



Jovanovic, A., Kwiatkowska, M., Norman, G., and Peyras, Q. (2017)
Symbolic optimal expected time reachability computation and controller
synthesis for probabilistic timed automata. *Theoretical Computer Science*,
669, pp. 1-21. (doi:[10.1016/j.tcs.2017.01.015](https://doi.org/10.1016/j.tcs.2017.01.015))

This is the author's final accepted version.

There may be differences between this version and the published version.
You are advised to consult the publisher's version if you wish to cite from
it.

<http://eprints.gla.ac.uk/134444/>

Deposited on: 16 January 2017

Enlighten – Research publications by members of the University of Glasgow
<http://eprints.gla.ac.uk/33640>

Symbolic Optimal Expected Time Reachability Computation and Controller Synthesis for Probabilistic Timed Automata

Aleksandra Jovanović^a, Marta Kwiatkowska^a, Gethin Norman^b,
Quentin Peyras^c

^a*Department of Computer Science, University of Oxford, Oxford, UK*

^b*School of Computing Science, University of Glasgow, Glasgow, UK*

^c*Département Informatique de l'ENS Cachan, Université Paris-Saclay, France*

Abstract

In this paper we consider the problem of computing the optimal (minimum or maximum) expected time to reach a target and the synthesis of an optimal controller for a probabilistic timed automaton (PTA). Although this problem admits solutions that employ the digital clocks abstraction or statistical model checking, symbolic methods based on zones and priced zones fail due to the difficulty of incorporating probabilistic branching in the context of dense time. We work in a generalisation of the setting introduced by Asarin and Maler for the corresponding problem for timed automata, where *simple* and *nice* functions are introduced to ensure finiteness of the dense-time representation. We find restrictions sufficient for value iteration to converge to the optimal expected time on the uncountable Markov decision process representing the semantics of a PTA. We formulate Bellman operators on the backwards zone graph of a PTA and prove that value iteration using these operators equals that computed over the PTA's semantics. This enables us to extract an ε -optimal controller from value iteration in the standard way.

Key words: Probabilistic Timed Automata, Controller Synthesis,
Probabilistic Verification, Symbolic Model Checking

1. Introduction

Systems which exhibit real-time, probabilistic and nondeterministic behaviour are widespread and ubiquitous in many areas such as medicine, telecommunications, robotics and transport. Timing constraints are often vital to the correctness of embedded devices and stochasticity is needed due to unreliable channels, randomisations and component failure. Finally, nondeterminism is an important concept which allows us to model and analyse systems operating in a distributed environment and/or exhibiting concurrency. A natural model for such systems, *probabilistic timed automata* (PTAs), a probabilistic extension

of timed automata (TAs) [1], was proposed in [2, 3, 4]. They are finite-state automata equipped with real-valued clocks which measure the passage of time and whose transitions are probabilistic. More specifically, transitions are expressed as discrete probability distributions over the set of edges, each such edge specifying a successor location and a set of clocks to reset.

An important class of properties on PTAs are *probabilistic reachability* properties. They allow us to check statements such as: “with probability 0.05 or less the system aborts” or “the data packet will be delivered within 1 second with minimum 0.95 probability”. Model checking algorithms for these properties are well studied. Forwards reachability [3] yields only approximate probability values (upper bounds on maximum reachability probabilities). An abstraction refinement method, based on stochastic games, has subsequently been proposed in [5] for the computation of exact values and implemented in PRISM [6]. An alternative method is backward reachability [7], also giving exact values. These are all symbolic algorithms based on *zones*, a structure that represents in a concise way sets of the automaton states with equivalent behaviour.

Another important class of properties, which is the focus of this paper, is *expected reachability*. They can express statements such as “the expected number of packets sent before failure is at least 100” or “the expected time until a message is delivered is at most 20ms”. These properties turned out to be more difficult to verify on PTAs and currently no symbolic approach exists. Even for TAs, the research first concentrated on checking whether there exist system behaviours that satisfy a certain property (for example, reaching the target set of states). In many situations this is not sufficient, as we often want to distinguish between behaviours that reach target states in 10 or 1,000 seconds. In [8], a backward fixed-point algorithm was proposed for controller synthesis for TAs, which generates a controller that reaches the target in minimum time. The analogous problem for priced timed automata, a model comprising more general reward (or cost) structures, was also considered. The minimum reward reachability for this model has been solved using the region graph method [9], and later extended for more efficient *priced zones* [10] and implemented in UPPAAL [11].

Contributions. We propose the first zone-based algorithms to compute the optimal expected time to reach a target set in a PTA and synthesise an ε -optimal controller. The semantics of a PTA is an uncountable Markov decision process (MDP). Under suitable restrictions, we are able to prove that value iteration converges to the optimal expected time on this MDP. We formulate Bellman operators on the backwards zone graph of a PTA and show that value iteration using these operators yields the same values as those computed on the MDP. This enables us to extract an ε -optimal controller from value iteration in the standard way. This problem has been open for several years, with previous symbolic zone-based methods, including priced zones, being unsuitable for computing expected values since accumulated rewards are *unbounded*. In order to represent the value functions we introduce rational simple and rational nice functions, a generalisation of Asarin and Maler’s classes of simple and nice functions [8].

Related work. Expected reachability properties of PTAs can be verified using the *digital clocks* method [12], which assumes an integral model of time as opposed to a dense model of time. Although this method suffers from state-space explosion, it has been shown useful in practice for the analysis of a number of real-world protocols, see for example [12, 13]. In addition, this approach has recently been extended to allow the analysis of partially observable PTAs against expected reachability properties [14]. In [15], the minimum expected reward for priced timed games has been solved using *statistical model checking* and UPPAAL-SMC [16]. This is orthogonal to numerical model checking, and is based on simulation and hypothesis testing, thus giving only approximate results which are not guaranteed to be correct.

In [17] the authors consider priced probabilistic timed automata and study reward-bounded probabilistic reachability, which determines whether the maximal probability to reach a set of target locations, within given bounds on the accumulated reward and elapsed time, exceeds a threshold. Although this problem is shown to be undecidable [18], a semi-decidable backwards algorithm using priced zones, which terminates if the problem is affirmative, is implemented in FORTUNA [19].

Outline. In Section 2 we define MDPs and give existing results concerning optimal reward computation for uncountable MDPs. Section 3 defines PTAs and introduces the assumptions needed for the adoption of the results of Section 2. In Section 4, we present our algorithms for computing optimal expected time reachability and synthesis of an ε -optimal controller using the backwards zone graph of a PTA. Section 4 also introduces a representation of the value functions that generalise the simple and nice functions of [8] and gives an example demonstrating the approach. We conclude with Section 5.

A preliminary conference version of this paper was published as [20], where only minimum expected time was considered.

2. Background

Let \mathbb{R} be the set of reals, \mathbb{R}_+ the set of non-negative reals, \mathbb{N} the natural numbers (including 0), \mathbb{Q} the rationals and \mathbb{Q}_+ the non-negative rationals. A discrete probability distribution over a (possibly uncountable) set S is a function $\mu : S \rightarrow [0, 1]$ such that $\sum_{s \in S} \mu(s) = 1$ and the set $\{s \in S \mid \mu(s) > 0\}$ is finite. We denote by $\text{Dist}(S)$ the set of distributions over S . A distribution $\mu \in \text{Dist}(S)$ is a point distribution if there exists $s \in S$ such that $\mu(s) = 1$.

Markov Decision Processes (MDPs) is a widely used formalism for modelling systems which exhibit both nondeterministic and probabilistic behaviour.

Definition 1. An MDP is a tuple $\mathcal{M} = (S, s_0, A, P_{\mathcal{M}}, R_{\mathcal{M}})$, where:

- S is a (possibly uncountable) set of states;
- $s_0 \in S$ is an initial state;

- A is a (possibly uncountable) set of actions;
- $P_{\mathcal{M}} : (S \times A) \rightarrow \text{Dist}(S)$ is a (partial) probabilistic transition function;
- $R_{\mathcal{M}} : (S \times A) \rightarrow \mathbb{R}$ is a reward function.

A state s of an MDP \mathcal{M} has a set of enabled actions, denoted by $A(s)$, given by the set of actions for which $P_{\mathcal{M}}(s, \cdot)$ is defined. A transition in \mathcal{M} from state s is first made by nondeterministically selecting an available action $a \in A(s)$. After the choice is made, a successor state s' is selected randomly according to the probability distribution $P_{\mathcal{M}}(s, a)$, i.e. the probability that a transition to s' occurs is equal to $P_{\mathcal{M}}(s, a)(s')$, and the reward $R_{\mathcal{M}}(s, a)$ is accumulated when making this transition.

An infinite *path* of an MDP \mathcal{M} is a sequence $\omega = s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} \dots$ of transitions such that $P_{\mathcal{M}}(s_i, a_i)(s_{i+1}) > 0$ for all $i \in \mathbb{N}$, and it represents a particular resolution of both nondeterminism and probability. A finite path is a prefix of an infinite path ending in a state. The $(i+1)$ th state of a path ω is denoted by $\omega(i)$ and the action associated with the $(i+1)$ th transition by $\omega[i]$. We denote the set of all infinite (finite) paths of \mathcal{M} by $IPaths_{\mathcal{M}}$ ($FPaths_{\mathcal{M}}$) and the last state of a finite path ω by $last(\omega)$.

A *strategy* (also called an *adversary* or *policy*) of an MDP \mathcal{M} resolves the choice between available actions in each state, based on the execution so far.

Definition 2. A strategy of an MDP \mathcal{M} is a function $\sigma : FPaths_{\mathcal{M}} \rightarrow \text{Dist}(A)$ such that $\sigma(\omega)(a) > 0$ only if $a \in A(last(\omega))$. The set of strategies of \mathcal{M} is denoted by $\Sigma_{\mathcal{M}}$.

For a fixed strategy σ and state s of an MDP \mathcal{M} , we can define a probability measure \mathcal{P}_s^σ over the set of infinite paths starting in s [21]. A strategy σ is memoryless if its choices only depend on the current state, and deterministic if $\sigma(\omega)$ is a point distribution for all $\omega \in FPaths_{\mathcal{M}}$.

Two fundamental quantitative properties of an MDP are the probability of reaching a set of target states and the expected reward accumulated before reaching a target. For a strategy σ , state s and set of target states F of an MDP \mathcal{M} , the probability of reaching F and expected reward accumulated before reaching F from s under σ are given by:

$$\begin{aligned} \mathbb{P}_{\mathcal{M}}^\sigma(s, F) &\stackrel{\text{def}}{=} \mathcal{P}_s^\sigma \{ \omega \in IPaths_{\mathcal{M}} \mid \omega(i) \in F \text{ for some } i \in \mathbb{N} \} \\ \mathbb{E}_{\mathcal{M}}^\sigma(s, F) &\stackrel{\text{def}}{=} \int_{\omega \in IPaths_{\mathcal{M}}} \text{rew}(\omega, F) d\mathcal{P}_s^\sigma \end{aligned}$$

where for any infinite path ω :

$$\text{rew}(\omega, F) \stackrel{\text{def}}{=} \sum_{i=0}^{k_F} R_{\mathcal{M}}(\omega(i), \omega[i])$$

where $k_F = \min\{k-1 \mid \omega(k) \in F\}$ if there exists $k \in \mathbb{N}$ such that $\omega(k) \in F$ and $k_F = \infty$ otherwise. Note that, this definition of expected reachability deviates from the standard definition for MDPs, see e.g. [22], where the result is set to infinity in states for which the probability of reaching the target is less than 1.

However, for expected time properties of (time-divergent) PTAs – the focus of this paper – the two definitions are equivalent.

The standard approach is to analyse the optimal values of these properties, i.e. the minimum and maximum values over all strategies:

$$\begin{aligned}\mathbb{P}_{\mathcal{M}}^{\min}(s, F) &\stackrel{\text{def}}{=} \inf_{\sigma \in \Sigma_{\mathcal{M}}} \mathbb{P}_{\mathcal{M}}^{\sigma}(s, F) \\ \mathbb{P}_{\mathcal{M}}^{\max}(s, F) &\stackrel{\text{def}}{=} \sup_{\sigma \in \Sigma_{\mathcal{M}}} \mathbb{P}_{\mathcal{M}}^{\sigma}(s, F) \\ \mathbb{E}_{\mathcal{M}}^{\min}(s, F) &\stackrel{\text{def}}{=} \inf_{\sigma \in \Sigma_{\mathcal{M}}} \mathbb{E}_{\mathcal{M}}^{\sigma}(s, F) \\ \mathbb{E}_{\mathcal{M}}^{\max}(s, F) &\stackrel{\text{def}}{=} \sup_{\sigma \in \Sigma_{\mathcal{M}}} \mathbb{E}_{\mathcal{M}}^{\sigma}(s, F).\end{aligned}$$

The optimal values can be computed using *Bellman operators* [23]. More precisely, under certain conditions on the MDP and target set under study, using a Bellman operator an optimal value can be obtained through a number of techniques, including *value iteration* and *policy iteration*, see for example [24, 25]. Concerning optimal expected reachability we have the following definition.

