfrak{p} = (s_0, q_0, \xi_0, t_0, e_0) \cdots (s_n, q_n, \xi_n, t_n, e_n)$ where $n \in \mathbb{N}_0 \cup \{-1\}$. If $n = -1$, then $\mathfrak{p}$ is empty. A *p-template* is a finite sequence of the form $B = (s_{n+1}, q_{n+1}, I_{n+1}, e_{n+1}) \cdots (s_{n+m}, q_{n+m}, I_{n+m}, e_{n+m})$ such that $m \ge 1$, s_i is a state of S, q_i is a location of Q, I_i is an interval in $\mathbb{R}_{>0}$, and e_i is an event of $E(s_i)$ for every $n{+}1 \le i \le n{+}m$. Each p-template B determines the corresponding *p-cylinder* $Play(B) \subseteq Play$ consisting of all sequences of the form $(s_{n+1}, q_{n+1}, \xi_{n+1}, t_{n+1}, e_{n+1}) \cdots (s_{n+m}, q_{n+m}, \xi_{n+m}, t_{n+m}, e_{n+m}) \cdots$ where, for all $n + 1 \le i \le n{+}m$, it holds that $t_i \in I_i$ and $\xi_i = \xi[(\mathcal{E} \setminus E(s_{i+1})) := \vec{0}]$ such that $F_{\mathcal{A}}((s_{i-1}, q_{i-1}, \xi_{i-1}, t_{i-1}, e_{i-1})) =$

(c, q, ξ). For each p-cylinder $Play(B)$, the probability measure $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}}(Play(B))$ closely follows the definition of the probability measure from Section 2. We put $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}}(Play(B)) = T_{n+1}$, where the expression T_i is defined inductively for all $n+1 \leq i \leq n+m+1$ as follows:

$$T_i = \begin{cases} \int_{I_i} State_i \cdot Location_i \cdot Win_i \cdot T_{i+1}\, dt_i & \text{if } n+1 \leq i \leq n+m; \\ 1 & \text{if } i = n+m+1 \end{cases}$$

The expression T_{n+1} contains m nested integrals. Note, that when constructing T_{i+1}, we already have variables $t_0, \ldots, t_i$ available (each t_i is either fixed in $\mathfrak{p}$, or it is introduced in some of the preceding integrals).

The sub-terms $State_i$ and Win_i are the same as in Section 2. The sub-term $Location_i$ reflects whether the automaton moves to q_i as the next location, assuming that the current p-history is $\mathfrak{p}_i = (s_0, q_0, \xi_0, t_0, e_0) \cdots (s_{i-1}, q_{i-1}, \xi_{i-1}, t_{i-1}, e_{i-1})$. In the definition we use an indicator function $\delta(q, q')$ defined as 1 if $q = q'$ and 0 otherwise. We define

> $Location_i = \delta(q_i, q)$, where $n < i < n+m$ and q is the location in the last p-control of $\mathfrak{p}_i$ ($q = q_0$ if $i = n+1$ and $\mathfrak{p}$ is empty).

Now, the probability measure $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}}$ is defined to be a unique extension of $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}}(Play(B))$ on $\mathcal{F}$ according to the extension theorem (see, e.g., [19]).

3.1.2 Region relation

Although the state-space of $\mathcal{G}_{\mathcal{A}}$ is uncountable, we can define a variant of *region relation* over p-histories which has a finite index, and then work with finitely many *regions*.

For a given $x \in \mathbb{R}_{\geq 0}$, we use $frac(x)$ to denote the fractional part of x, and $int(x)$ to denote the integral part of x. For $x, y \in \mathbb{R}_{\geq 0}$, we say that x and y *agree on integral part* if $int(x) = int(y)$ and neither or both x, y are integers. A *relevant bound* of a clock x is the largest constant c that appears in all guards. A *relevant bound* of an event e is u_e if $u_e < \infty$, and ℓ_e otherwise. We say that an element $a \in \mathcal{E} \cup \mathcal{X}$ is *relevant* for ξ if $\xi(a) \leq r$ where r is the relevant bound of a. Finally, we put $\xi_1 \approx \xi_2$ if

- for all relevant $a \in \mathcal{E} \cup \mathcal{X}$ we have that $\xi_1(a)$ and $\xi_2(a)$ agree on integral parts;
- for all relevant $a, b \in \mathcal{E} \cup \mathcal{X}$ we have that $frac(\xi_1(a)) \leq frac(\xi_1(b))$ if and only if $frac(\xi_2(a)) \leq frac(\xi_2(b))$.

The equivalence classes of $\approx$ are called *time areas*. Now we can define the promised *region relation* $\sim$ on p-histories. Let $\mathfrak{p}_1$ and $\mathfrak{p}_2$ be p-histories such that (c_1, q_1, ξ_1) is the last p-control of

$\mathfrak{p}_1$ and (c_2, q_2, ξ_2) is the last p-control of $\mathfrak{p}_2$. We put $\mathfrak{p}_1 \sim \mathfrak{p}_2$ iff $c_1 = c_2$, $q_1 = q_2$ and $\xi_1 \approx \xi_2$. Note that $\sim$ is an equivalence with a finite index. The equivalence classes of $\sim$ are called *regions*. A *target region* is a region that contains such p-histories whose last p-controls have a target location in the second component. The sets of all regions and target regions are denoted by $\mathcal{R}$ and $\mathcal{R}_T$, respectively.

Remark Let us note that the region construction described above can also be applied to configurations of timed automata, where it coincides with the standard region construction of [13].

This region construction is very useful as the region equivalence is a congruence with respect to one-step reachability. This is formalized in the following lemma and proved in Appendix A.1. We will often use this lemma in argumentation throughout the whole paper. In order to state the lemma, we introduce the following notation. Let $B = r_1 \cdots r_n$ be a sequence of regions, by $Play(B)$ we denote the set of p-runs that follow this sequence in the first n steps.

Lemma 3.3. *Let r and t be regions and $\mathfrak{p}, \bar{\mathfrak{p}}$ be p-histories of the region r. For every action a available in r, and every strategy σ_a and π_a choosing a in r, we have that*

$$\mathcal{P}_{\mathfrak{p}}^{\sigma_a,\pi_a}(Play(t)) > 0 \quad \text{iff} \quad \mathcal{P}_{\bar{\mathfrak{p}}}^{\sigma_a,\pi_a}(Play(t)) > 0.$$

3.1.3 Strategies in the product game

For every $\mathcal{S} \subseteq \mathcal{R}$, let $Reach(\mathcal{S})$ be the set of all p-plays that visit a region of $\mathcal{S}$ (i.e., some prefix of the p-play belongs to some $r \in \mathcal{S}$). We say that a strategy $\sigma \in \Sigma$ is *almost-sure winning* in $\mathcal{G}_{\mathcal{A}}$ for a p-history $\mathfrak{p}$ if for every $\pi \in \Pi$ we have that $\mathcal{P}_{\mathfrak{p}}^{\sigma,\pi}(Reach(\mathcal{R}_T)) = 1$. The relationship between almost-sure winning strategies in $\mathcal{G}$ and $\mathcal{G}_{\mathcal{A}}$ is formulated in the next proposition.

Proposition 3.4. *Let $\sigma \in \Sigma$ and $\mathfrak{p}$ be a p-history. Then σ is almost-sure winning for $\mathfrak{p}$ in $\mathcal{G}_{\mathcal{A}}$ iff σ is almost-sure winning for $H(\mathfrak{p})$ in $\mathcal{G}$.*

Proof. The proposition is a direct consequence of following observation. Let C be the set of runs that are accepted by the automaton $\mathcal{A}$. One can easily see that for any pair of strategies σ and π and any starting p-history $\mathfrak{p}$ the following equation holds.

$$\mathcal{P}_{\mathfrak{p}}^{\sigma,\pi}(Reach(\mathcal{R}_T)) = \mathcal{P}_{H(\mathfrak{p})}^{\sigma,\pi}(C)$$

Notice that in the definition of p-runs the valuation vectors ξ are defined to be consistent with the previous p-history of the p-run. $\square$

Another observation about strategies in $\mathcal{G}_{\mathcal{A}}$ which is heavily used in the next sections concerns *regional* strategies. Formally, a strategy $\tau \in \Sigma \cup \Pi$ is regional if for all p-histories $\mathfrak{p}_1$ and $\mathfrak{p}_2$ such that $\mathfrak{p}_1 \sim \mathfrak{p}_2$ we have that $\tau(\mathfrak{p}_1) = \tau(\mathfrak{p}_2)$.

Proposition 3.5. *Every regional strategy $\tau \in \Sigma \cup \Pi$ is a DTA strategy.*

Proof. Intuitively, we transform τ into a DTA $A_{\mathcal{G}_{\mathcal{A}}}$ whose regions are in one-to-one correspondence with the regions of $\mathcal{G}_{\mathcal{A}}$. The automaton $A_{\mathcal{G}_{\mathcal{A}}}$ reads a sequence of stamps of $\mathcal{G}$ and simulates the behavior of $\mathcal{G}_{\mathcal{A}}$. It has a special clock for every clock of $\mathcal{A}$ and every event of $\mathcal{E}$, and uses its locations to store also the current state of the game.

Each regional strategy of $G_{\mathcal{A}}$ determines a DTA strategy on automaton $A_{G_{\mathcal{A}}}$. Recall that for an automaton strategy we need an automaton over alphabet $S \cup \mathcal{E}$ that reads the history $(s_0 t_0 e_0) \cdots (s_n t_n e_n)$ where for each $0 \leq i \leq n$ the symbols s_i and e_i are input letters and t_i is a time stamp. We define $A_{G_{\mathcal{A}}} = (((S \cup C_\Box \cup C_\Diamond) \times Q) \cup \{q_{init}\}, S \cup \mathcal{E}, \mathcal{E} \cup \mathcal{X}, \hookrightarrow, q_{init}, \emptyset)$ where $\hookrightarrow = \hookrightarrow_{init} \cup \hookrightarrow_{event} \cup \hookrightarrow_{state}$. A guard that is satisfied for any valuation is denoted as $\mathtt{tt}$. The set of events not scheduled in s is denoted as $N(s)$. The sets of edges are defined as follows

$$
\begin{aligned}
\hookrightarrow_{init} &= \{\, (q_{init}, s, \mathtt{tt}, \emptyset, (s, q_0)) \mid s \in S\} \\
\hookrightarrow_{event} &= \{\, ((s,q), e, g, X \cup \{e\} \cup N(s), (c, q')) \mid \\
&\qquad s \in S, e \in \mathcal{E}, (q, L(e,s), g, X, q') \in \longrightarrow, c = F(s,e)\} \\
\hookrightarrow_{state} &= \{\, ((c,q), s, \mathtt{tt}, N(s), (s,q)) \mid c \in (C_\Box \cup C_\Diamond), q \in Q, s \in S\}
\end{aligned}
$$

For each p-history $\mathfrak{p}$ where x is the last p-stamp in $\mathfrak{p}$, it can be easily shown by induction on the length of $\mathfrak{p}$ that $\mathfrak{p}$ is in the same region as the configuration of the automaton $A_{G_{\mathcal{A}}}$ after reading the history $H(\mathfrak{p})$. Precisely, let $F_{\mathcal{A}}(x) = (c, q, \xi)$, the automaton will be in a configuration $((c,q), \xi)$ after reading the history $H(\mathfrak{p})$. Therefore, let σ be a strategy constant on regions. The strategy σ is according to Definition 2.3 a DTA strategy, as well. □

Note that due to Proposition 3.5, every regional strategy can be effectively transformed into a DTA strategy.