Definition 3. *Let \mathcal{M} be an MDP with state space S and F a set of target states of \mathcal{M} . The Bellman operators $T_{\mathcal{M}}^{\min}, T_{\mathcal{M}}^{\max} : (S \rightarrow \mathbb{R}) \rightarrow (S \rightarrow \mathbb{R})$ for optimal expected reachability are defined as follows. For any function $f : S \rightarrow \mathbb{R}$ and state $s \in S$:*

$$\begin{aligned}T_{\mathcal{M}}^{\min}(f)(s) &= \begin{cases} 0 & \text{if } s \in F \\ \inf_{a \in A(s)} \{R_{\mathcal{M}}(s, a) + \sum_{s' \in S} P_{\mathcal{M}}(s, a)(s') \cdot f(s')\} & \text{otherwise} \end{cases} \\ T_{\mathcal{M}}^{\max}(f)(s) &= \begin{cases} 0 & \text{if } s \in F \\ \sup_{a \in A(s)} \{R_{\mathcal{M}}(s, a) + \sum_{s' \in S} P_{\mathcal{M}}(s, a)(s') \cdot f(s')\} & \text{otherwise.} \end{cases}\end{aligned}$$

Value iteration for these operators corresponds to repeatedly applying an operator when starting from some initial approximation f_0 until some convergence criterion is met, e.g. computing $(T_{\mathcal{M}}^{\min})^{n+1}(f_0) = T_{\mathcal{M}}^{\min}((T_{\mathcal{M}}^{\min})^n(f_0))$ until $\|(T_{\mathcal{M}}^{\min})^{n+1}(f_0) - (T_{\mathcal{M}}^{\min})^n(f_0)\| \leq \varepsilon$ for some threshold ε . Value iteration is simple to implement, with low memory requirements; however, convergence is not guaranteed in all cases, see [26] for an extension guaranteeing convergence. On the other hand, policy iteration starts with an arbitrary, deterministic and memoryless strategy, and then tries repeatedly to construct an improved (deterministic and memoryless) strategy. This is achieved by computing the expected reachability values for the current strategy and, if possible, updating the actions choices so that the expected reachability values decrease.

We now adapt the results of [27] for optimal total expected rewards for possibly uncountable-state and uncountable-action set MDPs. The conditions imposed by [27] correspond, in our setting, to those given below (since the MDPs constructed from PTAs only contain discrete distributions and non-negative reward values, the assumptions we require are weaker) except we have strengthened Assumption 4(b) over that given in [27] to allow the same assumptions to be used for both minimum and maximum expected reachability. This strengthening is just to simplify the presentation and imposes no further restrictions on the class of PTAs we can analyse.

Assumption 4. For any MDP $\mathcal{M}=(S, s_0, A, P_{\mathcal{M}}, R_{\mathcal{M}})$ and target set $F \subseteq S$:

- (a) $A(s)$ is compact for all $s \in S$;
- (b) $R_{\mathcal{M}}$ is bounded and $a \mapsto R_{\mathcal{M}}(s, a)$ is continuous for all $s \in S$;
- (c) if σ is a memoryless, deterministic strategy which is not proper, then $\mathbb{E}_{\mathcal{M}}^{\sigma}(s, F)$ is unbounded for some $s \in S$;
- (d) there exists a proper, memoryless, deterministic strategy;

where a strategy σ is called proper if $\mathbb{P}_{\mathcal{M}}^{\sigma}(s, F)=1$ for all $s \in S$.

The relevance of Assumption 4 for PTAs will be discussed after introducing the semantics of a PTA, which is an infinite-state MDP. In particular, in Assumption 9 we will give the requirements on a PTA that ensure that its semantics meets Assumption 4. Using Assumption 4 we have the following result for minimum expected reachability.

Theorem 5 ([27]). *If \mathcal{M} and F are an MDP and the corresponding target set for which Assumption 4 holds and the minimum expected reward values are bounded below, then:*

- there exists a memoryless, deterministic strategy that achieves the minimum expected reward of reaching F ;
- the minimum expected reward values are the unique solutions to $T_{\mathcal{M}}^{\min}$;
- value iteration over $T_{\mathcal{M}}^{\min}$ converges to the minimum expected reward values when starting from any bounded function;
- policy iteration converges to the minimum expected reward values when starting from any proper, memoryless, deterministic strategy.

In the case of maximum expected reachability we can adapt the above theorem by negating all the reward values in the MDP under study. This leads to the following corollary.

Corollary 6. *If \mathcal{M} and F are an MDP and the corresponding target set for which Assumption 4 holds and the maximum expected reward values are bounded above, then:*

- there exists a memoryless, deterministic strategy that achieves the maximum expected reward of reaching F ;
- the maximum expected reward values are the unique solutions to $T_{\mathcal{M}}^{\max}$;
- value iteration over $T_{\mathcal{M}}^{\max}$ converges to the maximum expected reward values when starting from any bounded function;
- policy iteration converges to the maximum expected reward values when starting from any proper, memoryless, deterministic strategy.

3. Probabilistic Timed Automata

We now introduce PTAs, a modelling framework for systems which incorporate probabilistic, nondeterministic and real-time behaviour.

3.1. Clocks, Clock Valuations and Zones

Let \mathcal{X} be a set of real-valued variables called clocks, which increase at the same, constant rate. A function $v : \mathcal{X} \rightarrow \mathbb{R}_+$ is called clock valuation and the set of all clock valuations is denoted by $\mathbb{R}_+^{\mathcal{X}}$. Let $\mathbf{0}$ be the clock valuation that assigns 0 to all clocks in \mathcal{X} . For any $R \subseteq \mathcal{X}$ and clock valuation v , we write $v[R]$ for the clock valuation such that, for any $x \in \mathcal{X}$, we have $v[R](x)=0$ if $x \in R$ and $v[R](x)=v(x)$ otherwise. For $t \in \mathbb{R}_+$, $v+t$ denotes the clock valuation such that $(v+t)(x)=v(x)+t$ for all $x \in \mathcal{X}$. A zone over \mathcal{X} is an expression of the form:

$$\zeta ::= \mathbf{true} \mid x \leq d \mid c \leq x \mid x+c \leq y+d \mid \neg\zeta \mid \zeta \wedge \zeta$$

where $x, y \in \mathcal{X}$ and $c, d \in \mathbb{N}$. The set of zones over \mathcal{X} is denoted $Zones(\mathcal{X})$. A clock valuation v satisfies a zone ζ , denoted $v \models \zeta$, if ζ resolves to true after substituting each occurrence of a clock x with $v(x)$. The semantics of a zone ζ is given by the set of clock valuations which satisfy it.

We require a number of classical operations on zones [28, 29]. For any zone ζ , the zone $\nearrow\zeta$ represents the set of valuations reachable from a valuation in ζ by letting time pass. Conversely, $\swarrow\zeta$ represents the valuations that can reach ζ by letting time pass. Furthermore, for a set of clocks R , $\zeta[R]$ represents the valuations obtained from those in ζ by resetting the clocks R and $[R]\zeta$ the valuations which result in a valuation in ζ when the clocks in R are reset.

3.2. Syntax and Semantics of PTAs

We now present the formal syntax and semantics of PTAs.

Definition 7. A PTA \mathcal{P} is a tuple $(L, l_0, \mathcal{X}, Act, \text{enab}, \text{prob}, \text{inv})$ where:

- L is a finite set of locations;
- $l_0 \in L$ is an initial location;
- \mathcal{X} is a finite set of clocks;
- Act is a finite set of actions;
- $\text{enab} : (L \times Act) \rightarrow Zones(\mathcal{X})$ is an enabling condition;
- $\text{prob} : (L \times Act) \rightarrow \text{Dist}(2^{\mathcal{X}} \times L)$ is a probabilistic transition function;
- $\text{inv} : L \rightarrow Zones(\mathcal{X})$ is an invariant condition.

A state of PTA \mathcal{P} is a pair $(l, v) \in L \times \mathbb{R}_+^{\mathcal{X}}$ such that the clock valuation v satisfies the invariant $\text{inv}(l)$. A transition is a time-action pair (t, a) corresponding to letting time t elapse and then performing the action a . In a state (l, v) , time can elapse as long as the invariant $\text{inv}(l)$ remains continuously satisfied and action a can be performed only if the enabling condition $\text{enab}(l, a)$ is then satisfied. If the transition (t, a) is performed in the state (l, v) , then the set of clocks to reset and successor location are selected randomly according to the probability distribution $\text{prob}(l, a)$.

The semantics of a PTA \mathcal{P} is an infinite-state MDP and, in the definition given below, the values of the reward function of this MDP correspond to the elapsed time. Alternative reward values can be defined, for example, based on linearly-priced timed automata [30].

Definition 8. For a PTA $\mathcal{P}=(L, l_0, \mathcal{X}, \text{Act}, \text{prob}, \text{inv})$ the semantics of \mathcal{P} is given by the (infinite-state) MDP $\llbracket \mathcal{P} \rrbracket=(S, s_0, \mathbb{R}_+ \times \text{Act}, P_{\llbracket \mathcal{P} \rrbracket}, R_{\llbracket \mathcal{P} \rrbracket})$ where:

- $S=\{(l, v) \in L \times \mathbb{R}_+^{\mathcal{X}} \mid v \models \text{inv}(l)\}$ and $s_0=(l_0, \mathbf{0})$;
- for any $(l, v) \in S$ and $(t, a) \in \mathbb{R}_+ \times \text{Act}$ we have $P_{\llbracket \mathcal{P} \rrbracket}((l, v), (t, a))=\mu$ if and only if $v+t' \models \text{inv}(l)$ for all $0 \leq t' \leq t$, $v+t \models \text{enab}(l, a)$ and for any $(l', v') \in S$:

$$\mu(l', v') = \sum_{R \subseteq \mathcal{X} \wedge v'=(v+t)[R]} \text{prob}(l, a)(R, l')$$

- $R_{\llbracket \mathcal{P} \rrbracket}((l, v), (t, a))=t$ for all $(l, v) \in S$ and $(t, a) \in \mathbb{R}_+ \times \text{Act}$.

For a location-action pair $(l, a) \in L \times \text{Act}$ of a PTA \mathcal{P} , an element $(R, l') \in 2^{\mathcal{X}} \times L$ such that $\text{prob}(l, a)(R, l')>0$ is called an *edge* of (l, a) and the set of edges of (l, a) is denoted $\text{edges}(l, a)$.

As in [31, 5, 32], a transition of a PTA's semantics corresponds to selecting a time-action pair. This differs from the standard approach, e.g. see [12, 7], where in the semantics there are separate “time” and “action” transitions. The choice of definition we have used here is motivated by the fact that, by taking this approach, both the presentation and proofs are greatly simplified. We emphasise, however, that the semantics presented here does not restrict the class of systems that can be modelled. In particular, for any PTA \mathcal{P} with the corresponding MDP \mathcal{M} under the standard semantics, we can construct a PTA in linear time such that its MDP semantics constructed using Definition 8 is equivalent to \mathcal{M} . Essentially, for any location in which the invariant is unbounded (meaning time can diverge), we add a transition to a new location in which time can diverge (by adding a loop which resets all clocks). However, the reverse is not true since using Definition 8 one can model the requirement that, in a location, eventually an action is taken without placing a bound on the time at which the action is taken, while modelling this behaviour is not possible if using the original semantics.

Example 1. Consider the PTA \mathcal{P}_1 shown in Figure 1 where the target set equals $\{l_3\}$. From the state (l_0, v) , if action a is chosen, then the minimum expected

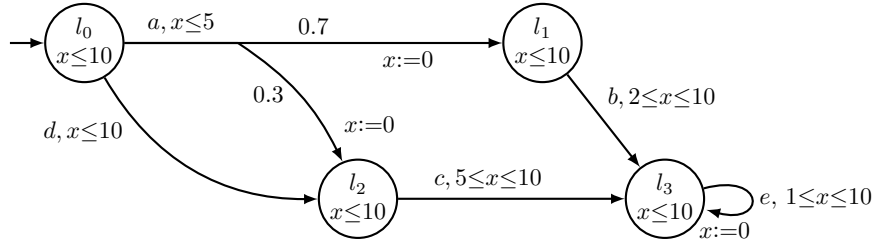


Figure 1: Example PTA \mathcal{P}_1

time equals $0.3 \cdot 5 + 0.7 \cdot 2 = 2.9$. On the other hand, if action d is selected, then the minimum expected time equals $5 - v(x)$ if $v(x) \leq 5$ and 0 otherwise. Therefore, in the initial state, i.e. when $v(x) = 0$, the minimum expected time equals $\min\{2.9, 5 - 0\} = 2.9$. In this example, the optimal choices are to take transitions as soon as they become available. However, as we will see, this does not hold in general since we might need to wait longer in a location in order for an enabling condition to be satisfied later.