3.2 Almost-sure winning strategies

This section outlines the proof of our main result, namely that if player □ can win almost-surely, she can win almost-surely with a strategy that has a finite description.

Theorem 3.6. *Let $\mathfrak{p}$ be a p-history. If there is a strategy $\sigma \in \Sigma$ which is almost-sure winning in $\mathcal{G}_{\mathcal{A}}$ for $\mathfrak{p}$, then there is a DTA strategy $\sigma^* \in \Sigma$ which is almost-sure winning for $\mathfrak{p}$.*

Note that due to Proposition 3.5, it suffices to show that in $\mathcal{G}_{\mathcal{A}}$ there is a regional strategy that is almost-sure winning for $\mathfrak{p}$.

Now we provide a quick overview of the whole proof. We start with an observation that an almost-sure winning strategy must never reach with positive probability a bad region from that it has zero probability of coming to the target. In other words, from any p-history in a reachable region with positive probability it must have non-zero probability of winning. We call every strategy that satisfies this condition a *candidate strategy*. The first part of the proof is to show that there is a regional candidate strategy. But non-zero probability of winning is not sufficient, in the second part of the proof we show that every regional candidate strategy wins with probability 1.

Now, we define the notion of candidate strategy.

Definition 3.7. *Let σ be a strategy and $\mathfrak{p}$ a p-history. We define $\mathcal{S}(\sigma, \mathfrak{p})$ to be a set of regions $\{r \in \mathcal{R} \mid \exists \pi \in \Pi : \mathcal{P}^{\sigma,\pi}_{\mathfrak{p}}(Reach(r)) > 0\}$. We write $\mathfrak{p}' \in \mathcal{S}(\sigma, \mathfrak{p})$ to denote $\mathfrak{p}' \in \bigcup_{r \in \mathcal{S}(\sigma,\mathfrak{p})} r$.*

A strategy $\sigma \in \Sigma$ is a candidate for a p-history $\mathfrak{p}$ *if for every p-history $\mathfrak{p}' \in \mathcal{S}(\sigma, \mathfrak{p})$ and, for every $\pi \in \Pi$, we have that $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}'}(Reach(\mathcal{R}_T)) > 0$.*

The first part of the proof follows from the following proposition.

Proposition 3.8. *Let $\mathfrak{p}$ be a p-history. If there is a strategy $\sigma \in \Sigma$ which is almost-sure winning in $\mathcal{G}_{\mathcal{A}}$ for $\mathfrak{p}$, then there is a regional strategy $\sigma^* \in \Sigma$ candidate for $\mathfrak{p}$.*

The formal proof is in Appendix A.2, here we provide a brief intuition only. Observe that even σ may not be a candidate strategy for $\mathfrak{p}$. Indeed it may visit one part of a region with positive probability and the other part with probability zero. In the visited part of the region it must choose some good action (in order to win almost-surely) but in the non-visited part it may choose a bad action. If this bad action leads to zero probability of winning, σ is not a candidate strategy. Still, we have to mimic in some sense the behavior of the strategy σ by a regional strategy. The key idea is to choose in a region only such action that is *really* used by σ, i.e. an action that is in this region chosen by σ in runs with non-zero measure. It can be easily shown that in every region reachable by σ there must be some good action that leads to reaching the target (whereas other actions may postpone it). Since we need to find a regional strategy, we need to show that this good action is actually good for the whole region. This is guaranteed by Lemma 3.3 that may be regarded as a probabilistic variant of a classical result for reachability in timed automata. In more detail, the lemma states that if we reach some region in one step with positive probability, we reach it in one step with positive probability from *any* p-history that is in the same region as our current p-history. In other words, regions form a congruence

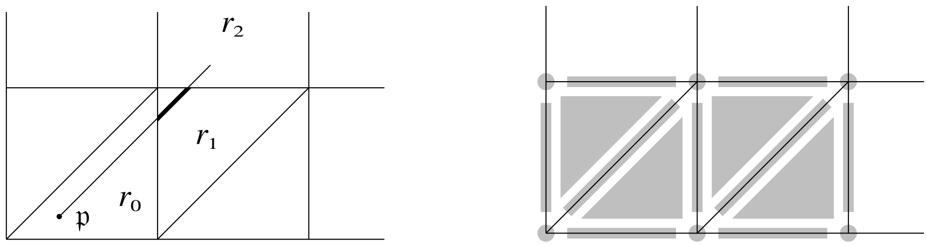

Figure 3: Intuition for δ-separated parts of regions.

with respect to forward reachability with positive probability. Therefore a good action achieves positive probability of winning for the whole region.

Yet, positive probability of winning of strategy σ^* guaranteed by Proposition 3.8 is not sufficient, in the following we need to show that σ^* wins almost-surely.

Remark If we consider the restricted case of one-player games with bounded intervals and exponentially distributed unbounded events, we can already easily prove that σ^* is almost-sure winning using [13] as follows. Fixing σ^* resolves all non-determinism and yields a system of the type considered by [13]. Since we are guaranteed the positive probability of reaching the target, we may apply Lemma 3 of [13]. However, in the setting of two-player games, we cannot use this argument directly and some (non-trivial) changes are required.

To finish the proof of the main theorem for two-player games, we show that every regional candidate strategy wins with probability 1. This proof is somewhat intricate. Note that we are guaranteed that for every p-history in a positively reachable region, the probability of reaching a target is positive. However, it can be arbitrarily small. Therefore, even if we pass through the reachable regions infinitely often, it is not clear that we eventually reach a target almost surely (because the probabilities of reaching the target may rapidly decrease as the play goes on). The almost-sure winning would be guaranteed if the probabilities were bounded from below by a positive constant. Such a bound does not exist in general, because the transition probabilities between two regions may be arbitrarily small. These probabilities approach zero as the starting p-history approaches the boundary of its region (i.e. as the fractional parts of two clocks approach each other). The left part of Figure 3 shows the region graph of a system with two clocks and a single state. There is also a single event, which is positive on $(0, 1)$ and its associated clock is not depicted. Now observe that if $\mathfrak{p}$ comes closer and closer to the diagonal, the probability that the (only) event happens in the region r_1 is smaller and smaller. Similarly, if $\mathfrak{p}$ comes closer and closer to the bottom or right boundary of the region, respectively, the probability that the event happens in the region r_2 or r_0 is smaller and smaller.

In order to bound from below the probabilities of reaching the target, we use a technique of Alur et al. [13]. We restrict ourselves to *δ-separated* parts of regions. As these parts are reached from all the remaining parts of regions (as shown later), this restriction is without loss of generality. The δ-separated parts of regions are depicted in gray in the right part of Figure 3. Here, we are at least δ-away from the boundary of the region. Being away from the boundary by a fixed δ then intuitively guarantees that any region that is reachable in one step is reachable with a probability bounded from below. Indeed, the possible waiting time that lead us to that region lies in an interval that has length at least δ, and the probability that an event happens during an interval of this minimal size is bounded from below. It may happen that some p-clocks synchronize and for some time have equal value (several clocks of the automaton may be reset with the same transition or several events are newly scheduled with the same transition or any combination of the previous two). The lower bound for one-step transitions is guaranteed even for situations where some p-clocks are synchronized. Therefore, we can properly define δ-separation as follows.

Definition 3.9. *Let $\delta > 0$. We say that a set $D \subseteq \mathbb{R}_{\geq 0}$ is δ-separated if for every $x, y \in D$ either $frac(x) = frac(y)$ or $|frac(x) - frac(y)| > \delta$. Further, we say that a p-history with the last p-control (s, q, ξ) is δ-separated if the set $\{0\} \cup \{\xi(a) \mid a \in \mathcal{E} \cup \mathcal{X}, a \text{ is relevant for } \xi\}$ is δ-separated.*

The reason for considering only relevant p-clocks is purely technical (intuitively, a p-clock is relevant if it has a sufficiently low value). We need to prove that a path of fixed length (to the target) from a δ-separated p-history has probability bounded from below. This can be iteratively shown using the one-step bounds but we need to keep after each step some separation (not necessarily δ-separation, δ'-separation is sufficient for any fixed $\delta \geq \delta' > 0$). Notice that some constraints put on a p-history to be δ-separated are not necessary for bounding the one-step probabilities but are necessary for guaranteeing the separation after each step. This reasoning leads to the following result, proven in Appendix A.3.

Proposition 3.10. *Let σ^* be a regional strategy candidate for a p-history $\mathfrak{p}$. For every $\delta > 0$ there is $\varepsilon > 0$ and $n \in \mathbb{N}$ such that for every δ-separated p-history $\mathfrak{p}' \in \mathcal{S}(\sigma^*, \mathfrak{p})$ and every strategy π we have that $\mathcal{P}^{\sigma^*,\pi}_{\mathfrak{p}'}(Reach^{\leq n}(\mathcal{R}_T)) > \varepsilon$ where $Reach^{\leq n}(\mathcal{R}_T)$ is the set of p-runs that reach the target within n steps.*

What happens if we after n steps leave the δ-separated parts of the regions? The proof is concluded by the following observation that regardless of the decisions of the players a δ-separated p-history is reached almost-surely.

Proposition 3.11. *There is $\delta > 0$ such that for every regional strategy $\sigma \in \Sigma$ and every $\pi \in \Pi$, a δ-separated p-history is reached almost surely from every p-history $\mathfrak{p}$.*

The proof is in Appendix A.4. This way, we repeat infinitely many times a chance to reach the target with probability bounded from below. Therefore, we eventually reach it almost surely whence Theorem 3.6 follows.

3.3 The algorithm

In this section, we show that the existence of a DTA almost-sure winning strategy is decidable in exponential time, and we also show how to compute such a strategy if it exists. Due to Proposition 3.4, this problem can be equivalently considered in the setting of the product game $\mathcal{G}_{\mathcal{A}}$. Due to Proposition 3.5, an almost-sure winning DTA strategy can be constructed as a strategy that is constant on every region of $\mathcal{G}_{\mathcal{A}}$. We show that this problem can be further reduced to the problem of computing wining strategies in a *finite* stochastic game $\mathcal{G}^{\mathcal{A}}$ with reachability objectives induced by the product game $\mathcal{G}_{\mathcal{A}}$. First, we define the game $\mathcal{G}^{\mathcal{A}}$ and show how to compute it. The complexity discussion of this reduction follows.