Now consider the maximum expected time to reach the target set $\{l_3\}$. If action a is chosen in state (l_0, v) , then $v(x) \leq 5$ and the maximum expected time is $0.3 \cdot (5 - v(x) + 10) + 0.7 \cdot (5 - v(x) + 10) = 15 - v(x)$. On the other hand, if action d is chosen in (l_0, v) , then $v(x) \leq 10$ and the maximum expected time is $10 - v(x)$. Therefore, in the initial state, the maximum expected time equals $\max\{15 - 0, 10 - 0\} = 15$. In this case, we should not choose the action which is available as late as possible (action d) as it reduces the expected time it takes to reach the target. Instead, to achieve the maximum expected time in the initial state, we should wait 5 time units, take action a and then wait as long as possible in locations l_1 and l_2 before taking actions b and c respectively. ■

3.3. Assumptions on PTAs

For Theorem 5 and Corollary 6 to be applicable to the semantics of a PTA, we need to ensure Assumption 4 holds and the optimal value function is bounded (from below in the case of minimum and above in the case of maximum). To this end, we introduce the following assumptions on the PTAs we consider.

Assumption 9. For any PTA \mathcal{P} we have:

- (a) all invariants of \mathcal{P} are bounded;
- (b) only non-strict inequalities are allowed in clock constraints (\mathcal{P} is closed);
- (c) all invariant and enabling conditions of \mathcal{P} are convex;
- (d) \mathcal{P} is structurally non-zeno [33] (this can be identified syntactically and in a compositional fashion [34] and guarantees time-divergent behaviour).

Consider any PTA $\mathcal{P} = (L, l_0, \mathcal{X}, Act, \text{prob}, \text{inv})$ which satisfies Assumption 9 with semantics $\llbracket \mathcal{P} \rrbracket = (S, s_0, A, P_{\llbracket \mathcal{P} \rrbracket}, R_{\llbracket \mathcal{P} \rrbracket})$. Assumption 9(a) and Assumption 9(b)

are necessary and sufficient to ensure $A(s)$ is compact¹ for all states $s \in S$, i.e. Assumption 4(a) holds. Assumption 9(c) is standard for TAs and PTAs and is a technical requirement for proving the correctness of our algorithm for maximum expected reachability (see the proof of Proposition 17).

The fact $\llbracket \mathcal{P} \rrbracket$ satisfies Assumption 4(b) follows from Definition 8 as, for any $(t, a) \in A = (\mathbb{R}_+ \times Act)$, we have $R_{\llbracket \mathcal{P} \rrbracket}(s, (t, a)) = t$ for all $s \in S$. Assumption 9(d) (structurally non-zeno) is sufficient for ensuring that $\llbracket \mathcal{P} \rrbracket$ satisfies Assumption 4(c). More precisely, if for a strategy σ of $\llbracket \mathcal{P} \rrbracket$ the probability of reaching the target is less than 1, then there is a non-negligible set of paths under σ which never reach the target and, since σ is non-zeno, the elapsed time (and hence the accumulated reward) must diverge on all the paths in this set.

The remaining requirements of $\llbracket \mathcal{P} \rrbracket$, Assumption 4(d) and that the optimal values are bounded (see Theorem 5 and Corollary 6), hold if we restrict attention to a specific sub-MDP of $\llbracket \mathcal{P} \rrbracket$. More precisely, in the case of minimum expected reachability, we restrict to the sub-MDP which contains the states $s \in S$ for which $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\max}(s, F) = 1$ and, in the case of maximum expected reachability, to the sub-MDP which contains the states $s \in S$ for which $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}(s, F) = 1$. In either case, it follows from the region graph construction [3] that there exists a memoryless, deterministic strategy that reaches the target with probability 1 from all states of the sub-MDP, and hence this strategy will also be proper, meaning Assumption 4(d) holds. Regarding the optimal values, clearly the minimum values are bounded below as all rewards are non-negative. For maximum values, using either the digital clocks approach [12] or the region graph construction [3], we can show that the maximum values are less than or equal to maximum expected reachability reward values on a finite-state MDP, preserving optimal reachability probabilities. Therefore, as we have restricted to states for which the target is reached with minimum probability 1, the maximum values are bounded above [24].

Assumption 9 imposed several restrictions on PTAs we analyse. However, boundedness is not actually a restriction since bounded TAs are as expressive as standard TAs [9] and this result carries over to PTAs. Also, the fact that PTAs must be closed is not a severe restriction, as any PTA can be infinitesimally approximated by one with closed constraints [35]. However, since it relies on the region graph construction [36, 1], it can lead to an exponential blow-up, and therefore has practical implications. Non-zenoness is a standard assumption for both TAs and PTAs, as it prevents unrealistic behaviours, i.e. executions for which time does not diverge.

A final assumption on PTAs, which is again standard, is that we assume that they are *well-formed*. This means that, for each state (l, v) , action a such that $v \models \text{enab}(l, a)$ and edge $(R, l') \in \text{edges}(l, a)$, we have $v[R] \models \text{inv}(l')$, i.e. all transitions lead to valid states.

¹Recall a subset of \mathbb{R}_+ is compact if it is closed and bounded.

4. Optimal Expected Time Controller Synthesis for PTAs

In this section we present our algorithms for the optimal expected time computation and controller synthesis for PTAs based on a backwards exploration of the state space. We adopt backwards as opposed to forwards search since, although forwards has proven successful in the context of TAs, for PTAs it yields only upper bounds for maximum probabilistic reachability [3]. For the remainder of the section we fix a PTA $\mathcal{P}=(L, l_0, \mathcal{X}, Act, \text{enab}, \text{prob}, \text{inv})$, target set of locations F and suppose $\llbracket \mathcal{P} \rrbracket=(S, s_0, \mathbb{R}_+ \times Act, P_{\llbracket \mathcal{P} \rrbracket}, R_{\llbracket \mathcal{P} \rrbracket})$ and $S_F=\{(l, v) \in L \times \mathbb{R}_+^{\mathcal{X}} \mid l \in F \wedge v \models \text{inv}(l)\}$.

We first define symbolic states and the operations we require on them. Next we present the backwards reachability algorithm for generation of the zone graph. Following this we show how value iteration over the zone graph can be used for optimal expected time reachability computation. We then introduce rational simple and rational nice functions to represent the functions encountered during this computation, and finally give our approach for controller synthesis.

4.1. Symbolic States and Operations

A symbolic state \mathbf{z} of \mathcal{P} is a location-zone pair $(l, \zeta) \in L \times \text{Zones}(\mathcal{X})$ representing the set of PTA states $\{(l, v) \in \{l\} \times \mathbb{R}_+^{\mathcal{X}} \mid v \models \zeta \wedge \text{inv}(l)\}$. Let $\mathbf{Z}_F \stackrel{\text{def}}{=} \{(l, \text{inv}(l)) \mid l \in F\}$, i.e. the symbolic states representing the target set. For any symbolic states $\mathbf{z}=(l, \zeta)$ and $\mathbf{z}'=(l, \zeta')$ let $\mathbf{z} \wedge \mathbf{z}'=(l, \zeta \wedge \zeta')$, $\mathbf{z} \subseteq \mathbf{z}'$ if and only if $\zeta \subseteq \zeta'$ and $\mathbf{z}=\emptyset$ if and only if $\zeta=\text{false}$. The time and discrete predecessor operations for a symbolic state $\mathbf{z}=(l, \zeta)$, locations l' and l'' , action a and set of clocks R are defined as follows:

$$\begin{aligned} \text{tpre}(\mathbf{z}) &\stackrel{\text{def}}{=} (l, \text{inv}(l) \wedge \surd \zeta) \\ \text{dpre}(l', a, (R, l''))(\mathbf{z}) &\stackrel{\text{def}}{=} \begin{cases} (l', \text{false}) & \text{if } l \neq l'' \\ (l', \text{enab}(l', a) \wedge [R]\zeta) & \text{otherwise.} \end{cases} \end{aligned}$$

4.2. Backward Reachability Algorithm

We use a slightly modified version of the backward reachability algorithm taken from [7] (the same operations are performed, we just add action labels to the edge tuples). The modified version is given in Figure 2.

The backwards algorithm returns a zone graph $\mathbf{G}=(\mathbf{Z}, \mathbf{E})$ with symbolic states as vertices. Termination of the algorithm is guaranteed by the fact that only finitely many zones can be generated. As demonstrated in [7], from this graph one can build a finite state MDP $\llbracket \mathbf{G} \rrbracket$ for computing the exact maximum probabilistic reachability values of $\llbracket \mathcal{P} \rrbracket$. The MDP $\llbracket \mathbf{G} \rrbracket$ has state space \mathbf{Z} , action set 2^E and, if $\mathbf{z} \in \mathbf{Z}$ and $E \in 2^E$, then $P_{\llbracket \mathbf{G} \rrbracket}(\mathbf{z}, E)$ is defined if and only if there exists $a \in Act$ such that both the following conditions hold:

- $(\mathbf{z}'', a', (R, l'), \mathbf{z}') \in E$ implies $\mathbf{z}''=\mathbf{z}$ and $a'=a$;
- $(\mathbf{z}, a, (R, l'), \mathbf{z}') \neq (\mathbf{z}, a, (\tilde{R}, \tilde{l}'), \tilde{\mathbf{z}}') \in E$ implies $(R, l') \neq (\tilde{R}, \tilde{l}')$;

BackwardsReach(\mathcal{P}, F)

```

1  Z := ∅
2  E := ∅
3  Y := {(l, inv(l)) | l ∈ F}
4  while (Y ≠ ∅)
5    choose (y ∈ Y)
6    Y := Y \ {y}
7    Z := Z ∪ {y}
8    for ((l, a) ∈ (L \ F) × Act) and ((R, l') ∈ edges(l, a))
9      z := dpre(l, a, R, l')(tpre(y))
10     if (z ≠ ∅)
11       if (z ∉ Z)
12         Y := Y ∪ {z}
13         E := E ∪ {(z, a, (R, l'), y)}
14         for ((z̃, a, (R̃, l̃'), ỹ) ∈ E) such that ((R̃, l̃') ≠ (R, l'))
15           if ((z ∧ z̃ ≠ ∅) ∧ (z ∧ z̃ ∉ Z))
16             Y := Y ∪ {z ∧ z̃}
17     for (z ∈ Z) and ((z', a, (R, l'), z'') ∈ E)
18       if (z ⊆ z')
19         E := {(z, a, (R, l'), z'')} ∪ E
20     return G := (Z, E)

```

Figure 2: Backward reachability algorithm

where $P_{\llbracket \mathbb{G} \rrbracket}(z, E)(z') = \sum \{\text{prob}(l, a)(R, l') \mid (z, a, (R, l'), z') \in E\}$ for $z' \in Z$.

The following theorem shows the correspondence between the maximum probabilistic reachability values for $\llbracket \mathcal{P} \rrbracket$ and $\llbracket \mathbb{G} \rrbracket$.

Theorem 10 ([7]). *If $\mathbb{G} = (Z, E)$ is the zone graph returned by BackwardsReach(\mathcal{P}, F), then for any state s of $\llbracket \mathcal{P} \rrbracket$ we have:*

- $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\max}(s, S_F) > 0$ if and only if there exists $z \in Z$ such that $s \in \text{tpre}(z)$;
- if $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\max}(s, S_F) > 0$, then $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\max}(s, S_F) = \max \{\mathbb{P}_{\llbracket \mathbb{G} \rrbracket}^{\max}(z, Z_F) \mid z \in Z \wedge s \in \text{tpre}(z)\}$.

Given a zone graph $\mathbb{G} = (Z, E)$, for any $(l, \zeta) \in Z$ let $E(l, \zeta) \subseteq 2^E$ represent the following sets of edges: $E \in E(l, \zeta)$ if and only if there exists $a \in \text{Act}$ such that $\text{edges}(l, a) = \{(R_1, l_1), \dots, (R_n, l_n)\}$ and

$$E = \{(z, a, (R_1, l_1), z_1), \dots, (z, a, (R_n, l_n), z_n)\}$$

for some $z_1, \dots, z_n \in Z$.

Example 2. Consider the PTA \mathcal{P}_2 given in Figure 3, which is adapted from an example presented in [7]. For the target set $\{l_2\}$, after following the backwards algorithm, the resulting finite-state MDP $\llbracket \mathbb{G} \rrbracket$ is presented in Figure 4. In the figure, the thicker arrows correspond to the edges generated in the main

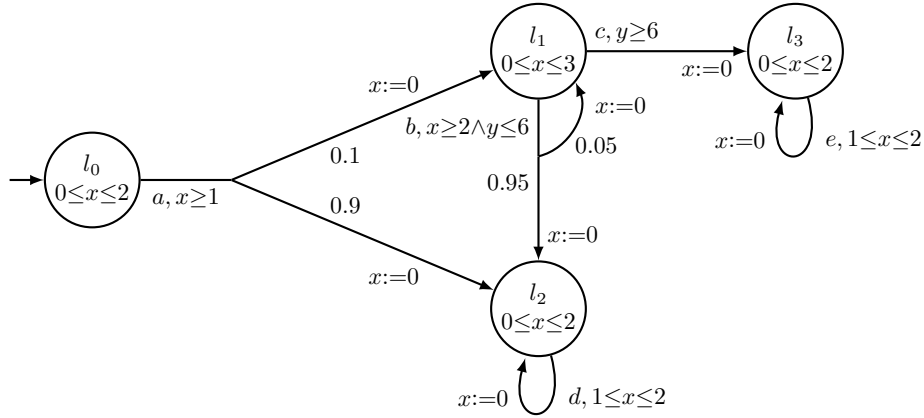


Figure 3: Example PTA \mathcal{P}_2 (adapted from [7])

algorithm (see Figure 2), while the remaining arrows correspond to the edges generated during the MDP construction that follows. Using Theorem 10 it follows that, from the initial state $(l_0, \mathbf{0})$, the maximum probability of reaching the target $\{l_2\}$ equals 0.99525 and corresponds to the maximum probability of the symbolic state $(l_0, 1 \leq x \leq 2 \wedge y < 3)$ reaching the target set in the MDP of Figure 4. This maximum probability is achieved by always taking the actions a and b in locations l_0 and l_1 as soon as they become enabled. ■

4.3. Minimum Expected Time Computation using the Zone Graph

We now consider the case of computing minimum expected time reachability values. The first step in the computation is to find those states for which the minimum expected time to reach the target is finite, i.e. states for which the maximum probability of reaching the target is 1. For states for which the maximum reachability probability is less than 1, since we assume \mathcal{P} is non-zero (Assumption 9(d)) the minimum expected time to reach the target in these states equals infinity. Using Theorem 10 we can find the states s of $\llbracket \mathcal{P} \rrbracket$ for which $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\max}(s, S_F) = 1$ by computing the symbolic states \mathbf{z} for which $\mathbb{P}_{\llbracket \mathbf{G} \rrbracket}^{\max}(\mathbf{z}, Z_F) = 1$. Finding these symbolic states does not require numerical computation [37], and hence we do not need to build $\llbracket \mathbf{G} \rrbracket$, but can use \mathbf{G} directly in the computation.

For the remainder of this section we assume we have computed the states of $\llbracket \mathbf{G} \rrbracket$, and hence of $\llbracket \mathcal{P} \rrbracket$, for which the maximum reachability probability is 1, and $\llbracket \mathcal{P} \rrbracket_{\min}$ and $\mathbf{G}_{\min} = (Z_{\min}, E_{\min})$ are the sub-MDP and sub-graph restricted to these states. Using Theorem 10, $s \in S_{\min}$ if and only if there exists $\mathbf{z} \in Z_{\min}$ such that $s \in \text{tpre}(\mathbf{z})$. Since the minimum expected time to reach the target is infinity for the states not considered, if we compute the minimum expected time reachability values for the states of the constructed sub-MDP, we will have found the values for all states of $\llbracket \mathcal{P} \rrbracket$.

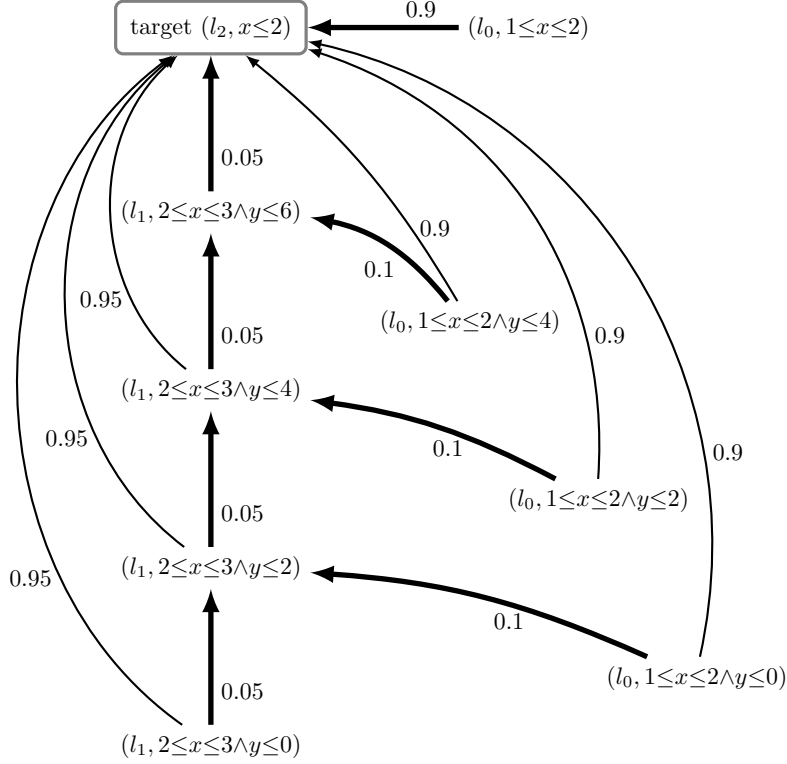


Figure 4: MDP $\mathbb{[G]}$ generated for the backwards algorithm for the PTA \mathcal{P}_2 and target $\{l_2\}$

Following the discussion in Section 3.3, $\mathbb{[P]}_{\min}$ now satisfies Assumption 4 and we can use Theorem 5. In particular, value iteration for the Bellman operator $T_{\mathbb{[P]}_{\min}}^{\min}$ (see Definition 3) for the target set S_F converges to the minimum expected time when starting from any bounded function. We now present a value iteration method over \mathbb{G}_{\min} and prove a correspondence with value iteration using $T_{\mathbb{[P]}_{\min}}^{\min}$.

Definition 11. *The operator $T_{\mathbb{G}_{\min}}^{\min} : (\mathbb{Z}_{\min} \rightarrow (S_{\min} \rightarrow \mathbb{R}_+)) \rightarrow (\mathbb{Z}_{\min} \rightarrow (S_{\min} \rightarrow \mathbb{R}_+))$ on the zone graph \mathbb{G}_{\min} is such that for $g : \mathbb{Z}_{\min} \rightarrow (S_{\min} \rightarrow \mathbb{R}_+)$, $\mathbf{z} = (l, \zeta) \in \mathbb{Z}_{\min}$ and $s = (l, v) \in S_{\min}$ where $s \in \text{tpre}(\mathbf{z})$ we have $T_{\mathbb{G}_{\min}}^{\min}(g)(\mathbf{z})(s)$ equals 0 if $l \in F$ and otherwise equals*

$$\inf_{t \in \mathbb{R}_+ \wedge v + t \in \zeta} \min_{E \in \mathbb{E}(\mathbf{z})} \left\{ t + \sum_{(\mathbf{z}', a), (R, l'), \mathbf{z}' \in E} \text{prob}(l, a)(R, l') \cdot g(\mathbf{z}')(l', (v+t)[R]) \right\}.$$

Proposition 12. *If $f : S_{\min} \rightarrow \mathbb{R}_+$ and $g : \mathbb{Z}_{\min} \rightarrow (S_{\min} \rightarrow \mathbb{R}_+)$ are functions such that $f(s) = g(\mathbf{z})(s)$ for all $s \in S_{\min}$ and $\mathbf{z} \in \mathbb{Z}_{\min}$ such that $s \in \text{tpre}(\mathbf{z})$, then for any $s \in S_{\min}$ and $n \in \mathbb{N}$ we have:*

$$(T_{\mathbb{[P]}_{\min}}^{\min})^n(f)(s) = \min\{ (T_{\mathbb{G}_{\min}}^{\min})^n(g)(\mathbf{z})(s) \mid \mathbf{z} \in \mathbb{Z}_{\min} \wedge s \in \text{tpre}(\mathbf{z}) \}.$$

PROOF. Consider any $f : S_{\min} \rightarrow \mathbb{R}_+$ and $g : Z_{\min} \rightarrow (S_{\min} \rightarrow \mathbb{R}_+)$ such that $f(s) = g(\mathbf{z})(s)$ for all $s \in S_{\min}$ and $\mathbf{z} \in Z_{\min}$ such that $s \in \text{tpre}(\mathbf{z})$. The proof is by induction on $n \in \mathbb{N}$. If $n=0$, then the result follows by construction of f and g and since $(T_{[\mathcal{P}]_{\min}}^{\min})^0(f) = f$ and $(T_{G_{\min}}^{\min})^0(g) = g$.

Next we assume the proposition holds for some $n \in \mathbb{N}$. For any $s=(l, v) \in S_{\min}$, if $l \in F$, then by the construction of the zone graph (see Figure 2, line 3), Definition 3 and Definition 11 we have:

$$(T_{[\mathcal{P}]_{\min}}^{\min})^{n+1}(f)(s) = 0 = \min \{ (T_{G_{\min}}^{\min})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in Z_{\min} \wedge s \in \text{tpre}(\mathbf{z}) \}.$$

It therefore remains to consider the case when $s=(l, v) \in S_{\min}$ and $l \notin F$. Now consider any $(t', a') \in A(s)$. By construction of $[\mathcal{P}]_{\min}$, for any $(R, l') \in E(l, a')$ we have $s' = (l', (v+t')[R]) \in \text{tpre}(\mathbf{z}')$ for some $\mathbf{z}' \in Z_{\min}$ (as otherwise $s' \notin S_{\min}$, and hence the minimum expected time to reach F from s' is infinite).

Now, for any $(R, l') \in E(l, a')$, by the induction hypothesis there exists $\mathbf{z}'_{(R, l')} = (l', \zeta_{(R, l')}) \in Z_{\min}$ with $(l', (v+t')[R]) \in \text{tpre}(\mathbf{z}'_{(R, l')})$ such that:

$$(T_{G_{\min}}^{\min})^n(g)(\mathbf{z}'_{(R, l')})(l', (v+t')[R]) = (T_{[\mathcal{P}]_{\min}}^{\min})^n(f)(l', (v+t')[R]). \quad (1)$$

Since $(t', a') \in A(s)$ and $(l', (v+t')[R]) \in \text{tpre}(l', \zeta_{(R, l')})$, it follows from Definition 8 that $(l, v+t) \in \text{dpre}(l, a', (R, l'))(\text{tpre}(\mathbf{z}'_{(R, l')}))$.

Given that the edge $(R, l') \in \text{edges}(l, a')$ was arbitrary, by the construction of the zone graph (see Figure 2, lines 8–19), there exists $\mathbf{z}=(l, \zeta) \in Z_{\min}$ such that $v+t' \in \zeta$ and edge set:

$$E' = \{ (\mathbf{z}, a', (R, l'), \mathbf{z}'_{(R, l')}) \mid (R, l') \in \text{edges}(l, a') \} \in E(\mathbf{z}). \quad (2)$$

Furthermore, by definition of tpre we have $s \in \text{tpre}(\mathbf{z})$. Now, by Definition 11, $(T_{G_{\min}}^{\min})^{n+1}(g)(\mathbf{z})(s)$ equals:

$$\begin{aligned} & \inf_{t \in \mathbb{R}_+ \wedge v+t \in \zeta} \min_{E \in E(\mathbf{z})} \left\{ t + \sum_{(\mathbf{z}, a', (R, l'), \mathbf{z}') \in E} \text{prob}(l, a')(R, l') \cdot (T_{G_{\min}}^{\min})^n(g)(\mathbf{z}')(l', (v+t)[R]) \right\} \\ & \leq \min_{E \in E(\mathbf{z})} \left\{ t' + \sum_{(\mathbf{z}, a', (R, l'), \mathbf{z}') \in E} \text{prob}(l, a')(R, l') \cdot (T_{G_{\min}}^{\min})^n(g)(\mathbf{z}')(l', (v+t')[R]) \right\} \\ & \hspace{20em} (\text{since } v+t' \in \zeta) \\ & \leq t' + \sum_{(\mathbf{z}, a', (R, l'), \mathbf{z}'_{(R, l')}) \in E'} \text{prob}(l, a')(R, l') \cdot (T_{G_{\min}}^{\min})^n(g)(\mathbf{z}'_{(R, l')})(l', (v+t')[R]) \\ & \hspace{20em} (\text{since } E' \in E(\mathbf{z})) \\ & = t' + \sum_{(R, l') \in \text{edges}(l, a')} \text{prob}(l, a')(R, l') \cdot (T_{[\mathcal{P}]_{\min}}^{\min})^n(f)(l', (v+t')[R]) \\ & \hspace{20em} (\text{by (1) and (2)}) \\ & = R_{[\mathcal{P}]_{\min}}(s, (t', a')) + \sum_{s' \in S_{\min}} P_{[\mathcal{P}]_{\min}}(s, (t', a'))(s') \cdot (T_{[\mathcal{P}]_{\min}}^{\min})^n(f)(s') \\ & \hspace{20em} (\text{by Definition 8}) \end{aligned}$$

Therefore, since $(t', a') \in A(s)$ was arbitrary, it follows from Definition 3 that:

$$(T_{[\mathcal{P}]_{\min}}^{\min})^{n+1}(f)(s) \geq \min \{ (T_{\mathbf{G}_{\min}}^{\min})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in \mathbf{Z}_{\min} \wedge s \in \mathbf{tpre}(\mathbf{z}) \}. \quad (3)$$