The product $\mathcal{G}_{\mathcal{A}}$ induces a game $\mathcal{G}^{\mathcal{A}}$ whose vertices are the regions of $\mathcal{G}_{\mathcal{A}}$ as follows. Player $\odot$, where $\odot \in \{\Box, \Diamond\}$, plays in regions $(c, q, [\xi]_{\approx})$ [1] where $c \in C_{\odot}$. In a region $r = (c, q, [\xi]_{\approx})$, she chooses an arbitrary action $a \in A(c)$ and this action a leads to a stochastic vertex $(r, a) = ((c, q, [\xi]_{\approx}), a)$. From this stochastic vertex there are transitions to all regions $r' = (c', q', [\xi']_{\approx})$, such that r' is reachable from all $\mathfrak{p} \in r$ in one step using action a with some positive probability in the product $\mathcal{G}_{\mathcal{A}}$. One of these probabilistic transitions is taken at random according to the uniform distribution. From the next region the play continues in the same manner. Player $\Box$ tries to reach the set $\mathcal{R}_T$ of target regions (which is the same as in the product game) and player $\Diamond$ tries to avoid it. We say that a strategy σ of player $\Box$ is almost-sure winning for a vertex v if she reaches $\mathcal{R}_T$ almost surely when starting from v and playing according to σ.

At first glance, it might seem surprising that we set all probability distributions in $\mathcal{G}^{\mathcal{A}}$ as uniform. Note that in different parts of a region r, the probabilities of moving to r' are different. Due to Lemma 3.3 (see Appendix A.1), they are all positive or all zero. Since we are interested only in *qualitative* reachability, this is sufficient for our purposes.

Moreover, note that since we are interested in non-zero probability behaviour, there are no transitions to regions which are reachable only with zero probability (such as when an event occurs at an integral time).

[1] Note that a region is a set of p-histories such that their last p-controls share the same control c, location q, and equivalence class $[\xi]_{\approx}$. Hence, we can represent a region by a triple $(c, q, [\xi]_{\approx})$.

Algorithm 1 Decide whether $\Box$ has a DTA almost-sure winning strategy for $\mathfrak{h}$

input: events $\mathcal{E}$, game $\mathcal{G} = (S, E, C_\Box, C_\Diamond, Act, F, A, \mu_0)$, DTA $\mathcal{A} = (Q, \Sigma, X, \longrightarrow, q_0, T)$, and the history $\mathfrak{h}$

output: YES + an almost-sure winning DTA strategy if it exists, or NO, otherwise

$\mathcal{G}^{\mathcal{A}} \leftarrow$ construct the discrete game by Algorithm 2 in Appendix B.2
$W \leftarrow$ compute the set of almost-sure winning vertices in $\mathcal{G}^{\mathcal{A}}$ by [21]
$\sigma \leftarrow$ compute the optimal positional strategy in $\mathcal{G}^{\mathcal{A}}$ by [21]
$v \leftarrow$ compute the vertex in $\mathcal{G}^{\mathcal{A}}$ such that the run of $\mathcal{A}$ over $\mathfrak{h}$ results in the region v in $\mathcal{G}^{\mathcal{A}}$
if $v \in W$ **then**
  return YES + the DTA strategy σ' induced by the regional strategy σ
else
  return NO

The following proposition states the correctness of the reduction. Note that a regional strategy for the product game $\mathcal{G}_{\mathcal{A}}$ induces a unique positional strategy for the game $\mathcal{G}^{\mathcal{A}}$, and vice versa. Indeed, this bijection between strategies is given by the bijection between regions of the product game $\mathcal{G}_{\mathcal{A}}$ and the vertices of the game $\mathcal{G}^{\mathcal{A}}$. Slightly abusing the notation, we thus consider the strategies to be strategies in both games.

Proposition 3.12. *Let $\mathcal{G}$ be a game and $\mathcal{A}$ be a deterministic timed automaton. For every p-history $\mathfrak{p}$ in a region r, a positional strategy is almost-sure winning for r in $\mathcal{G}^{\mathcal{A}}$ iff it is almost-sure winning for $\mathfrak{p}$ in $\mathcal{G}_{\mathcal{A}}$,*

The Algorithm 1 constructs the regions of the product $\mathcal{G}_{\mathcal{A}}$ and the induced game graph of the game $\mathcal{G}^{\mathcal{A}}$ (see Appendix B.2). Since there are exponentially many regions (w.r.t. the number of clocks and events), the size of $\mathcal{G}^{\mathcal{A}}$ is exponential in the size of $\mathcal{G}$ and $\mathcal{A}$. Note that two-player stochastic games with qualitative reachability objectives are easily solvable in polynomial time [21]. Due to the bijection H (mapping p-histories to histories), Theorem 3.6, and Proposition 3.12, there is an almost-sure winning strategy for $\mathfrak{h}$ in $\mathcal{G}$ with $\mathcal{A}$ iff there is an almost-sure winning strategy for $\mathfrak{p} = H^{-1}(\mathfrak{h})$ in $\mathcal{G}_{\mathcal{A}}$ iff there is an almost-sure winning strategy for $r \ni \mathfrak{p}$ in $\mathcal{G}^{\mathcal{A}}$. Since all transformations of the strategies are trivially effective, we obtain the following theorem and thus conclude the proof of Theorem 3.2.

Theorem 3.13. *Let $\mathcal{G}$ be a SRTG, $\mathcal{A}$ be a deterministic timed automaton, and $\mathfrak{h}$ be a history. The problem whether player $\Box$ has a (DTA) almost-sure winning strategy for $\mathfrak{h}$ is solvable in time* *exponential in $|\mathcal{G}|$ and $|\mathcal{A}|$. A DTA almost-sure winning strategy is computable in exponential time if it exists.*

4 Conclusions and Future Work

An interesting question is whether the positive results presented in this paper can be extended to more general classes of objectives that can be encoded, e.g., by deterministic timed automata with ω-regular acceptance conditions. Another open problem are algorithmic properties of ε-optimal strategies in stochastic real-time games.

References

[1] P. Haas, G. Shedler, Regenerative generalized semi-Markov processes, Stochastic Models 3 (3) (1987) 409–438.

[2] R. Schassberger, Insensitivity of steady-state distributions of generalized semi-Markov processes, Advances in Applied Probability 10 (1978) 836–851.

[3] W. Whitt, Continuity of generalized semi-Markov processes, Mathematics of Operations Research 5 (4) (1980) 494–501.

[4] R. Alur, C. Courcoubetis, D. Dill, Verifying automata specifications of probabilistic real-time systems, in: Real-Time: Theory in Practice, Vol. 600 of LNCS, Springer, 1992, pp. 28–44.

[5] J. Norris, Markov Chains, Cambridge University Press, 1998.

[6] T. Brázdil, V. Forejt, J. Krčál, J. Křetínský, A. Kučera, Continuous-time stochastic games with time-bounded reachability, in: Proceedings of FST&TCS 2009, Vol. 4 of LIPIcs, Schloss Dagstuhl, 2009, pp. 61–72.

[7] M. Rabe, S. Schewe, Optimal time-abstract schedulers for CTMDPs and Markov games, in: Eighth Workshop on Quantitative Aspects of Programming Languages, 2010.

[8] A. Maitra, W. Sudderth, Finitely additive stochastic games with Borel measurable payoffs, Int. Jour. of Game Theory 27 (1998) 257–267.

[9] S. Ross, Stochastic Processes, Wiley, 1996.

[10] D. Bertsekas, Dynamic Programming and Optimal Control, Athena Scientific, 2007.

[11] C. Baier, H. Hermanns, J.-P. Katoen, B. Haverkort, Efficient computation of time-bounded reachability probabilities in uniform continuous-time Markov decision processes, TCS 345 (2005) 2–26.

[12] M. Neuhäußer, M. Stoelinga, J.-P. Katoen, Delayed nondeterminism in continuous-time Markov decision processes, in: Proceedings of FoSSaCS 2009, Vol. 5504 of LNCS, Springer, 2009, pp. 364–379.

[13] R. Alur, D. Dill, A theory of timed automata, TCS 126 (2) (1994) 183–235, fundamental Study.

[14] C. Baier, N. Bertrand, P. Bouyer, T. Brihaye, M. Größer, Almost-sure model checking of infinite paths in one-clock timed automata, in: Proceedings of LICS 2008, IEEE, 2008, pp. 217–226.

[15] N. Bertrand, P. Bouyer, T. Brihaye, N. Markey, Quantitative model-checking of one-clock timed automata under probabilistic semantics, in: Proceedings of 5th Int. Conf. on Quantitative Evaluation of Systems (QEST'08), IEEE, 2008, pp. 55–64.

[16] P. Bouyer, V. Forejt, Reachability in stochastic timed games, in: Proceedings of ICALP 2009, Vol. 5556 of LNCS, Springer, 2009, pp. 103–114.

[17] R. Alur, C. Courcoubetis, D. Dill, Model-checking for probabilistic real-time systems, in: Proceedings of ICALP'91, Vol. 510 of LNCS, Springer, 1991, pp. 115–136.

[18] T. Chen, T. Han, J.-P. Katoen, A. Mereacre, Quantitative model checking of continuous-time Markov chains against timed automata specifications, in: Proceedings of LICS 2009, IEEE, 2009, pp. 309–318.

[19] P. Billingsley, Probability and Measure, Wiley, 1995.

[20] D. Martin, The determinacy of Blackwell games, Journal of Symbolic Logic 63 (4) (1998) 1565–1581.

[21] L. de Alfaro, T. Henzinger, O. Kupferman, Concurrent reachability games, Tech. Rep. UCB/ERL M98/33, EECS Department, University of California, Berkeley (1998).
URL `http://www.eecs.berkeley.edu/Pubs/TechRpts/1998/3449.html`

A Deterministic timed automaton strategies - appendix proofs

A.1 Proof of Lemma 3.3

Lemma 3.3. *Let r and t be regions and $\mathfrak{p}, \bar{\mathfrak{p}}$ be p-histories of the region r. For every action a available in r, and every strategy σ_a and π_a choosing a in r, we have that*

$$\mathcal{P}_{\mathfrak{p}}^{\sigma_a,\pi_a}(Play(t)) > 0 \quad \textit{iff} \quad \mathcal{P}_{\bar{\mathfrak{p}}}^{\sigma_a,\pi_a}(Play(t)) > 0.$$

Proof. Assume $\mathcal{P}_{\mathfrak{p}}^{\sigma_a,\pi_a}(Play(t)) > 0$, we must show that also $\mathcal{P}_{\bar{\mathfrak{p}}}^{\sigma_a,\pi_a}(Play(t)) > 0$. The other direction follows from symmetry.

We first introduce a few definitions. Let $u = (c, q, [\xi]_\approx)$ be a region (recall, that a region is a set of p-histories such that their last p-controls have the same control c, location q and p-vector in the same time area $[\xi]_\approx$). To simplify the argumentation, we assume w.l.o.g. that the flow function F in the game G is injective. If it is not the case we can duplicate controls that have multiple preimages (and give them the same actions). Hence, let the flow function F be injective. Then for the region u with the control c there is a unique state s denoted as $State(u)$ and a unique event $e \in \mathcal{E}$ denoted as $Event(u)$ such that $F(s,e) = c$.

Let $r = (c_r, q_r, [\xi_r]_\approx)$ be the starting region and let (c_r, q_r, ξ) and $(c_r, q_r, \bar{\xi})$ be the last p-controls of $\mathfrak{p}$ and $\bar{\mathfrak{p}}$, respectively. Notice that $\xi, \bar{\xi} \in [\xi_r]_\approx$. We can understand the transition from (c_r, q_r, ξ) and $(c_r, q_r, \bar{\xi})$ to the target region $t = (c_t, q_t, [\xi_t]_\approx)$ as a sequence of three sub-steps:

- taking the fixed action a, moving to the state $s = State(t)$, resetting the events $\mathcal{E} \setminus E(s)$ not scheduled in s, and resetting a set of clocks X by taking the edge $(q_r, L(s), g, X, q_t)$ such that g is satisfied for all p-vectors in $[\xi_r]_\approx$,
- waiting for time w and $\bar{w}$, respectively, and
- triggering and resetting the event $e = Event(t)$ and moving to the control c_t.

As regards the action step, the probability of performing this step is $a(s)$ for any p-vector in $[\xi_r]_\approx$. It must hold that $a(s) > 0$. Let $R = X \cup (\mathcal{E} \setminus E(s))$ be the set of p-clocks reset in this step. We denote $\xi_a = \xi[R := \vec{0}]$ and $\bar{\xi}_a = \bar{\xi}[R := \vec{0}]$. Because $\xi \approx \bar{\xi}$ then obviously $\xi_a \approx \bar{\xi}_a$ as well.

By the time step we must move to some p-vector ξ_b from which it is possible to move to the target region by resetting the event e. Now comes a crucial observation that the time areas form an equivalence with respect to the forward reachability by a time step, in some sense. Let $[\xi_a]_\approx$

and $[\xi_b]_\approx$ be time areas and $\xi_a, \bar{\xi}_a \in [\xi_a]_\approx$ be p-vectors. The set of possible waiting times to reach the time area $[\xi_b]_\approx$ from the p-vector ξ_a is denoted by $X(\xi_a, [\xi_b]_\approx) = \{t \in \mathbb{R}_{>0} \mid \xi_a +_s t \in [\xi_b]_\approx\}$. It can be easily checked that the set $X(\xi_a, [\xi_b]_\approx)$ has positive Lebesgue measure if and only if the set $X(\bar{\xi}_a, [\xi_b]_\approx)$ has positive Lebesgue measure.

Before the time step we have two p-vectors, ξ_a and $\bar{\xi}_a$ in the same time area both "reached" with positive probability. Let A be the set of p-vectors such that the event e is triggered and the game moves to the desired control c_t if and only if the time step ends up in a p-vector from this set A. If we show that A is in fact a union of a set of time areas B, we are almost done. Because $\mathcal{P}^{\sigma_a,\pi_a}_{\mathfrak{p}}(Play(t)) > 0$, for some time area $[\xi_b]_\approx \in B$ it must hold that the set $X(\xi_a, [\xi_b]_\approx)$ has positive Lebesgue measure. Due to the previous observation also the set $X(\bar{\xi}_a, [\xi_b]_\approx)$ has positive Lebesgue measure. Because the density function f_e is positive on (l_e, u_e) we have that $\mathcal{P}^{\sigma_a,\pi_a}_{\bar{\mathfrak{p}}}(Play(t)) > 0$.

Now it remains to show that A is a union of time areas. First, the lower and upper bounds of the events give us a set of time areas C where the event e can be triggered (thanks to the fact that the events have their p-clocks in the p-vector). Second, the p-vectors from that we reach the target time area $[\xi_t]_\approx$ by the reset of e form a set of whole time areas D. Clearly, $A = \bigcup(C \cup D)$ is a union of time areas.

□

A.2 Proof of Proposition 3.8

Proposition 3.8. *Let $\mathfrak{p}$ be a p-history. If there is a strategy $\sigma \in \Sigma$ which is almost-sure winning in $\mathcal{G}_\mathcal{A}$ for $\mathfrak{p}$, then there is a regional strategy $\sigma^* \in \Sigma$ candidate for $\mathfrak{p}$.*

We proceed in two steps:

- Firstly, we show that every regional strategy σ' that selects only the actions used "significantly" by σ, stays in $\mathcal{S}(\sigma, \mathfrak{p})$ almost surely, i.e. $\mathcal{S}(\sigma', \mathfrak{p}) \subseteq \mathcal{S}(\sigma, \mathfrak{p})$. We firstly define these significantly used actions. For every region $r \in \mathcal{S}(\sigma, \mathfrak{p})$ and action a, let $p\text{-}hist(r, a)$ be the set of all p-histories ending in r where, moreover, σ assigns a positive probability to a. We then denote by A_r the set of all $a \in Act$ for which there is $\pi \in \Pi$ such that $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}}(p\text{-}hist(r, a)) > 0$.

  Claim A.1. *Let σ' be a regional strategy that selects only the actions from A_r in every $r \in \mathcal{S}(\sigma, \mathfrak{p})$. Then for all $\pi \in \Pi$ and $\mathfrak{p}' \in \mathcal{S}(\sigma, \mathfrak{p})$ it holds that $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(Reach(\mathcal{R} \setminus \mathcal{S}(\sigma, \mathfrak{p}))) = 0$.*

To see this, realize that from Lemma 3.3, we do not visit (with positive probability) any other regions than we did with σ, when we use only the actions of A_r. Hence, we stay in $\mathcal{S}(\sigma, \mathfrak{p})$ almost surely. For details, see Appendix A.2.1.

- Secondly, we build a regional strategy σ^* that can reach a target region with positive probability from every p-history of $\mathcal{S}(\sigma, \mathfrak{p})$. We proceed in iterations. In each iteration we pick some region and show that from this region there is a short path to the target. Furthermore, we can guarantee a positive probability of following this path by a regional strategy. If we change the behavior of the strategy outside this path it does not change the probability of following it and thus reaching the target. Therefore, for the next iteration, we add to the set of target regions the region we picked in the previous iteration and the regions on the short path to the target (because from all these regions we can reach the set $\mathcal{R}_T$ with positive probability) and repeat the whole process until there are no more non-target regions.

  For the first iteration, let the set of target regions $X = \mathcal{R}_T$. In each iteration:

  – Let us fix a region $r \in \mathcal{S}(\sigma, \mathfrak{p})$ such that $r \notin X$. Realize that then there is a p-history $\mathfrak{p}' \in r$ for which σ is almost-sure winning (since σ is almost-sure winning and for every $r \in \mathcal{S}(\sigma, \mathfrak{p})$ there is $\pi \in \Pi$ such that r is visited with positive probability, there must be a p-history $\mathfrak{p}' \in r$ for which σ is almost-sure winning). In particular, $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}'}(Reach(X)) > 0$ for every $\pi \in \Pi$.

  – We show how to transform σ into a regional strategy σ' such that $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(Reach(X)) > 0$.

    Claim A.2. *Let r be a region of $\mathcal{S}(\sigma, \mathfrak{p})$, $p' \in r$ a p-history, and X a set of target regions such that $\mathcal{R}_T \subseteq X$ and $r \notin X$. There is a regional strategy σ' that selects only actions from $A_{r'}$ in every $r' \in \mathcal{S}(\sigma, \mathfrak{p})$ such that for every strategy $\pi \in \Pi$ we have a finite sequence B of regions ending in X satisfying $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(Play(B)) > 0$ and $|B| \leq |\mathcal{R}|$.*

    Here, we provide the intuition only, the full proof of this claim is in Appendix A.2.2. We start with the intuition for one-player games, i.e., the situation when $C_\Diamond = \emptyset$. Then there must be a sequence of regions $B = r_0 \cdots r_n$ visited on the way from $\mathfrak{p}'$ to a target, selecting some actions $a_0, \ldots, a_{n-1}$. We fix these actions for the respective regions (if some region is visited several times, we fix the last action taken) and thus obtain the desired regional strategy σ'.

    In the general case of two-player games, we have to consider a tree of regions and actions instead of a single sequence, because every possible behaviour of the opponent in the first n steps has to be taken into account. That is the reason for having for every strategy of the opponent a different sequence of regions B.

- We have the positive probability guaranteed only for the p-history $\mathfrak{p}'$, we need it for the whole region r and all the regions in the sequence to the target. Let us fix a strategy π of the opponent. There is a sequence of regions $B = r_0 \cdots r_n$ ending in X such that $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(Play(B)) > 0$. We want to prove that for any $0 \leq i < n$ and any p-history $\mathfrak{p}''$ in the region r_i the probability to reach the set X is positive. We show by induction on i. As regards the base of the induction, the probability to move from the region r_{n-1} to the region r_n is positive or zero for all p-histories in r_{n-1} due to Lemma 3.3. If it were zero then clearly $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(Play(B)) = 0$. Hence, for all $\mathfrak{p}'' \in r_{n-1}$ it holds that $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(Reach(X)) > 0$. Now, assume the statement holds for $i+1$, we show it for i. Let $\mathfrak{p}'' \in r_i$. The probability to reach X from $\mathfrak{p}''$ is at least the probability to move to r_{i+1} and reach the target from there. The probability to move to r_{i+1} must be positive for some p-history in r_i because $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(Play(B)) > 0$. Hence, it is positive also for $\mathfrak{p}''$ due to Lemma 3.3. Thanks to the induction hypothesis we reach the target from any p-history in r_{i+1} with positive probability. In total, we reach X from $\mathfrak{p}''$ with positive probability.

  We have shown that for any region in the sequence B and any p-history in this region the probability to reach X is positive. By the same arguments as in the induction step, we get positive probability to reach X for any $\mathfrak{p}'' \in r$, also.

- Let Y be the set of all regions used by σ' to reach the target. I.e. $r' \in Y$ iff $r' \notin X$ and there is a strategy π such that r' is in the sequence B provided by Claim A.2 for π or $r' = r$. For any p-history $\mathfrak{p}' \in Y$ and any $\pi \in \Pi$ we have $\mathcal{P}^{\sigma',\pi}_{\mathfrak{p}'}(A) > 0$ where A is a set of runs that stay in the set of regions Y until they reach a region from X. We put $\sigma^*(\mathfrak{p}') = \sigma'(\mathfrak{p}')$ for all $\mathfrak{p}' \in Y$. For the next iteration we set $X := X \cup Y$.

We repeat these iterations until $X = \mathcal{S}(\sigma, \mathfrak{p})$. It is easy to see that the iteratively constructed strategy σ^* can reach $\mathcal{R}_T$ with positive probability from any p-history in $\mathcal{S}(\sigma, \mathfrak{p})$. Because of the first step of the proof, we have $\mathcal{S}(\sigma^*, \mathfrak{p}) \subseteq \mathcal{S}(\sigma, \mathfrak{p})$. Hence, σ^* is a candidate strategy for $\mathfrak{p}$.

□

A.2.1 Proof of Claim A.1

Recall that we have fixed a strategy σ almost-sure winning for a p-history $\mathfrak{p}$.