Next we consider any $\mathbf{z}=(l, \zeta) \in \mathbf{Z}_{\min}$ such that $v+t \in \zeta$ for some $t \in \mathbb{R}_+$ (i.e. $\mathbf{z} \in \mathbf{Z}_{\min}$ such that $s \in \mathbf{tpre}(\mathbf{z})$). For any $t' \in \mathbb{R}_+$ such that $v+t' \in \zeta$ and $E' \in \mathbf{E}(l, \zeta)$ by construction of the zone graph (see Figure 2, lines 8–19) there exists $a' \in Act$ where:

$$E' = \{ (\mathbf{z}, a', (R, l'), \mathbf{z}'_{(R, l')}) \mid (R, l') \in \mathbf{edges}(l, a') \} \quad (4)$$

and $(l', (v+t')[R]) \in \mathbf{tpre}(\mathbf{z}'_{(R, l')})$ for all $(R, l') \in \mathbf{edges}(l, a')$. We have by the induction hypothesis for any $(R, l') \in \mathbf{edges}(l, a')$:

$$(T_{[\mathcal{P}]_{\min}}^{\min})^n(f)(l', (v+t')[R]) \leq (T_{\mathbf{G}_{\min}}^{\min})^n(g)(\mathbf{z}'_{(R, l')})(l', (v+t')[R]). \quad (5)$$

Furthermore, by Definition 8 we have $(t', a') \in A(s)$. Now by Definition 3:

$$\begin{aligned} & (T_{[\mathcal{P}]_{\min}}^{\min})^{n+1}(f)(s) \\ &= \inf_{(t, a) \in A(l, v)} \left\{ R_{[\mathcal{P}]_{\min}}(s, (t, a)) + \sum_{s' \in S_{\min}} P_{[\mathcal{P}]_{\min}}(s, (t, a))(s') \cdot (T_{[\mathcal{P}]_{\min}}^{\min})^n(f)(s') \right\} \\ &\leq R_{[\mathcal{P}]_{\min}}(s, (t', a')) + \sum_{s' \in S_{\min}} P_{[\mathcal{P}]_{\min}}(s, (t', a'))(s') \cdot (T_{[\mathcal{P}]_{\min}}^{\min})^n(f)(s') \\ &\hspace{15em} (\text{since } (t', a') \in A(s)) \\ &= t' + \sum_{(R, l') \in \mathbf{edges}(l, a')} \mathbf{prob}(l, a')(R, l') \cdot (T_{[\mathcal{P}]_{\min}}^{\min})^n(f)(l', (v+t')[R]) \\ &\hspace{15em} (\text{by Definition 8}) \\ &\leq t' + \sum_{(R, l') \in \mathbf{edges}(l, a')} \mathbf{prob}(l, a')(R, l') \cdot (T_{\mathbf{G}_{\min}}^{\min})^n(g)(\mathbf{z}'_{(R, l')})(l', (v+t')[R]) \\ &\hspace{15em} (\text{by (5)}) \\ &= t' + \sum_{(\mathbf{z}, a, (R, l'), \mathbf{z}'_{(R, l')}) \in E'} \mathbf{prob}(l, a')(R, l') \cdot (T_{\mathbf{G}_{\min}}^{\min})^n(g)(\mathbf{z}'_{(R, l')})(l', (v+t')[R]) \\ &\hspace{15em} (\text{by (4)}) \end{aligned}$$

Since $\mathbf{z}=(l, \zeta) \in \mathbf{Z}_{\min}$ such that $v+t \in \zeta$ for some $t \in \mathbb{R}_+$, $t' \in \mathbb{R}_+$ such that $v+t' \in \zeta$ and $E' \in \mathbf{E}(l, \zeta)$ were arbitrary, by Definition 11 it follows that:

$$(T_{[\mathcal{P}]_{\min}}^{\min})^{n+1}(f)(s) \leq \min \{ (T_{\mathbf{G}_{\min}}^{\min})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in \mathbf{Z}_{\min} \wedge s \in \mathbf{tpre}(\mathbf{z}) \}. \quad (6)$$

Combining (3) and (6) we have:

$$(T_{[\mathcal{P}]_{\min}}^{\min})^{n+1}(f)(s) = \min \{ (T_{\mathbf{G}_{\min}}^{\min})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in \mathbf{Z}_{\min} \wedge s \in \mathbf{tpre}(\mathbf{z}) \}$$

and hence, since $s \in S_{\min}$ was arbitrary, the proposition holds by induction. \square

MinProbReach_{>0}(\mathcal{P}, F)

```

1  for ( $l \in L$ )
2    if ( $l \in F$ )
3       $\zeta_l := \text{inv}(l)$ 
4    else
5       $\zeta_l := \text{false}$ 
6     $\langle \xi_l \rangle_{l \in L} := \langle \text{false} \rangle_{l \in L}$ 
7    while ( $\langle \zeta_l \rangle_{l \in L} \neq \langle \xi_l \rangle_{l \in L}$ )
8       $\langle \xi_l \rangle_{l \in L} := \langle \zeta_l \rangle_{l \in L}$ 
9      for ( $l \in L \setminus F$ )
10     for ( $a \in \text{Act}$ )
11       if ( $\text{enab}(l, a) \neq \text{false}$ )
12          $\zeta := \text{enab}(l, a)$ 
13         for ( $(R, l') \in \text{edges}(l, a)$ )
14            $\zeta := \zeta \wedge [R](\text{inv}(l') \setminus \xi_{l'})$ 
15            $\zeta := \text{inv}(l) \wedge \zeta$ 
16            $\zeta_l := \zeta_l \setminus \zeta$ 
17        $\zeta_l := \xi_l \vee \zeta_l$ 
18  return  $\langle \zeta_l \rangle_{l \in L}$ 

```

Figure 5: Algorithm for minimum probability of reaching the target is greater than 0

4.4. Maximum Expected Time Computation using the Zone Graph

In this section we use the zone graph (see Section 4.2) for computing maximum expected time reachability values. The first step is to restrict the graph to those states for which the maximum expected time to reach the target is finite, that is, those states for which the minimum probability of reaching the target equals 1. One way to achieve this would be to use the algorithms presented in [7]. However, as we have restricted attention to structurally non-zero PTAs (see Assumption 9(d)), we can instead consider a simpler alternative based on algorithms developed for MDPs. More precisely, we can extend the algorithm presented in [37] for finding the states of an MDP for which the minimum probability of reaching the target equals 1. This algorithm requires as input the set of states for which the minimum probability of reaching the target is greater than 0, for which an algorithm is also presented in [37]. Our extensions of these algorithms to structurally non-zero PTAs are shown in Figures 5 and 6.

The main difference between the algorithms of [37] for MDPs and the extension to non-zero PTAs presented here is in algorithm MinProbReach_{>0} (see Figure 5) when finding the states for which, for all available actions of an MDP or time-action pairs of a PTA, one remains within the currently computed set of states with probability greater than 0. More precisely, for PTAs we first find the complement set, i.e. the set of states for which, for at least one available time-action pair, the probability of leaving the currently computed states is 1. This is due to the fact that this complement set is straightforward to compute

MinProbReach₌₁(\mathcal{P}, F)

```

1   $\langle \zeta_l \rangle := \text{MinProbReach}_{>0}(\mathcal{P}, F)$ 
2   $\langle \xi_l \rangle_{l \in L} := \langle \text{inv}(l) \rangle_{l \in L}$ 
3  while ( $\langle \zeta_l \rangle_{l \in L} \neq \langle \xi_l \rangle_{l \in L}$ )
4     $\langle \xi_l \rangle_{l \in L} := \langle \zeta_l \rangle_{l \in L}$ 
5    for ( $l \in L \setminus F$ )
6       $\zeta := \text{false}$ 
7      for ( $a \in \text{Act}$ )
8        if ( $\text{enab}(l, a) \neq \text{false}$ )
9          for ( $(R, l') \in \text{edges}(l, a)$ )
10              $\zeta := \zeta \vee (\surd(\text{enab}(l) \wedge [R](\text{inv}(l') \setminus \xi_{l'})))$ 
11     $\zeta_l := \zeta_l \setminus \zeta$ 
12  return  $\langle \zeta_l \rangle_{l \in L}$ 

```

Figure 6: Algorithm for minimum probability of reaching the target equals 1

using zone operations, and it is not apparent that the original set can be computed directly using zone operations. To compute this complement set, we first find, for each location l and action a , the clock valuations v such that, when performing the action a in the state (l, v) all corresponding edges lead one out of the currently computed set of states (lines 11–14). Second, we take the time predecessor to find the clock valuations v such that for the state (l, v) there exists a time-action pair (t, a) for which the probability of leaving the currently computed set of states equals 1. Since these states are in the complement of what we require, each time we remove these states from those under consideration (line 16) and, for each location l , once all actions have been considered, add the remaining set of states to the currently computed set of states (line 17).

The algorithm $\text{MinProbReach}_{=1}$ (see Figure 6) is a more straightforward extension of the algorithm presented in [37] for MDPs. For this algorithm we need to find those states for which there exists a time-action pair which leaves the currently computed set of states with probability greater than 0 and then remove these states from the currently computed set of states. This can be achieved straightforwardly using zone operations by considering each location and enabled action in turn (lines 5–11).

Convergence of the presented algorithms follows from the region graph construction for PTAs [3] and the fact that both algorithms are monotone in the sense that at, after each iteration of $\text{MinProbReach}_{>0}(\mathcal{P}, F)$, we have $\xi_l \subseteq \zeta_l$ for all $l \in L$ and, after each iteration of $\text{MinProbReach}_{=1}(\mathcal{P}, F)$, we have $\xi_l \supseteq \zeta_l$ for all $l \in L$. After demonstrating convergence, the correctness of these algorithms follows as for the MDP case [37]. Formally, we have the following results.

Proposition 13. *If $\langle \zeta_l \rangle_{l \in L}$ are the zones returned by $\text{MinProbReach}_{>0}(\mathcal{P}, F)$, then for any $s=(l, v) \in S$ we have $\mathbb{P}_{\mathcal{M}}^{\min}(s, S_F) > 0$ if and only if $v \in \zeta_l$.*

Proposition 14. *If $\langle \zeta_l \rangle_{l \in L}$ are the zones returned by $\text{MinProbReach}_{=1}(\mathcal{P}, F)$, then for any $s=(l, v) \in S$ we have $\mathbb{P}_{\mathcal{M}}^{\min}(s, S_F) = 1$ if and only if $v \in \zeta_l$.*

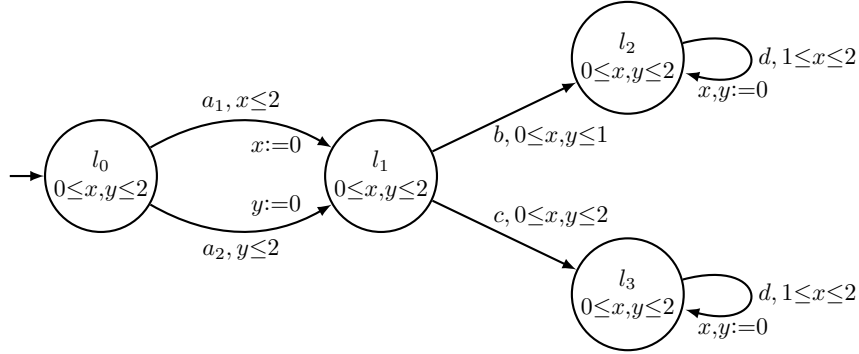


Figure 7: Example PTA \mathcal{P}_3 (see Example 3)

For both of the presented algorithms the operations applied to zones include disjunction and set difference, and hence non-convex zones can be introduced. This is unavoidable, even under Assumption 9(c), as the following example demonstrates.

Example 3. Consider the PTA \mathcal{P}_3 in Figure 7 where the target set equals $\{l_3\}$. The non-convex zone ζ , representing the set of clock valuations v such that the minimum probability of reaching the target from (l_1, v) equals 1, is given in Figure 8. It is obtained by preventing the choice of action b , i.e. removing the part of the zone in which the enabling condition of a transition labelled b is satisfied (or becomes satisfied after letting time pass). ■

For the remainder of this section we assume that, using the algorithms presented in Figures 5 and 6, we have computed the states S_{\max} of $\llbracket \mathcal{P} \rrbracket$, for which the minimum reachability probability is 1 and $\llbracket \mathcal{P} \rrbracket_{\max}$ is the sub-MDP restricted to those states. As discussed in Section 3.3, $\llbracket \mathcal{P} \rrbracket_{\max}$ now satisfies Assumption 4 and we can apply Corollary 6. For states not considered, i.e. states for which the minimum reachability probability is less than 1, since we assume \mathcal{P} is non-zero (Assumption 9(d)), the maximum expected time to reach the target equals infinity for these states. Therefore, if we compute the maximum expected time reachability values for the states of the constructed sub-MDP, we will have found the values for all states of $\llbracket \mathcal{P} \rrbracket$. Furthermore, we assume we have computed the zone graph (see Figure 2), restrict the symbolic states of the zone graph to represent only states of $\llbracket \mathcal{P} \rrbracket$ for which the minimum probability of reaching the target equals 1 and the resulting zone graph is given by $\mathbf{G}_{\max} = (\mathbf{Z}_{\max}, \mathbf{E}_{\max})$.