Claim A.1. *Let σ' be a regional strategy that selects only the actions from A_r in every $r \in \mathcal{S}(\sigma, \mathfrak{p})$. Then for all $\pi \in \Pi$ and $\mathfrak{p}' \in \mathcal{S}(\sigma, \mathfrak{p})$ it holds that $\mathcal{P}_{\mathfrak{p}'}^{\sigma',\pi}(\mathit{Reach}(\mathcal{R} \setminus \mathcal{S}(\sigma, \mathfrak{p}))) = 0$.*

Proof. For a contradiction, let us assume that there is a region $s \notin \mathcal{S}(\sigma, \mathfrak{p})$ such that $\mathcal{P}_{\mathfrak{p}'}^{\sigma',\pi}(\mathit{Reach}(s)) > 0$ for some $\mathfrak{p}' \in \mathcal{S}(\sigma, \mathfrak{p})$ and $\pi \in \Pi$. Therefore, we can prove that there is a sequence of regions $r_1 \cdots r_{n-1}s$ such that $\mathcal{P}_{\mathfrak{p}'}^{\sigma',\pi}(\mathit{Play}(r_1 \cdots r_{n-1}s)) > 0$. Indeed, for all $i \in \mathbb{N}$, let S_i denote the set of runs that reach s in the ith step and let n be the smallest index for which $\mathcal{P}_{\mathfrak{p}'}^{\sigma',\pi}(S_n) > 0$. We partition S_n according to the sequence of regions visited in the first n steps. Thus, we obtain finitely many equivalence classes, at least one of which is of non-zero $\mathcal{P}_{\mathfrak{p}'}^{\sigma',\pi}$-measure. This class corresponds to a sequence of regions $r_1 \cdots r_n s$.

Let v be the first region in this sequence not in $\mathcal{S}(\sigma, \mathfrak{p})$. Further, if $v \neq r_1$ then let u be the region preceding v, else let u be the region containing $\mathfrak{p}'$. Thus u is the last region in $\mathcal{S}(\sigma, \mathfrak{p})$. Let a be the action chosen by σ' in u. Hence also $a \in A_u$, and there must be a set of runs with positive $\mathcal{P}_{\mathfrak{p}}^{\sigma,\pi}$-measure (for some $\pi \in \Pi$) where σ gives positive weight to an action a and thus by Lemma 3.3 leading with positive probability to v. Altogether, we get a contradiction with $v \notin \mathcal{S}(\sigma, \mathfrak{p})$. ◻

A.2.2 Proof of Claim A.2

Recall that we have fixed a strategy σ almost-sure winning for a p-history $\mathfrak{p}$.

Claim A.2. *Let r be a region of $\mathcal{S}(\sigma, \mathfrak{p})$, $p' \in r$ a p-history, and X a set of target regions such that $\mathcal{R}_T \subseteq X$ and $r \notin X$. There is a regional strategy σ' that selects only actions from $A_{r'}$ in every $r' \in \mathcal{S}(\sigma, \mathfrak{p})$ such that for every strategy $\pi \in \Pi$ we have a finite sequence B of regions ending in X satisfying $\mathcal{P}_{\mathfrak{p}'}^{\sigma',\pi}(\mathit{Play}(B)) > 0$ and $|B| \leq |\mathcal{R}|$.*

Proof. Firstly, we prove the lemma for one-player games since the arguments are simpler. Then we prove the lemma for games in general.

For all i, we denote the set of runs that reach X in the ith step by X_i. Let n be the smallest index for which $\mathcal{P}_{\mathfrak{p}'}^{\sigma'}(X_n) > 0$. Indeed, there is such an n, since $\mathcal{P}_{\mathfrak{p}'}^{\sigma'}(\mathit{Reach}(\mathcal{R}_T)) > 0$ and $\mathcal{R}_T \subseteq X$.

We partition X_n according to the sequence of regions visited in the first n steps. Thus, we obtain finitely many parts, at least one of which is of non-zero $\mathcal{P}_{\mathfrak{p}'}^{\sigma'}$-measure. This part is represented by a sequence of regions $r_0 r_1 \cdots r_n$, where $r_n \in X$.

For all $i < n$, let a_i be an action chosen by σ on a set of runs of non-zero $\mathcal{P}_{\mathfrak{p}}^{\sigma}$-measure somewhere in r_i and when performed (note that σ is randomizing) the successor will be the state of r_{i+1} with non-zero probability.

For every $0 \leq i < n$ let $last(r_i)$ be the index of the last occurrence of r_i in the sequence of regions, i.e. $last(r_i) = \max\{j \mid r_j = r_i\}$ (note that σ is history-dependent). Let σ' be a DTA strategy that for all $0 \leq i < n$ chooses in region r_i the action $a_{last(r_i)}$, i.e. the action that has been chosen in r_i as the last one. This induces a sequence of regions and actions $s_0 = r_0, b_0 = a_{last(s_0)}, s_1 = r_{last(s_0)+1}, a_{last(s_1)}, s_2 = r_{last(s_1)+1}, \ldots, s_k = r_n$ where cycles on regions are eliminated. We set $B = s_0 \cdots s_k$.

By applying k times Lemma 3.3 we get that $\mathcal{P}^{\sigma'}_{\mathfrak{p}'}(Play(B)) > 0$.

We now extend this approach to games in general.

For all i, let X_i be the set of runs that reach X in the ith step. Since $\inf_{\pi\in\Pi} \mathcal{P}^{\sigma,\pi}_{\mathfrak{p}'}(Reach(X)) > 0$ it follows (using Lemma 3.3) that there is $n \in \mathbb{N}$ with $\inf_{\pi\in\Pi} \mathcal{P}^{\sigma,\pi}_{\mathfrak{p}'}(X_n) > 0$, i.e. for any opponent's strategy we have a guarantee to reach the target within n steps with positive probability using σ. Indeed, if there was a sequence of strategies $\pi_i \in \Pi$ such that for all $i \in \mathbb{N}$ we would have $\mathcal{P}^{\sigma,\pi_i}_{\mathfrak{p}'}(\bigcup_{0\leq j\leq i} X_j) = 0$ it would be easy to construct π such that $\mathcal{P}^{\sigma,\pi}_{\mathfrak{p}'}(\bigcup_{j\in\mathbb{N}_0} X_j) = 0$.

For each region s and $0 \leq i \leq n$, let $S_{s,i}$ be the set of runs from X_n that at the i-th position visit s. We define a success rate of a region s on i-th position as $\mathrm{sr}(s,i) = \inf_{\pi\in\Pi} \mathcal{P}^{\sigma,\pi}_{\mathfrak{p}'}(S_{s,i})$. Clearly, $s(r,0) > 0$. We say that an action a is *good for a region s in i-th position* if for at least one of a's target t we have positive success rate $\mathrm{sr}(t, i+1)$. If player $\Box$ controls a region $s \notin X$ with positive success rate on a position i, at least one of the enabled action must be good for the region s in the i-th position. Similarly, if player $\Diamond$ controls a region $s \notin X$ with positive success rate on a position i, all of his actions must be good for the region s in the i-th position.

We define σ' to take good action wherever possible and if more good actions are enabled in a region then we choose the one which is good for this region in the highest position.

For a fixed strategy π we can find the sequence B by taking the sequence of regions induced by π and σ' with good actions starting from $\mathfrak{p}'$ till X.

And again, we conclude by applying the Lemma 3.3 as above. $\Box$

A.3 Proof of Proposition 3.10

Proposition 3.10. *Let σ^* be a regional strategy candidate for a p-history $\mathfrak{p}$. For every $\delta > 0$ there is $\varepsilon > 0$ and $n \in \mathbb{N}$ such that for every δ-separated p-history $\mathfrak{p}' \in \mathcal{S}(\sigma^*, \mathfrak{p})$ and every strategy π we have that $\mathcal{P}^{\sigma^*,\pi}_{\mathfrak{p}'}(Reach^{\leq n}(\mathcal{R}_T)) > \varepsilon$, where $Reach^{\leq n}(\mathcal{R}_T)$ is the set of p-runs that reach the target within n steps.*

First notice that the region relation allows us to make the following abstract view on the system. In any region r one of the players has a set of available actions. For any action a there is a set of successor regions such that the probability of moving from r into a successor region after taking action a is positive no matter where in r we started. This holds because regions form a congruence with respect to the positive forward reachability in one step due to Lemma 3.3.

We denote these successor regions after taking action a as *a-successor* regions. Another observation proved in Appendix A.2.2 shows that for any p-history $\mathfrak{p}' \in \mathcal{S}(\sigma^*, \mathfrak{p})$ and any strategy π there is a sequence of regions $B = r_1 r_2 \cdots r_n$ such that $n \leq |\mathcal{R}|$, $r_n \in \mathcal{R}_T$, and $\mathcal{P}_{\mathfrak{p}'}^{\sigma^*,\pi}(Play(B)) > 0$. This allows us to define the *distance* of a region r from target regions as the minimal length n such that for every p-history $\mathfrak{p}' \in r$ and every strategy π there is a sequence of regions $B = r_1 r_2 \cdots r_n$ such that $n \leq |\mathcal{R}|$, $r_n \in \mathcal{R}_T$, and $\mathcal{P}_{\mathfrak{p}'}^{\sigma^*,\pi}(Play(B)) > 0$. Notice that every region of $\mathcal{S}(\sigma^*, \mathfrak{p})$ has distance lower or equal to $|\mathcal{R}|$. Let r be any region, and n be its distance. Observe that if r is a region of player $\Box$, there must be an action a such that some a-successor of r has distance $n - 1$. If r is a region of player $\Diamond$, for each action a there must be an a-successor of r with distance lower than n.

First of all, we show that for any action a and any a-successor t the probability of moving to t is bounded from below if we start in a δ-separated p-history. To be able to use the same argument for another step, we need to stay in the separated parts of the regions. It is not possible to keep the same separation, but it is sufficient to keep in each step a fixed fraction of the separation (we provide the bound for $1/3$ of δ).

Lemma A.3. *For each $\delta > 0$ there is $b > 0$ such that the following holds. For any regions r and t, any action a available in r, and every δ-separated p-history $\mathfrak{p}$ of r, we have*

$$\mathcal{P}_{\mathfrak{p}}^{\sigma_a,\pi_a}(Play(t)) > 0 \quad \Longrightarrow \quad \mathcal{P}_{\mathfrak{p}}^{\sigma_a,\pi_a}(Play_{\delta/3}(t)) > b$$

where σ_a and π_a choose a in r and $Play_{\delta/3}(t)$ is the set of p-runs that start in $(\delta/3)$-separated part of t.

This Lemma is proven in Appendix A.3.1. For a region r of player $\Box$ there must be an action a such that some a-successor t has lower distance. The strategy σ^* must take any such action a. Therefore, we have the probability bound b on moving from the δ-separated part of r to the $(\delta/3)$-separated part of a region with lower distance. Similarly for a region r of player $\Diamond$ and for any action a there must be an a-successor t_a with lower distance. The probability of moving to each t_a by action a is bounded by b. Any strategy π cannot decrease the probability of moving to a region with lower distance below the value b by mixing the available actions when starting in a δ-separated $\mathfrak{p} \in r$.

Now we can conclude the proof. For any π there is in each step a bounded probability of decreasing the distance and keeping the third of separation. Since the distance of regions in

$S(\sigma^*, \mathfrak{p})$ is bounded by $|\mathcal{R}|$, there is for an initial separation δ clearly a probability bound ε such that reaching the target in $|\mathcal{R}|$ steps is bounded by ε from below for any strategy π of the opponent.

A.3.1 Proof of Lemma A.3

Lemma A.3. *For each $\delta > 0$ there is $b > 0$ such that the following holds. For any regions r and t, any action a available in r, and every δ-separated p-history $\mathfrak{p}$ of r, we have*

$$\mathcal{P}^{\sigma_a,\pi_a}_{\mathfrak{p}}(Play(t)) > 0 \quad \Longrightarrow \quad \mathcal{P}^{\sigma_a,\pi_a}_{\mathfrak{p}}(Play_{\delta/3}(t)) > b$$

where σ_a and π_a choose a in r and $Play_{\delta/3}(t)$ is the set of p-runs that start in $(\delta/3)$-separated part of t.

Proof. Let $\delta > 0$, r and t be regions, $\mathfrak{p}$ be a δ-separated p-history in r, and a be an action available in r. Furthermore, we assume $\mathcal{P}^{\sigma_a,\pi_a}_{\mathfrak{p}}(Play(t)) > 0$. There must be some state s with $a(s) > 0$ through which we can reach t. It means that the flow of time from $\mathfrak{p}$ after moving to s has an intersection with some region r' such that triggering some event e in r' leads us in the target region t. Let I be this intersection time interval. The probability that event e is triggered in I must be positive. There are two cases, the interval I may be either bounded or unbounded.

First, we discuss the bounded case. Let $I = (u, v)$. Notice that u and v lie on adjacent region borders (i.e. there are p-clocks x and y such that $x + u$ and $y + v$ are integral values and no other p-clock has fractional value between x and y). Because $\mathfrak{p}$ is δ-separated, we have that $|u - v| \geq \delta$. It is easy to see that all p-histories reached in the interval $I' = (u + \delta/3, v - \delta/3)$ are $(\delta/3)$-separated. Indeed, the zero fractional value is put in an interval between x and y that is $\delta/3$ separated from both of them and the flow of time does not change the differences between other p-clocks.

The probability of moving after action a to the state s is fixed, $a(s) = p > 0$. We show that the probability of triggering the event e in the time interval I' is bounded from below. Let ξ be the current p-vector after moving to s. Notice that the upper bound of any event e' scheduled in s must not be closer than at the distance v from $\xi(e')$ because the only previous integral bound is at u and the probability of triggering e in (u, v) would then be zero. The probability of triggering e in I' can be underapproximated by the probability that e is triggered in I' and all other scheduled events are triggered in $(v - \delta/3, v)$. Recall that the events are independent. For each event e' scheduled in s we require it to be triggered in an interval of length at least $\delta/3$. For an event e' there is obviously a probability bound $b(e') > 0$ such that $\min_{x \in (l, M-\delta/3)} \int_x^{x+\delta/3} f_{e'}(t)dt \geq b(e')$ where l is the lower bound of e' and M is the maximal relevant bound in the system.

This probability bound also holds for the conditioned density which is actually the same as the unconditioned density but only shifted and scaled up. Therefore, integral over any interval of fixed length contained in the positive range of the conditioned density is greater or equal to $b(e')$. Hence, the probability that e is triggered in I' is greater than $b = \prod_{e \in \mathcal{E}} b(e')$ and $\varepsilon = p \cdot b > 0$.

Second, we discuss the unbounded case, i.e. the region t is the corner region that is unbounded in all dimension. Let $I = (u, \infty)$ for some $u \in \mathbb{R}_{\geq 0}$. Notice that any p-history in the region t is δ-separated for any δ because all the p-clocks are beyond their relevant bounds. The probability of triggering an event in the interval I is bounded from below. In fact, this probability is underapproximated by the probability that all the competing events will be triggered later than $a + 1$ and the event e will be triggered in the interval $(a, a + 1)$. These two random events are independent and both bounded from below by some b because all the events enabled in s have positive density on (a, ∞). Again, $\varepsilon = p \cdot b > 0$. $\square$

A.4 Proof of Proposition 3.11

Proposition 3.11. *There is $\delta > 0$ such that for every regional strategy $\sigma \in \Sigma$ and every $\pi \in \Pi$, a δ-separated p-history is reached almost surely from every p-history $\mathfrak{p}$.*

We prove that there are $n \in \mathbb{N}$, $\delta > 0$, and $\varepsilon > 0$ such that for every p-history $\mathfrak{p}$ and every $\pi \in \Pi$, the probability of reaching a δ-separated p-history in n steps is greater than ε. Then, we just iterate the argument.

For this proof we need to specify conditions under which the probability of triggering a fixed event in any p-history from a fixed set of p-histories is bounded from below. Since the behavior of the timed automaton is not important for these conditions, we formulate it for the game $\mathcal{G}$ and not for the product game $\mathcal{G}_{\mathcal{A}}$.

Definition A.4. *Let M be the maximal relevant bound increased by 1. Let $\mathfrak{h}$ be a history. We say that a cylinder template B is* fully feasible for $\mathfrak{h}$ *if for all runs $\rho \in Play(B)$ it holds in the first $|\mathfrak{h}| + |B|$ steps of the concatenated run $\mathfrak{h}\rho$ that no event is triggered before its lower bound and that the total waiting time for each event is never longer than its upper bound and also never longer than M.*

In other words, in a cylinder of a fully feasible template there are no runs that are by definition infeasible with respect to the previous history $\mathfrak{h}$. In addition, we restrict events with infinite upper bounds to be triggered in up to time M. Now, for fully feasible cylinder templates we can provide probability bound for triggering an event.

Lemma A.5. *For each interval length $\varepsilon > 0$ there exists a probability bound $b > 0$ such that for any pair of strategies σ and π and for any $x \in \mathbb{R}_{\geq 0}$ and any templates B and $B' = B(s, [x, x+\varepsilon], e)$ that are fully feasible for $\mathfrak{h}$ it holds that a conditional probability*

$$\mathcal{P}^{\sigma,\pi}_{\mathfrak{h}}(Play(B') \mid A) \quad \geq \quad b$$

where A is the set of runs that match the template B and the following state is s, i.e. $A = \bigcup_{e' \in \mathcal{E}} Play(B(s, (0, \infty), e'))$.

Proof. Let n be the length of $\mathfrak{h}$ and m the length of B. According to the definition of the probability measure $\mathcal{P}^{\sigma,\pi}_{\mathfrak{h}}$, the left side of the inequality equals to

$$\frac{\int_{I_{n+1}} State_{n+1} \cdot Win_{n+1} \cdot \quad \cdots \quad \cdot \int_{[x,x+\varepsilon]} State_{n+m+1} \cdot Win^{e}_{n+m+1} \quad dt_{n+m+1} \cdots dt_{n+1}}{\sum_{e' \in \mathcal{E}} \int_{I_{n+1}} State_{n+1} \cdot Win_{n+1} \cdot \quad \cdots \quad \cdot \int_{(0,\infty)} State_{n+m+1} \cdot Win^{e'}_{n+m+1} \quad dt_{n+m+1} \cdots dt_{n+1}}$$

$$= \frac{State_{n+1} \cdot \int_{I_{n+1}} Win_{n+1} \cdot \quad \cdots \quad \cdot State_{n+m+1} \cdot \int_{[x,x+\varepsilon]} Win^{e}_{n+m+1} \quad dt_{n+m+1} \cdots dt_{n+1}}{State_{n+1} \cdot \int_{I_{n+1}} Win_{n+1} \cdot \quad \cdots \quad \cdot State_{n+m+1} \cdot \sum_{e' \in \mathcal{E}} \int_{(0,\infty)} Win^{e'}_{n+m+1} \quad dt_{n+m+1} \cdots dt_{n+1}}$$

$$= \frac{State_{n+1} \cdot \int_{I_{n+1}} Win_{n+1} \cdot \quad \cdots \quad \cdot State_{n+m+1} \cdot \int_{[x,x+\varepsilon]} Win_{n+m+1} \quad dt_{n+m+1} \cdots dt_{n+1}}{State_{n+1} \cdot \int_{I_{n+1}} Win_{n+1} \cdot \quad \cdots \quad \cdot State_{n+m+1} \cdot \quad 1 \quad dt_{n+m} \cdots dt_{n+1}} = (*).$$

In what follows, we show that $\int_{[x,x+\varepsilon]} Win_{n+m+1} \; dt_{n+m+1}$ is bounded from bellow, i.e., that $\inf_{\vec{t} \in \vec{I}} \int_{[x,x+\varepsilon]} Win_{n+m+1} dt_{n+m+1} \geq b > 0$ where by $\vec{t} \in \vec{I}$ we abbreviate the waiting times $t_{n+1} \in I_{n+1}$, $\ldots, t_{n+m} \in I_{n+m}$. This leads to $(*) \geq b$. According to the definition of Win_{n+m+1}, we have

$$\inf_{\vec{t} \in \vec{I}} \int_{x}^{x+\varepsilon} Win_{n+m+1} \; dt_{n+m+1} = \inf_{\vec{t} \in \vec{I}} \int_{x}^{x+\varepsilon} f_{e|b(e,n+m+1)}(t) \quad \cdot \prod_{\substack{e' \in E(s) \\ e' \neq e}} \int_{t}^{\infty} f_{e'|b(e',n+m+1)}(y) \, dy \, dt$$
$$= (**).$$

Now we use the fact that B is fully feasible for $\mathfrak{h}$. All the runs that match B must satisfy after the m-th step the condition that all events e scheduled in s have at least $x + \varepsilon$ time left to their upper bound or the global bound M. Otherwise, B' would not be fully feasible for $\mathfrak{h}$. So we know that for all $\vec{t} \in \vec{I}$ and any $e \in E(s)$ the function $f_{e|b(e,n+m+1)}$ is positive on $(x, x + \varepsilon)$. From the definition of $f_{e|b(e,n+m+1)}$ one can see that for each $\vec{t} \in \vec{I}$ it is the density function of the event e, only shifted by some constant and scaled up by a constant factor. Let $m_e = \min\{u_e, M\}$. For a fixed $\vec{t} \in \vec{I}$ there is a scale factor $j \geq 1$ and a shift constant $k \geq 0$ such that $l_e \leq k + x \leq m_e - \varepsilon$ and for any waiting time $t \in [x, x+\varepsilon]$ we have $f_{e|b(e,n+m+1)}(t) = j \cdot f_e(t+k) \geq f_e(t+k)$. So we can see that for any $t \in [x, x+\varepsilon]$ it holds $\inf_{\vec{t} \in \vec{I}} f_{e|b(e,n+m+1)}(t) \geq \inf_{z \in [l_e, m_e]} f_e(z)$. This infimum may equal zero because f_e is only guaranteed to be positive on (l_e, u_e) and $f_e(l_e)$ or $f_e(m_e)$ may be zero. Therefore, we restrict the interval. The function f_e is positive on the closed interval $[l_e + \varepsilon/4, m_e - \varepsilon/4]$, therefore it has a minimal value $a_e > 0$ on this interval. For any $t \in [x + 1/4 \cdot \varepsilon, x + 3/4 \cdot \varepsilon]$ it holds that $\inf_{\vec{t} \in \vec{I}} f_{e|b(e,n+m+1)}(t) \geq \inf_{z \in [l_e + \varepsilon/4, m_e - \varepsilon/4]} f_e(z) = a_e > 0$. Finally we get

$$\begin{aligned}
(**) \quad &\geq \quad \inf_{\vec{t} \in \vec{I}} \int_{x+\frac{1}{4}\cdot\varepsilon}^{x+\frac{1}{2}\cdot\varepsilon} f_{e|b(e,n+m+1)}(t) \quad \cdot \prod_{\substack{e' \in E(s) \\ e' \neq e}} \int_{x+\frac{1}{2}\cdot\varepsilon}^{x+\frac{3}{4}\cdot\varepsilon} f_{e'|b(e',n+m+1)}(y)\, dt \\
&\geq \quad \inf_{\vec{t} \in \vec{I}} \int_{x+\frac{1}{4}\cdot\varepsilon}^{x+\frac{1}{2}\cdot\varepsilon} a_e\, dy \quad \cdot \prod_{\substack{e' \in E(s) \\ e' \neq e}} \int_{x+\frac{1}{2}\cdot\varepsilon}^{x+\frac{3}{4}\cdot\varepsilon} a_{e'}\, dy \;\; dt \\
&\geq \quad \frac{1}{4} \cdot \varepsilon \cdot a_e \cdot \prod_{\substack{e' \in E(s) \\ e' \neq e}} \frac{1}{4} \cdot \varepsilon \cdot a_{e'} = b > 0.
\end{aligned}$$

□

To finish the proof of Proposition 3.11 it is sufficient to prove the following lemma which is its one-shot version (note that iteration of this lemma proves the proposition).