Next we introduce a technical lemma concerning $\llbracket \mathcal{P} \rrbracket_{\max}$ that we will require when proving the correspondence between value iteration over $\llbracket \mathcal{P} \rrbracket_{\max}$ using Definition 3 and value iteration over \mathbf{G}_{\max} that we will introduce.

Lemma 15. *If $(l, v) \in S_{\max}$ and $t \in \mathbb{R}_+$ such that $l \notin L_F$ and $(l, v+t) \in S$, then $(l, v+t) \in S_{\max}$. Furthermore if $v+t \models \text{enab}(l, a)$ for some $a \in \text{Act}$, then for any $(R, l') \in \text{edges}(l, a)$ we have $(l', (v+t)[R]) \in S_{\max}$.*

PROOF. Consider any $(l, v) \in S_{\max}$ and $t \in \mathbb{R}_+$ such that $l \notin L_F$ and $(l, v+t) \in$

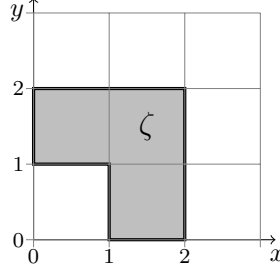


Figure 8: Non-convex zone ζ (see Example 3)

S . Using Assumption 9(c), for any strategy σ we can construct a strategy σ' such that:

$$\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\sigma'}((l, v), S_F) = \mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\sigma}((l, v+t), S_F)$$

by, under σ' , from (l, v) first letting t time units elapse and then following the choices of σ when starting from $(l, v+t)$. Therefore, by definition, it follows that $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}((l, v), S_F) \leq \mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}((l, v+t), S_F)$. Hence, since $(l, v) \in S_{\max}$, we have $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}((l, v+t), S_F) \geq 1$ and $(l, v+t) \in S_{\max}$ as required.

Now suppose that $a \in Act$ such that $v+t \models \text{enab}(l, a)$ and $(l', R) \in \text{edges}(l, a)$. Using Corollary 6 we have:

$$\begin{aligned} \mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}((l, v), S_F) &= \inf_{a \in A(l, v)} \left\{ \sum_{s' \in S} P_{\llbracket \mathcal{P} \rrbracket}(s, a)(s') \cdot \mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}((l', (v+t)[R']), S_F) \right\} \\ &\geq \sum_{(R', l'') \in \text{edges}(l, a)} \text{prob}(l, a)(R', l'') \cdot \mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}((l'', (v+t)[R']), S_F) \quad (\text{by Definition 8.}) \end{aligned}$$

Therefore, since $(l, v) \in S_{\max}$, $\text{prob}(l, a)$ is a probability distribution, it follows that $\mathbb{P}_{\llbracket \mathcal{P} \rrbracket}^{\min}((l', (v+t)[R']), S_F) = 1$ as required. \square

We are now in a position to present a value iteration operator for \mathbf{G}_{\max} and demonstrate a correspondence with value iteration over $\llbracket \mathcal{P} \rrbracket_{\max}$ using $T_{\llbracket \mathcal{P} \rrbracket_{\max}}^{\max}$ (see Definition 3).

Definition 16. *The operator $T_{\mathbf{G}_{\max}}^{\max} : (\mathbf{Z}_{\max} \rightarrow (S_{\max} \rightarrow \mathbb{R}_+)) \rightarrow (\mathbf{Z}_{\max} \rightarrow (S_{\max} \rightarrow \mathbb{R}_+))$ on the zone graph $\mathbf{G}_{\max} = (\mathbf{Z}_{\max}, \mathbf{E}_{\max})$ is such that for $g : \mathbf{Z}_{\max} \rightarrow (S_{\max} \rightarrow \mathbb{R}_+)$, $\mathbf{z} = (l, \zeta) \in \mathbf{Z}_{\max}$ and $s = (l, v) \in S_{\max}$ where $s \in \text{tpre}(\mathbf{z})$ we have $T_{\mathbf{G}_{\max}}^{\max}(g)(\mathbf{z})(s)$ equals 0 if $l \in F$ and otherwise equals*

$$\sup_{t \in \mathbb{R}_+ \wedge v+t \in \zeta} \max_{E \in \mathbf{E}(l, \zeta)} \left\{ t + \sum_{(z, a, (R, l'), z') \in E} \text{prob}(l, a)(R, l') \cdot g(\mathbf{z}')(l', (v+t)[R]) \right\}.$$

Proposition 17. *If $f : S_{\max} \rightarrow \mathbb{R}_+$ and $g : \mathbf{Z}_{\max} \rightarrow (S_{\max} \rightarrow \mathbb{R}_+)$ are functions such that $f(s) = g(\mathbf{z})(s)$ for all $s \in S_{\max}$ and $\mathbf{z} \in \mathbf{Z}_{\max}$ such that $s \in \text{tpre}(\mathbf{z})$, then for any $s \in S_{\max}$ and $n \in \mathbb{N}$ we have:*

$$(T_{\llbracket \mathcal{P} \rrbracket_{\max}}^{\max})^n(f)(s) = \max\{ (T_{\mathbf{G}_{\max}}^{\max})^n(g)(\mathbf{z})(s) \mid \mathbf{z} \in \mathbf{Z}_{\max} \wedge s \in \text{tpre}(\mathbf{z}) \}.$$

PROOF. Consider any $f : S_{\max} \rightarrow \mathbb{R}_+$ and $g : Z_{\max} \rightarrow (S_{\max} \rightarrow \mathbb{R}_+)$ such that $f(s) = g(\mathbf{z})(s)$ for all $s \in S_{\max}$ and $\mathbf{z} \in Z_{\max}$ such that $s \in \mathbf{tpre}(\mathbf{z})$. The proof is by induction on $n \in \mathbb{N}$. If $n=0$, then the result follows by construction of f and g and since $(T_{\mathbb{P}]_{\max}^{\max})^0(f) = f$ and $(T_{\mathbb{G}_{\max}^{\max}}^{\max})^0(g) = g$.

Next we assume the proposition holds for some $n \in \mathbb{N}$. For any $s=(l, v) \in S_{\max}$, if $l \in F$, then by the construction of the zone graph (see Figure 2, line 3), Definition 3 and Definition 16 we have:

$$(T_{\mathbb{P}]_{\max}^{\max})^{n+1}(f)(s) = 0 = \max \{ (T_{\mathbb{G}_{\max}^{\max}}^{\max})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in Z_{\max} \wedge s \in \mathbf{tpre}(\mathbf{z}) \}.$$

It therefore remains to consider the case when $s=(l, v) \in S_{\max}$ and $l \notin F$. Consider any $(t', a') \in A(s)$ and $(R, l') \in \mathbf{edges}(l, a')$. Now, using Lemma 15, we have $(l', (v+t')[R]) \in S_{\max}$. Therefore, by the induction hypothesis there exists $\mathbf{z}'_{(R, l')} \in Z_{\max}$ with $(l', (v+t')[R]) \in \mathbf{tpre}(\mathbf{z}'_{(R, l')})$ such that:

$$(T_{\mathbb{G}_{\max}^{\max}}^{\max})^n(g)(\mathbf{z}'_{(R, l')})(l', (v+t')[R]) = (T_{\mathbb{P}]_{\max}^{\max})^n(f)(l', (v+t')[R]). \quad (7)$$

Now, since $(t', a') \in A(s)$ and $(l', (v+t')[R]) \in \mathbf{tpre}(\mathbf{z}'_{(R, l')})$, it follows from Definition 8 that $(l, v+t) \in \mathbf{dpre}(l, a', (R, l'))(\mathbf{tpre}(\mathbf{z}'_{(R, l')}))$.

Since the edge $(R, l') \in \mathbf{edges}(l, a')$ was arbitrary, by the construction of the zone graph (see Figure 2, lines 8–19) and Lemma 15, there exists $\mathbf{z}=(l, \zeta) \in Z_{\max}$ such that $v+t' \in \zeta$ and edge set:

$$E' = \{ (\mathbf{z}, a', (R, l'), \mathbf{z}'_{(R, l')}) \mid (R, l') \in \mathbf{edges}(l, a') \} \in \mathbf{E}(\mathbf{z}). \quad (8)$$

Furthermore, by definition of \mathbf{tpre} we have $s \in \mathbf{tpre}(\mathbf{z})$. Using this result and following the same arguments as the proof of Proposition 12 it follows that:

$$(T_{\mathbb{P}]_{\max}^{\max})^{n+1}(f)(s) \leq \max \{ (T_{\mathbb{G}_{\max}^{\max}}^{\max})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in Z_{\max} \wedge s \in \mathbf{tpre}(\mathbf{z}) \}. \quad (9)$$

Next we consider any $\mathbf{z}=(l, \zeta) \in Z_{\max}$ such that $v+t \in \zeta$ for some $t \in \mathbb{R}_+$ (i.e. $\mathbf{z} \in Z_{\max}$ such that $s \in \mathbf{tpre}(\mathbf{z})$). For any $t' \in \mathbb{R}_+$ such that $v+t' \in \zeta$ and $E' \in \mathbf{E}(l, \zeta)$, by construction of the zone graph (see Figure 2, lines 8–19) and Lemma 15 there exists $a' \in \mathbf{Act}$ where:

$$E' = \{ (l, \zeta), a', (R, l'), \mathbf{z}'_{(R, l')} \mid (R, l') \in \mathbf{edges}(l, a') \} \quad (10)$$

and $(l', (v+t')[R]) \in \mathbf{tpre}(\mathbf{z}'_{(R, l')})$ for all $(R, l') \in \mathbf{edges}(l, a')$. Now, again following the arguments of the proof of Proposition 12, we have that:

$$(T_{\mathbb{P}]_{\max}^{\max})^{n+1}(f)(s) \geq \max \{ (T_{\mathbb{G}_{\max}^{\max}}^{\max})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in Z_{\max} \wedge s \in \mathbf{tpre}(\mathbf{z}) \}. \quad (11)$$

Finally, combining (9) and (11) yields:

$$(T_{\mathbb{P}]_{\max}^{\max})^{n+1}(f)(s) = \max \{ (T_{\mathbb{G}_{\max}^{\max}}^{\max})^{n+1}(g)(\mathbf{z})(s) \mid \mathbf{z} \in Z_{\max} \wedge s \in \mathbf{tpre}(\mathbf{z}) \}$$

and hence, since $s \in S_{\max}$ was arbitrary, the proposition holds by induction. \square

4.5. Rational Simple Functions and Rational Nice Functions

In [8], the authors introduce simple functions and show that all value functions encountered during the iterative procedure for computing the optimal time reachability for TAs belong to this special class. For a zone ζ , a function $f : \zeta \rightarrow \mathbb{R}_+$ is *simple* if and only there exists $c_j, d_l \in \mathbb{N}$, $x_l \in \mathcal{X}$, C_j and D_l are zones for $1 \leq j \leq M$, $1 \leq l \leq N$ and some $M, N \in \mathbb{N}$ such that for any $v \in \zeta$:

$$f(v) = \begin{cases} c_j & \text{if } v \in C_j \\ d_l - v(x_l) & \text{if } v \in D_l \end{cases}$$

When it comes to PTAs, due to the presence of probabilistic branching, simple functions are not a sufficient, neither in terms of their domain (i.e. zones) nor the representation. This is demonstrated by the following example.

Example 4. We return to the PTA \mathcal{P}_1 of Example 1 (see Figure 1). Expressing the minimum expected time in the initial location as a function $f : \mathbb{R}_+^{\mathcal{X}} \rightarrow \mathbb{R}_+$ we have:

$$f(v) = \begin{cases} 2.9 & \text{if } x \leq 2.1 \\ 5 - v(x) & \text{if } 2.1 \leq x \leq 5 \\ 0 & \text{if } 5 \leq x \leq 10 \end{cases}$$

and hence it cannot be represented using simple functions. ■

We now introduce *rational simple functions* to represent the functions encountered during value iteration. For the remainder of the section suppose $\mathcal{X} = \{x_1, \dots, x_n\}$ and k is the maximum constant appearing in \mathcal{P} . From Assumption 9(a) we have that \mathcal{P} is bounded, and hence all clock values in \mathcal{P} are bounded by k . We first define polyhedra with rational time bounds and then use these to define rational simple functions.

Definition 18. A (convex) k -polyhedron $C \subseteq \{v \in \mathbb{R}_+^{\mathcal{X}} \mid v(x) \leq k \text{ for } x \in \mathcal{X}\}$ is defined by finitely many linear inequalities; formally, it is of the form:

$$C = \{v \in \mathbb{R}_+^{\mathcal{X}} \mid \sum_{i=1}^n q_{ij} \cdot v(x_i) \leq f_j \text{ for } 1 \leq j \leq M\}$$

where $q_{ij}, f_j \in \mathbb{Q}$ and $f_j \leq k$ for all $1 \leq i \leq n$ and $1 \leq j \leq M$ for some $M \in \mathbb{N}$.

Definition 19. For zone ζ , a function $f : \zeta \rightarrow \mathbb{R}_+$ is rational k -simple if and only if it can be represented as:

$$f(v) = \begin{cases} c_j & \text{if } v \in C_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) & \text{if } v \in D_l \end{cases}$$

where $c_j, d_l, p_{il} \in \mathbb{Q}_+$ such that $\sum_{i=1}^n p_{il} \leq 1$ and C_j, D_l are k -polyhedra for all $1 \leq i \leq n$, $1 \leq j \leq M$ and $1 \leq l \leq N$ for some $M, N \in \mathbb{N}$.

Furthermore, a function $f : \mathbf{Z} \rightarrow (S \rightarrow \mathbb{R}_+)$ is rational k -simple if the function $f(l, \zeta)(l, \cdot) : \zeta \rightarrow \mathbb{R}_+$ is rational k -simple for all $(l, \zeta) \in \mathbf{Z}$.

In the remainder of the section we will prove that all the value functions encountered when computing the optimal expected time reachability using value iteration and either $T_{\mathbf{G}_{\min}}^{\min}$ or $T_{\mathbf{G}_{\max}}^{\max}$ belong to the class of rational simple functions. This is accomplished by considering the different operations performed by $T_{\mathbf{G}_{\min}}^{\min}$ and $T_{\mathbf{G}_{\max}}^{\max}$ (see Definition 11 and Definition 16) and analysing their effect on rational simple functions. First, we consider the operation of resetting the clocks.

Definition 20. *If $f : \zeta \rightarrow \mathbb{R}_+$ is a rational k -simple function and $R \subseteq \mathcal{X}$, let $f[R] : [R]\zeta \rightarrow \mathbb{R}_+$ be the function where $f[R](v) = f(v[R])$ for all $v \in \zeta$.*

The following lemma demonstrates that resetting clocks preserves rational simplicity.

Lemma 21. *If $f : \zeta \rightarrow \mathbb{R}_+$ is rational k -simple and $R \subseteq \mathcal{X}$, then $f[R] : [R]\zeta \rightarrow \mathbb{R}_+$ is rational k -simple.*

PROOF. For any k -polyhedron C and $R \subseteq \mathcal{X}$, let $[R]C$ be the k -polyhedron $\{v \in \mathbb{R}_+^{\mathcal{X}} \mid v[R] \in C \wedge v(x) \leq k \text{ for } x \in \mathcal{X}\}$.

Now consider any $R \subseteq \mathcal{X}$ and rational k -simple function $f : \zeta \rightarrow \mathbb{R}_+$ such that for any $v \in \zeta$:

$$f(v) = \begin{cases} c_j & \text{if } v \in C_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) & \text{if } v \in D_l \end{cases} \quad (12)$$

where $c_j, d_l, p_{il} \in \mathbb{Q}_+$ such that $\sum_{i=1}^n p_{il} \leq 1$ and C_j, D_l are k -polyhedra for all $1 \leq j \leq n$, $1 \leq l \leq M$ and $1 \leq l \leq N$ for some $M, N \in \mathbb{N}$. By Definition 20, for any $v \in [R]\zeta$ we have:

$$\begin{aligned} f[R](v) &= f(v[R]) \\ &= \begin{cases} c_j & \text{if } v[R] \in C_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v[R](x_i) & \text{if } v[R] \in D_l \end{cases} && \text{(by (12))} \\ &= \begin{cases} c_j & \text{if } v \in [R]C_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v[R](x_i) & \text{if } v \in [R]D_l \end{cases} && \text{(by definition of } [R]C) \\ &= \begin{cases} c_j & \text{if } v \in [R]C_j \\ d_l - \sum_{i=1}^n p'_{il} \cdot v(x_i) & \text{if } v \in [R]D_l \end{cases} \end{aligned}$$

where $p'_{il} = 0$ if $x_i \in R$ and $p'_{il} = p_{il}$ otherwise. It therefore follows that $f[R]$ is rational k -simple as required. \square

The next operation performed by $T_{\mathbf{G}_{\min}}^{\min}$ and $T_{\mathbf{G}_{\max}}^{\max}$ yields functions of the form $v \mapsto t + f(l, \zeta)(l, v + t)$. This motivates the introduction of rational k -nice functions, based on Asarin and Maler's k -nice functions [8].

Definition 22. A k -bipolyhedron is a set of the form $\{(v, t) \mid v \in C \wedge v+t \in D\}$ where C and D are k -polyhedra. For a zone ζ , a function $g : (\zeta \times \mathbb{R}_+) \rightarrow \mathbb{R}_+$ is rational k -nice if and only if it can be represented as:

$$g(v, t) = \begin{cases} c_j + t & \text{if } (v, t) \in F_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t & \text{if } (v, t) \in G_l \end{cases}$$

where $c_j, d_l, p_{il} \in \mathbb{Q}_+$ such that $\sum_{i=1}^n p_{il} \leq 1$ and F_j, G_l are rational k -bipolyhedra for all $1 \leq i \leq n$, $1 \leq j \leq M$ and $1 \leq l \leq N$ for some $M, N \in \mathbb{N}$.

Next we show that rational nicety is perserved under taking convex combinations of functions of the form $v \mapsto t + f(l, \zeta)(l, v+t)$.

Lemma 23. A convex combination of rational k -nice functions is rational k -nice.

PROOF. It is sufficient to consider a binary convex combination, as any other convex combination can be rewritten as a sequence of binary convex combinations. Therefore, consider any zone ζ , rationals $r, r' \in \mathbb{Q}_+$ and rational k -nice functions $g, g' : (\zeta \times \mathbb{R}_+) \rightarrow \mathbb{R}_+$ such that $r+r' = 1$ and for any $v \in \zeta$:

$$g(v, t) = \begin{cases} c_j + t & \text{if } (v, t) \in F_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t & \text{if } (v, t) \in G_l \end{cases}$$

$$g'(v, t) = \begin{cases} c'_{j'} + t & \text{if } (v, t) \in F'_{j'} \\ d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t & \text{if } (v, t) \in G'_{l'} \end{cases}$$

where $c_j, d_l, p_{il}, c'_{j'}, d'_{l'}, p'_{il'} \in \mathbb{Q}_+$ such that $\sum_{i=1}^n p_{il} \leq 1$ and $\sum_{i=1}^n p'_{il'} \leq 1$, and $C_j, D_l, C'_{j'}, D'_{l'}$ are k -polyhedra for all $1 \leq i \leq n$, $1 \leq j \leq M$, $1 \leq l \leq N$, $1 \leq j' \leq M'$ and $1 \leq l' \leq N'$ for some $M, M', N, N' \in \mathbb{N}$. Let $h : (\zeta \times \mathbb{R}_+) \rightarrow \mathbb{R}_+$ be the function such that $h(v, t) = r \cdot g(v, t) + r' \cdot g'(v, t)$ for all $(v, t) \in \zeta \times \mathbb{R}_+$. Considering any $(v, t) \in \zeta \times \mathbb{R}_+$ we have the following four cases to consider.

- If $(v, t) \in F_j \cap F'_{j'}$, for some j and j' , then $h(v, t) = r \cdot c_j + r' \cdot c'_{j'}$.
- If $(v, t) \in F_j \cap G'_{l'}$ for some j and l' , then

$$\begin{aligned} h(v, t) &= r \cdot (c_j + t) + r' \cdot \left(d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t \right) \\ &= (r \cdot c_j + r' \cdot d'_{l'}) - \sum_{i=1}^n (r' \cdot p'_{il'}) \cdot v(x_i) + \left(r + r' - \sum_{i=1}^n (r' \cdot p'_{il'}) \right) \cdot t \\ &\quad \text{(rearranging)} \\ &= (r \cdot c_j + r' \cdot d'_{l'}) - \sum_{i=1}^n (r' \cdot p'_{il'}) \cdot v(x_i) + (1 - \sum_{i=1}^n (r' \cdot p'_{il'})) \cdot t \\ &\quad \text{(since } r+r'=1) \end{aligned}$$

and $\sum_{i=1}^n r' \cdot p'_{il'} = r' \cdot (\sum_{i=1}^n p'_{il'}) \leq r' \cdot 1 \leq 1$ since g' is rational k -nice.

- If $(v, t) \in G_l \cap F'_{j'}$, for some l and j' , then similarly to the above using the fact that $r+r'=1$:

$$\begin{aligned} h(v, t) &= r \cdot \left(d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t \right) + r' \cdot (c'_{j'} + t) \\ &= (r \cdot d_l + r' \cdot c'_{j'}) - \sum_{i=1}^n (r \cdot p_{il}) \cdot v(x_i) + (1 - \sum_{i=1}^n (r \cdot p_{il})) \cdot t \end{aligned}$$

and $\sum_{i=1}^n r \cdot p'_{il} \leq 1$ since g is rational k -nice.

- If $(v, t) \in G_l \cap G'_{l'}$, for some l and l' , then again using the fact that $r+r'=1$ we have:

$$\begin{aligned} h(v, t) &= r \cdot \left(d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t \right) \\ &\quad + r' \cdot \left(d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t \right) \\ &= (r \cdot d_l + r' \cdot d'_{l'}) + \sum_{i=1}^n (r \cdot p_{il} + r' \cdot p'_{il'}) \cdot v(x_i) + (1 - \sum_{i=1}^n (r \cdot p_{il} + r' \cdot p'_{il'})) \cdot t \end{aligned}$$

and $\sum_{i=1}^n (r \cdot p_{il} + r' \cdot p'_{il'}) = r \cdot (\sum_{i=1}^n p_{il}) + r' \cdot (\sum_{i=1}^n p'_{il'}) \leq r \cdot 1 + r' \cdot 1 = 1$ since g and g' are rational k -nice.

As these are all the cases to consider and the intersection of k -polyhedra is a k -polyhedron, it follows that h is a rational k -nice function as required. \square

After the convex combination, $T_{\mathbf{G}_{\min}}^{\min}$ and $T_{\mathbf{G}_{\max}}^{\max}$ take a minimum or maximum value respectively, and therefore we show that these operations also preserve k -nicety.

Lemma 24. *The minimum and maximum of rational k -nice functions are rational k -nice.*

PROOF. We prove the case for the minimum of rational k -nice functions; the case for maximum follows similarly. Given rational k -nice functions $g, g' : (\zeta \times \mathbb{R}_+) \rightarrow \mathbb{R}_+$ where for $(v, t) \in \zeta \times \mathbb{R}_+$:

$$\begin{aligned} g(v, t) &= \begin{cases} c_j + t & \text{if } (v, t) \in F_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t & \text{if } (v, t) \in G_l \end{cases} \\ g'(v, t) &= \begin{cases} c'_{j'} + t & \text{if } (v, t) \in F'_{j'} \\ d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t & \text{if } (v, t) \in G'_{l'} \end{cases} \end{aligned}$$

where $c_j, d_l, p_{il}, c'_{j'}, d'_{l'}, p'_{il'} \in \mathbb{Q}_+$ such that $\sum_{i=1}^n p_{il} \leq 1$ and $\sum_{i=1}^n p'_{il'} \leq 1$, and $C_j, D_l, C'_{j'}, D'_{l'}$ are k -polyhedra for all $1 \leq i \leq n, 1 \leq j \leq M, 1 \leq l \leq N, 1 \leq j' \leq M'$ and $1 \leq l' \leq N'$ for some $M, M', N, N' \in \mathbb{N}$. Letting $h = \min\{g, g'\}$ and considering $h(v, t)$ for any $(v, t) \in \zeta \times \mathbb{R}_+$, we have the following four cases to consider.

- If $(v, t) \in F_j \cap F'_{j'}$, for some j and j' , then

$$h(v, t) = \begin{cases} c_j + t & \text{if } (v, t) \in F_j \cap H \\ c'_{j'} + t & \text{if } (v, t) \in F'_{j'} \cap H' \end{cases}$$

where $H = \{(v, t) \in \zeta \times \mathbb{R}_+ \mid c_j + t \leq c'_{j'} + t\} = \{(v, t) \in \zeta \times \mathbb{R}_+ \mid c_j \leq c'_{j'}\}$ and similarly $H' = \{(v, t) \in \zeta \times \mathbb{R}_+ \mid c'_{j'} \leq c_j\}$.

- If $(v, t) \in F_j \cap G'_{l'}$ for some j and l' , then

$$h(v, t) = \begin{cases} c_j + t & \text{if } (v, t) \in F_j \cap H \\ d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t & \text{if } (v, t) \in G'_{l'} \cap H' \end{cases}$$

where

$$\begin{aligned} H &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid c_j + t \leq d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t\} \\ &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid \sum_{i=1}^n p'_{il'} \cdot (v(x_i) + t) \leq d'_{l'} - c_j\} \quad (\text{rearranging}) \\ &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid \sum_{i=1}^n p'_{il'} \cdot (v+t)(x_i) \leq d'_{l'} - c_j\} \\ &\quad (\text{by definition of } v+t) \end{aligned}$$

and similarly $H' = \{(v, t) \in \zeta \times \mathbb{R}_+ \mid \sum_{i=1}^n p'_{il'} \cdot (v+t)(x_i) \leq c_j - d'_{l'}\}$.