Lemma A.6. *The probability of reaching a δ-separated p-history is globally bounded from below, i.e. there is $n \in \mathbb{N}$, $\delta > 0$, $b > 0$ such that for each p-history $\mathfrak{p}$ and pair of strategies σ and π, the probability of reaching a δ-separated p-history from $\mathfrak{p}$ in n steps is greater that b.*

Proof. We show that the fractional parts of p-clocks in a p-vector are controlled by randomness and that the players cannot avoid their eventual separation. The only way the players can interfere is by choosing the successor state (indirectly by choosing actions). For each possible decision of a player there is a probable enough response of the nature such that in n steps the nature leads the play in a δ-separated p-history.

The nature responses by choosing a waiting interval I and an event e. Hence, together with the players it forms in the game a cylinder template $(s_1, I_1, e_1)(s_2, I_2, e_2)\cdots$. Since we do not care for the time automaton part of the product game in this lemma, we use the probability space from the game. We first formalize when such a choice of the nature is probable enough. Let p_i be the probability of $Play(B_i)$ where B_i is the formed template $B_i = (s_1, I_1, e_1)\cdots(s_i, I_i, e_i)$. A player may then lead the game into several states with different probabilities. But there must be some preferred state s such that the set of runs in $Play(B_i)$ that continue into this state have at least average probability, i.e. at least $\frac{p_i}{|S|}$. So the nature puts $s_{i+1} = s$ and needs to choose an appropriate event e_{i+1} and interval I_{i+1} of fixed length such that B_{i+1} is a fully feasible template (see the definition above Lemma A.5). From Lemma A.5 we have that the probability $p_{i+1} > \frac{p_i}{|S|} \cdot d$ where d is some constant that depends only on the globally fixed interval length (the length of I_{i+1}).

For any i, such cylinder template B_i has a bounded probability $b = p_i > \frac{c^i}{|S|^i}$. We need to find a number $n \in \mathbb{N}$, an interval length $w > 0$, and an appropriate tactics of the nature such that for any decisions of the players

1. the template B_n ends in δ-separated p-history,
2. the template B_n is fully feasible and
3. all intervals in the template have length w.

First, we focus on the δ-separation (point 1). Observe that if a p-history is not δ-separated, there are (at least) two p-clocks having their fractional parts closer than δ. Moreover, if none of the clocks is reset during the n steps, their fractional parts stay closer than δ. On the other hand, if it takes enough time so that both p-clocks exceed their relevant bounds, they are then according to the definition δ-separated. Here, we use the global bound M. When M time units elapse, all p-clocks become irrelevant or must be reset at least once. We need such n and such a tactics that n steps take at least M time units. Again, let $m_e = \min\{u_e, M\}$. If we trigger every event e in the interval $[m_e - 1/2, m_e]$ then e can occur at most twice per time unit because $m_e \geq 1$. In total, there cannot be more than $2 \cdot |\mathcal{E}|$ event occurrences per time unit. As a result, $n = 2 \cdot |\mathcal{E}| \cdot M$ steps take at least M time units.

This does not guarantee δ-separation on its own. We need to utilize a simple combinatorial argument. For this, we observe how the fractional parts of p-clock values evolve in time. Imagine a $[0, 1]$ line segment, each p-clock has its point on this segment according to its fractional value. When the time flows, all the points are shifted to the right. When a point reaches 1, it then continues again from 0. We need that every event e is triggered in the interval $[m_e - 1/2, m_e]$. To achieve it, every time a new event is scheduled, we put a reservation on some point x in the interval $[0, 1/2]$. The reservation points are shifted by the time flow as well. Notice that the reservation for event e is always in the distance x from the event. We trigger the event e when the reservation reaches the point 1 and the event is in the distance $x \in [0, 1/2]$ from m_e. All the newly scheduled events and the clocks reset by the transition move to the reserved point (i.e. to 1 which identified with 0), the reservation is removed. If we always place the reservations δ-apart from all the other reservations and p-clocks, then the p-clocks moved to this reserved position after the event trigger are δ-separated from other p-clocks. At the beginning the p-clocks may not be δ-separated. Since every p-clock becomes irrelevant or is moved (at least once) to a δ-separated reserved position, we end up in a δ-separated p-history after n steps.

To assure the separated placement of reservations, we divide the $[0, 1]$ interval into $12 \cdot i$ segments of equal length where $i = |\mathcal{E} \cup \mathcal{X}|$. Since we put reservations only in the interval $[0, 1/2]$, we have $6 \cdot i$ segments available. When placing a new reservation, there are at most $(2i - 1)$ p-clocks and other reservations on this interval. Hence at most $(2i - 1)$ segments are occupied and somewhere there must be three free segments in a row. We put the reservation in the middle one. For δ equal to the length of one segment, this reservation is δ-separated from other points.

The previous explanation made a simplification, we planned for an event trigger when the reservation *point* exactly reaches 1. Instead, we need to reserve an interval of some fixed length w (see point 3) and allow event trigger anywhere in this intervals. We place the reservation interval in the middle of the selected segment. After an event trigger the p-clocks may be moved to any position in this reserved interval, i.e. may be at most $(w/2)$-apart from the middle of the segment. After n steps, the p-clocks may be $((w \cdot n)/2)$-apart from the middle of their segments. We still need them to be inside the segment, therefore we put $w = \varepsilon/n$.

So far we have only explained how we place the reservation for a newly scheduled events. Let (c, q, ξ) be the last p-control of the initial p-history $\mathfrak{p}$. We need to place reservations for all the events in the initial p-vector ξ. For the moment, let us assume that all the events are at least $(1/2)$-apart from their relevant bound. Let e be an event and $x = \mathit{frac}(\xi(e))$. We can place the reservation inside the image of $[x, x + 1/2]$ under the function *frac*. By the same arguments there must be three free segments next to each other somewhere in this area. Under this assumption our argumentation is complete. It is easy to see that our explanation defines a tactics for the nature to form the cylinder template B_n that satisfies the three points (notice that the formed template is fully feasible because the interval of length w up to the n steps never cross the lower or upper bounds of any event since they even do not cross the boundaries of segments).

Finally, we drop the assumption that in ξ each event is at least $(1/2)$-apart from its relevant bound. Let $V \subseteq \mathcal{E}$ be the set of events e such that $\xi(e) > m_e - 1$. If all events $e \in V$ are triggered during a half of a time unit, we reach for any decisions of the players a time vector where all events are at least $(1/2)$-apart from their relevant bound and move on to the previous case satisfying this assumption. It is evident that the probability p that all events in V fire in up to the half of a time unit is bounded from below. All events occur independently, i.e. $p = \prod_{e \in V} p_e$ where p_e is the probability for the event e to occur in up to the half of a time unit. For an event e in the interval $[m_e - 1/2, m_e]$ the probability p_e is 1. For an event e in the interval $[m_e - 1, m_e - 1/2]$ the probability $p_e \geq \int_{l_e+1/4}^{u_e-1/2} f_e(x)dx \geq 1/4 \cdot \inf\{f_e(x) \mid x \in [l_e + 1/4, u_e - 1/2]\} > 0$.

□

B Algorithm - appendix proofs

B.1 Proof of Proposition 3.12

Formally, the game $\mathcal{G}^{\mathcal{A}}$ is a tuple $(V, (V_\Box, V_\Diamond, V_\bigcirc), \rightsquigarrow)$ where

- $V = \mathcal{R} \cup (\mathcal{R} \times Act)$ is the set of vertices partitioned into the set $V_\Box = \{(c, q, \Xi) \in \mathcal{R} \mid c \in C_\Box\}$ of vertices of player $\Box$, the set $V_\Diamond = \{(c, q, \Xi) \in \mathcal{R} \mid c \in C_\Diamond\}$ of vertices of player $\Diamond$, and the set $V_\bigcirc = \mathcal{R} \times Act$ of vertices of the stochastic player.
- $\rightsquigarrow$ is the transition relation, where for every action a and regions $r = (c, q, \Xi)$ and $r' = (c', q', \Xi')$ we set
  - $r \rightsquigarrow (r, a)$ if action a is enabled in region r, i.e. $a \in A(c)$
  - $(r, a) \rightsquigarrow r'$ if the probability of moving from an arbitrary $\mathfrak{p} \in r$ to r' with action a is positive. This is a correct definition due to Lemma 3.3.

Proposition 3.12. *Let $\mathcal{G}$ be a game and $\mathcal{A}$ be a deterministic timed automaton. For every p-history $\mathfrak{p}$ in a region r, a positional strategy is almost-sure winning for r in $\mathcal{G}^{\mathcal{A}}$ iff it is almost-sure winning for $\mathfrak{p}$ in $\mathcal{G}_{\mathcal{A}}$,*

Proof. We start with the "only if" part and then proceed with the "if" part:

- From Proposition 3.10 and Proposition 3.11 we have that every regional strategy that is candidate for $\mathfrak{p}$ is almost-sure winning for $\mathfrak{p}$. We show that every positional strategy σ that is almost-sure winning for r in $\mathcal{G}^{\mathcal{A}}$ is a candidate for the p-history $\mathfrak{p}$. Recall that σ is candidate for $\mathfrak{p}$ if for any $\mathfrak{p}' \in \mathcal{S}(\sigma, \mathfrak{p})$ it reaches the target positively for any $\pi \in \Pi$. Let $\mathcal{S}$ be the set of regions reachable from r in the discrete game $\mathcal{G}^{\mathcal{A}}$ for some π. We show that $\mathcal{S}(\sigma, \mathfrak{p}) \subseteq \mathcal{S}$ and that for any $\mathfrak{p}' \in \mathcal{S}$ the strategy σ reaches in $\mathcal{G}^{\mathcal{A}}$ the target positively for any $\pi \in \Pi$.

Firstly, we show that $\mathcal{S}(\sigma, \mathfrak{p}) \subseteq \mathcal{S}$. Since we do not positively escape from $\mathcal{S}$ in $\mathcal{G}^{\mathcal{A}}$, there is no finite sequence of regions leading us away from $\mathcal{S}$ in $\mathcal{G}^{\mathcal{A}}$ using σ. Therefore, there is also no sequence of regions leading us away from $\mathcal{S}$ in $\mathcal{G}_{\mathcal{A}}$ such that the sequence has positive probability for σ. This follows by iterating Lemma B.1 which claims that the construction respects one-step qualitative reachability:

Lemma B.1. *Let r and t be regions and $\mathfrak{p} \in r$ a p-history. For every action a available in r and every strategy σ_a and π_a choosing a in r, we have that*

$$\mathcal{P}_{\mathfrak{p}}^{\sigma_a,\pi_a}(Play(t)) > 0 \quad \textit{iff} \quad r \rightsquigarrow (r,a) \rightsquigarrow t$$

Proof. The lemma directly follows from Lemma 3.3 and the construction of $\mathcal{G}^{\mathcal{A}}$. □

Secondly, we show that for any $\mathfrak{p}' \in \mathcal{S}$ the strategy σ reaches in $\mathcal{G}^{\mathcal{A}}$ the target positively for any $\pi \in \Pi$. Let r' be the region such that $\mathfrak{p}' \in r'$. Recall that $\mathcal{G}^{\mathcal{A}}$ is a discrete stochastic game with finite state space. Since σ is winning, it guarantees reaching the target T from r' with positive probability in at most $2 \cdot |\mathcal{R}|$ steps for any strategy π. Thus for each stochastic vertex positively reachable from r (for some π), we can color some of its outgoing transitions in such a way, that using only these colored transitions we always end up in T within $2 \cdot |\mathcal{R}|$ steps no matter what transitions have been taken in $V_\Diamond$. This coloring thus produces a set $\mathcal{B}$ of sequences of regions which are of length $\leq 2 \cdot |\mathcal{R}|$, end in T, and for every strategy π in $\mathcal{G}_{\mathcal{A}}$ and $\mathfrak{p}' \in r'$ we have $\mathcal{P}_{\mathfrak{p}'}^{\sigma,\pi}(\bigcup_{B \in \mathcal{B}} Play(B)) > 0$ by Lemma B.1. Hence for every $\mathfrak{p}' \in r'$ and $\pi \in \Pi$ also $\mathcal{P}_{\mathfrak{p}'}^{\sigma,\pi}(Reach(\mathcal{R}_T)) > 0$ in $\mathcal{G}_{\mathcal{A}}$.

- We show the "if" part by contraposition. Assume that a positional strategy σ is not almost-sure winning for r. We show that there is π such that there is a positively reachable region *bad* such that from *bad* the probability to reach T is zero. Hence, the regional strategy σ is not almost-sure winning for $\mathfrak{p}$.

  We know σ is not almost-sure winning for r in $\mathcal{G}^{\mathcal{A}}$. Therefore, there is a strategy π such that it reaches a region from where it prevents reaching the target at all. We fix the choices of σ and π in the game $\mathcal{G}^{\mathcal{A}}$ and proceed analogously as in the preceding case. For each stochastic vertex, we color an outgoing transition such that when using the colored transitions we always reach a state from where T is not reachable at all. This yields a sequence of regions B of length $\leq 2 \cdot |\mathcal{R}|$ ending in a bad region with $\mathcal{P}_{\mathfrak{p}}^{\sigma,\pi}(Play(B)) > 0$ by Lemma B.1. Thus we positively reach a bad region where the probability to reach the target is zero, hence $\inf_{\pi \in \Pi} \mathcal{P}_{\mathfrak{p}}^{\sigma,\pi}(Reach(\mathcal{R}_T)) < 1$.

□

B.2 Algorithm for computing the game graph of $\mathcal{G}^{\mathcal{A}}$

Now we give the algorithm for computing $\mathcal{G}^{\mathcal{A}}$ for a given game $\mathcal{G}$ and DTA $\mathcal{A}$. First, we need a symbolic representation for time areas. Let $\mathcal{E} \cup \mathcal{X} = a_1, \ldots, a_n$. Recall that the relevant bound of an p-clock a_i is the maximal value of a_i that matters for the region relation. We denote the relevant bound of a_i as b_i. Let Ξ be a time area. The set of all p-vectors $\xi \in \Xi$ may be described by a structure $(\Delta, \preceq, c, d)$ where

- $\Delta \subseteq \mathcal{E} \cup \mathcal{X}$ is the set of relevant p-clocks
- $\preceq$ is a pre-order of fractional parts on the set Δ, $a_i \preceq a_j$ if $frac(a_i) \leq frac(a_j)$
- c and d assign to each p-clock a_i from Δ a pair of integers $c(i)$ and $d(i)$ such that $0 \leq c(i) \leq d(i) \leq b_i$ and $c(i) = d(i)$ or $c(i) + 1 = d(i)$.

It is a straightforward observation that the set of all possible time area structures is finite and that there is a bijection between time areas and time area structures. Now we need to define two type of operations on time area structures, the reset operation and the time successor operation and one type of predicate, satisfaction of guards. Let $A = (\Delta, \preceq, c, d)$ be a time area structure. We say that A is *thin* if $c(i) = d(i)$ for some relevant p-clock a_i. Otherwise we say that A is *thick*.

Reset Let $X \subseteq (\mathcal{E} \cup \mathcal{X})$ be a reset set. We denote the X-reset of A as $A[X := \vec{0}]$. We define $A[X := \vec{0}] = (\Delta', \preceq', c', d')$ such that intuitively all p-clocks a_i in X become relevant in Δ', are placed on the beginning in the pre-order $\preceq'$ and are assigned 0 in c' and d'. Precisely,

- $\Delta' = \Delta \cup X$
- for $a, b \in \Delta \setminus X$, $a \preceq' b$ iff $a \preceq b$, for $a \in X$ we put
  - $a \preceq' b$ for all $b \in \Delta'$ and
  - $b \preceq' a$ for all $b \in X$ and for all $b \in \Delta$ such that $b \subseteq 0$,
- for $a \in \Delta \setminus X$, $c'(a) = c(a)$ and $d'(a) = d(a)$ and for $a \in X$, $c(a) = d(a) = 0$.

Time successor If $A = (\Delta, \preceq, c, d)$ is such that Δ is empty or that some event $a_i \in \Delta$ has reached its relevant bound ($c(i) = b_i$) the time successor of A is undefined. Otherwise we define the time successor of A, denoted by $succ(A)$ as a time area structure $(\Delta', \preceq', c', d')$ such that

- if A is thin, let $X \subseteq \Delta$ be the set of integral p-clocks that equal their relevant bound, and $Y \subseteq \Delta$ be the set of other integral p-clocks. Then $\Delta' = \Delta \setminus X$, $\preceq'$, c' and d' equal $\preceq$, c and d restricted to Δ' with the only exception of p-clocks a from Y for which $d'(a) = d(a) + 1$.

- if A is thick, let M_A be the set of maximal p-clocks w.r.t. $\preceq$, we put $\varDelta' = \varDelta$ and $d' = d$, all p-clocks a from M_A are placed on the beginning of $\preceq'$ and $c'(a) = c(a) + 1$, for other p-clocks $\preceq' = \preceq$ and $c' = c$.

Guard satisfaction We write that $A \models g$ if

- $g = x \leq b$ and $d(x) \leq b$ or if $g = x < b$ and $d(x) \leq b$ and $c(x) < b$
- $g = x \geq b$ and $c(x) \geq b$ or if $g = x > b$ and $c(x) \geq b$ and $d(x) > b$
- $g = g_1 \wedge g_2$ and $A \models g_1$ and $A \models g_2$

In Algorithm 2 we describe the procedure for constructing the region graph $G^{\mathcal{A}}$.

Theorem B.2. *Algorithm 2 terminates and constructs the game $\mathcal{G}^{\mathcal{A}}$.*

Proof. The algorithm terminates if for any time area structure A there is a n such that $succ^n(A)$ is undefined. Obviously *succ* of a thin structure is a thick structure and vice versa. In each iteration of *succ* on a thin structure $d(e)$ is incremented for at least one p-clock. Each p-clock has a finite relevant bound, so p-clocks are in sequence removed from the set $\varDelta$ until $\varDelta$ is empty or an event reaches its relevant bound. In both cases *succ* of such a time area is undefined.

Clearly, the set of vertices is constructed according to the definition of $G^{\mathcal{A}}$, all edges from non-stochastic to stochastic vertices are also added to $\rightsquigarrow$ according to the definition. We need to show that upon termination for any stochastic vertex (r, a) and non-stochastic vertex r' it holds that $r \rightsquigarrow r'$ if and only if for any $\mathfrak{p} \in r$ holds that $P_{\mathfrak{p}}^{\sigma_a,\pi_a}(Runs(r')) > 0$ where σ_a and π_a are arbitrary strategies that choose action a where possible. The argumentation closely copies the proof of Lemma 3.3. $\square$

Algorithm 2 Construct the game $G^{\mathcal{A}}$

input: events $\mathcal{E}$, game $\mathcal{G} = (S, E, C_\square, C_\diamond, Act, F, A, \mu_0)$, and DTA $\mathcal{A} = (Q, \Sigma, \mathcal{X}, \longrightarrow, q_0, T)$

output: game $\mathcal{G}^{\mathcal{A}}$

construct the set of all area structures AS

$V_\square \leftarrow C_\square \times Q \times AS$

$V_\diamond \leftarrow C_\diamond \times Q \times AS$

$V_\bigcirc \leftarrow (C_\square \cup C_\diamond) \times Q \times AS \times Act$

$V \leftarrow V_\square \cup V_\diamond \cup V_\bigcirc$

$\rightsquigarrow \leftarrow \emptyset$

for all non-stochastic vertices $(c, q, A) \in (V_\square \cup V_\diamond)$ **do**

  for all $a \in Act$ **do**

    if $a \in A(c)$ **then**

      add to $\rightsquigarrow$ a pair $((c, q, A), (c, q, A, a))$

for all stochastic vertices $(c, q, A, a) \in V_\bigcirc$ **do**

  for all states s in support of a **do**

    $B \leftarrow A[X := \vec{0}]$ where X is the set of p-clocks relevant in A that are not scheduled in s

    while B has a time successor **do**

      $B \leftarrow succ(B)$

      if B is thick **then**

        for all events e relevant in B **do**

          for all edges $(q, L(s, e), g, X, q')$ in the automaton $\mathcal{A}$ **do**

            if $B \models g$ **then**

              $B' \leftarrow B[X \cup \{e\} := \vec{0}]$

              $c' \leftarrow F(s, e)$

              add to $\rightsquigarrow$ a pair $((c, q, A, a), (c', q', B')$.

  $Prob((c, q, A, a)) \leftarrow$ uniform distribution

return $(V, (V_\square, V_\diamond, V_\bigcirc), \rightsquigarrow)$