- If $(v, t) \in G_l \cap F'_{j'}$ for some l and j' , then

$$h(v, t) = \begin{cases} d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t & \text{if } (v, t) \in G_l \cap H \\ c_{j'} + t & \text{if } (v, t) \in F'_{j'} \cap H' \end{cases}$$

and by a similar reduction to the case above we have:

$$\begin{aligned} H &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid \sum_{i=1}^n p_{il} \cdot (v+t)(x_i) \leq c_{j'} - d_l\} \\ H' &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid \sum_{i=1}^n p_{il} \cdot (v+t)(x_i) \leq d_l - c_{j'}\}. \end{aligned}$$

- If $(v, t) \in G_l \cap G'_{l'}$ for some l and l' , then

$$h(v, t) = \begin{cases} d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t & \text{if } (v, t) \in G_l \cap H \\ d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t & \text{if } (v, t) \in G'_{l'} \cap H' \end{cases}$$

where

$$\begin{aligned} H &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t \\ &\quad \leq d'_{l'} - \sum_{i=1}^n p'_{il'} \cdot v(x_i) + (1 - \sum_{i=1}^n p'_{il'}) \cdot t\} \\ &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid \sum_{i=1}^n (p'_{il'} - p_{il}) \cdot v(x_i) + \sum_{i=1}^n (p'_{il'} - p_{il}) \cdot t \leq d'_{l'} - d_l\} \\ &\quad (\text{rearranging}) \\ &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid -\sum_{i=1}^n (p'_{il'} - p_{il}) \cdot (v(x_i) + t) \leq d'_{l'} - d_l\} \\ &\quad (\text{rearranging again}) \\ &= \{(v, t) \in \zeta \times \mathbb{R}_+ \mid -\sum_{i=1}^n (p'_{il'} - p_{il}) \cdot (v+t)(x_i) \leq d'_{l'} - d_l\} \\ &\quad (\text{by definition of } v+t) \end{aligned}$$

and similarly $H' = \{(v, t) \in \zeta \times \mathbb{R}_+ \mid -\sum_{i=1}^n (p_{il} - p'_{il'}) \cdot (v+t)(x_i) \leq d_l - d'_{l'}\}$.

Since in each case H and H' are k -bipolyhedra, it follows from Definition 22 that the lemma holds. \square

The final operations performed by $T_{G_{\min}}^{\min}$ and $T_{G_{\max}}^{\max}$ concern taking the infimum or supremum over t of a function of the form $v \mapsto t + f(l, \zeta)(l, v+t)$. Hence, we now show that performing either of these operations on a rational nice function returns a rational simple function.

Lemma 25. *For any zone ζ , if $g : (\zeta \times \mathbb{R}_+) \rightarrow \mathbb{R}_+$ is rational k -nice, then the functions $f_1 : \zeta \rightarrow \mathbb{R}_+$ and $f_2 : \zeta \rightarrow \mathbb{R}_+$ where $f_1(v) = \inf_{t \in \mathbb{R}_+} g(v, t)$ and $f_2(v) = \sup_{t \in \mathbb{R}_+} g(v, t)$ for $v \in \zeta$ are rational k -simple.*

PROOF. We consider the case for f_1 ; the case for f_2 follows similarly again using results from [8]. Consider any zone ζ and rational k -nice function $g : (\zeta \times \mathbb{R}_+) \rightarrow \mathbb{R}_+$. By Definition 22, for any $(v, t) \in \zeta \times \mathbb{R}_+$, we have:

$$g(v, t) = \begin{cases} c_j + t & \text{if } (v, t) \in F_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot t & \text{if } (v, t) \in G_l \end{cases}$$

where $c_j, d_l, p_{il} \in \mathbb{Q}_+$ such that $\sum_{i=1}^n p_{il} \leq 1$,

$$F_j = \{(v, t) \mid v \in C_j \wedge v+t \in C'_j\} \quad \text{and} \quad G_l = \{(v, t) \mid v \in D_l \wedge v+t \in D'_l\}$$

for some k -polyhedra C_j, C'_j, D_l and D'_l for all $1 \leq i \leq n, 1 \leq j \leq M$ and $1 \leq l \leq N$ for some $M, N \in \mathbb{N}$.

Now, for any k -polyhedron C , let $\Delta(v, C) \stackrel{\text{def}}{=} \inf\{t \mid v+t \in C\}$, then similarly to [8] we have the function $\Delta(\cdot, C) : \zeta \rightarrow \mathbb{R}_+$ is k -simple over k -polyhedra. If $f_1(v) = \inf_{t \in \mathbb{R}_+} g(v, t)$, since $0 \leq \sum_{i=1}^n p_{il} \leq 1$, for any $v \in \zeta$:

$$f_1(v) = \begin{cases} c_j & \text{if } v \in C_j \cap C'_j \\ c_j + \Delta(v, C'_j) & \text{if } v \in C_j \setminus C'_j \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) & \text{if } v \in D_l \cap D'_l \\ d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot \Delta(v, D'_l) & \text{if } v \in D_l \setminus D'_l. \end{cases}$$

In all except the final case, since $\Delta(\cdot, C) : \zeta \rightarrow \mathbb{R}_+$ is k -simple, it follows that f_1 is rational k -simple. In this final case, by definition of k -simple functions we have the following two cases to consider.

- $\Delta(v, D'_l) = d'_l$ if $v \in D_l \setminus D'_l$ for some $d'_l \in \mathbb{Q}_+$, and therefore for any $v \in D_l \setminus D'_l$:

$$\begin{aligned} f_1(v) &= d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot \Delta(v, D'_l) \\ &= (d_l + (1 - \sum_{i=1}^n p_{il}) \cdot d'_l) - \sum_{i=1}^n p_{il} \cdot v(x_i) \end{aligned}$$

which is rational k -simple, since g is rational k -nice.

- $\Delta(v, D'_l) = d'_l - v(x_{i'_l})$ if $v \in D_l \setminus D'_l$ for some $d'_l \in \mathbb{Q}_+$ and $1 \leq i'_l \leq n$, and hence for any $v \in D_l \setminus D'_l$:

$$\begin{aligned} f_1(v) &= d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot \Delta(v, D'_l) \\ &= d_l - \sum_{i=1}^n p_{il} \cdot v(x_i) + (1 - \sum_{i=1}^n p_{il}) \cdot (d'_l - v(x_{i'_l})) \\ &= (d_l + (1 - \sum_{i=1}^n p_{il}) \cdot d'_l) - \sum_{i=1}^n p'_{il} \cdot v(x_i) \end{aligned}$$

where $p'_{il} = p_{il} + (1 - \sum_{i=1}^n p_{il})$ if $i=i'_l$ and $p'_{il}=p_{il}$ otherwise. Now since g is k -nice we have $p_{il} \in \mathbb{Q}_+$ for all $1 \leq i \leq n$ and $\sum_{i=1}^n p_{il} \leq 1$; it follows that p'_{il} for all $1 \leq i \leq n$ and that:

$$\sum_{i=1}^n p'_{il} = \sum_{i=1}^n p_{il} + 1 - \sum_{i=1}^n p_{il} = 1,$$

and hence f_1 is rational k -simple.

Therefore, we conclude that f_1 is rational k -simple as required. \square

We now combine the above results and show that rational simple functions are a suitable representation for value functions when computing optimal expected time using value iteration and either $T_{\mathbf{G}_{\min}}^{\min}$ or $T_{\mathbf{G}_{\max}}^{\max}$.

Proposition 26. *If $f : \mathbf{Z}_{\min} \rightarrow (S_{\min} \rightarrow \mathbb{R}_+)$ is a rational k -simple function, then $T_{\mathbf{G}_{\min}}^{\min}(f)$ is rational k -simple.*

PROOF. Consider any rational k -simple function, $\mathbf{z} \in \mathbf{Z}_{\min}$ and $E \in \mathbf{E}(\mathbf{z})$. For any $v \in \mathbb{R}_+^X$ and $t \in \mathbb{R}_+$ we have:

$$\begin{aligned} t + \sum_{(\mathbf{z}, a, (R, l'), \mathbf{z}_{(R, l')}) \in E} \mathbf{prob}(l, a)(R, l') \cdot f(\mathbf{z}_{(R, l')})(l', (v+t)[R]) \\ = t + \sum_{(\mathbf{z}, a, (R, l'), \mathbf{z}_{(R, l')}) \in E} \mathbf{prob}(l, a)(R, l') \cdot f[R](\mathbf{z}_{(R, l')})(l', v+t) \\ \hspace{15em} \text{(by Definition 20)} \\ = \sum_{(\mathbf{z}, a, (R, l'), \mathbf{z}_{(R, l')}) \in E} \mathbf{prob}(l, a)(R, l') \cdot (t + f[R](\mathbf{z}_{(R, l')})(l', v+t)) \end{aligned} \quad (13)$$

since $\mathbf{prob}(l, a)$ is a distribution. By construction f is rational k -simple, and hence for any $(\mathbf{z}, a, (R, l'), \mathbf{z}_{(R, l')}) \in E$ using Lemma 21 we have $f[R]$ is also rational k -simple. Therefore, it follows from Definition 22 that:

$$(v, t) \mapsto t + f[R](\mathbf{z}_{(R, l')})(l', v+t)$$

is rational k -nice. Thus, since $(\mathbf{z}, a, (R, l'), \mathbf{z}_{(R, l')}) \in E$ was arbitrary, using Lemma 23 and (13) we have that:

$$(v, t) \mapsto t + \sum_{(\mathbf{z}, a, (R, l'), \mathbf{z}_{(R, l')}) \in E} \mathbf{prob}(l, a)(R, l') \cdot f(\mathbf{z}_{(R, l')})(l', (v+t)[R])$$

is also rational k -nice. Since $E \in \mathbf{E}(\mathbf{z})$ was arbitrary and $\mathbf{E}(\mathbf{z})$ is finite, Lemma 24 tells us:

$$(v, t) \mapsto \min_{E \in \mathbf{E}(\mathbf{z})} \left\{ t + \sum_{(\mathbf{z}, a, (R, l'), \mathbf{z}_{(R, l')}) \in E} \mathbf{prob}(l, a)(R, l') \cdot f(\mathbf{z}_{(R, l')})(l', (v+t)[R]) \right\}$$

is again rational k -nice. Finally, using Definition 11 and Lemma 25, it follows that $T_{\mathbf{G}}(f)(\mathbf{z})$ is rational k -simple as required. \square

Proposition 27. *If $f : \mathbf{Z}_{\max} \rightarrow (S_{\max} \rightarrow \mathbb{R}_+)$ is a rational k -simple function, then $T_{\mathbf{G}_{\max}}^{\max}(f)$ is rational k -simple.*

PROOF. The proof follows similarly to Proposition 26. \square

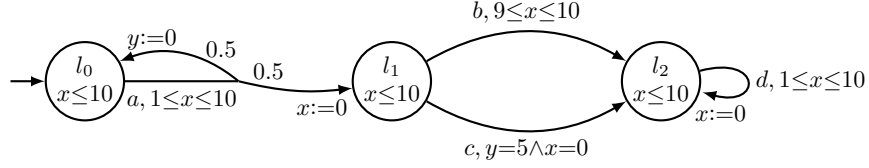


Figure 9: Example PTA \mathcal{P}_4

4.6. Controller Synthesis

We now give our approach for computing the optimal expected time of reaching the target F in the PTA \mathcal{P} and synthesising an ε -optimal strategy when starting from the initial state.

In the case of minimum expected time, we first build the backwards zone graph \mathbf{G} (see Figure 2), then, using Theorem 10 and graph-based algorithms [38], we can find the states of $\llbracket \mathcal{P} \rrbracket$ for which the maximum probability of reaching the target S_F equals 1 and remove these from the zone graph. Next, using Definition 11, we apply value iteration to the resulting zone graph \mathbf{G}_{\min} which, by Proposition 26, can be performed using rational k -simple functions (and rational k -nice functions). Convergence to the minimum expected reachability values of \mathcal{P} is guaranteed by Theorem 5 and Proposition 12. An ε -optimal deterministic, memoryless strategy can be synthesised once value iteration has converged by starting from the initial state and stepping through the backwards graph, in each state choosing the time and action that achieve the values returned by value iteration.

In the case of maximum expected time, we again first build the backwards zone graph \mathbf{G} . However, we now use the algorithms presented in Figure 5 and Figure 6 to find the states of $\llbracket \mathcal{P} \rrbracket$ for which the minimum probability of reaching the target is less than 1 and remove these from the zone graph. Using Definition 16, we then apply value iteration to the resulting zone graph \mathbf{G}_{\max} which, by Proposition 27, can be performed using rational k -simple functions (and rational k -nice functions). Convergence to the maximum expected reachability values of \mathcal{P} is guaranteed by Corollary 6 and Proposition 17. An ε -optimal deterministic, memoryless strategy can then be synthesised in the same manner as above.

Example 5. The PTA \mathcal{P}_4 in Figure 9 presents an example, where waiting longer than necessary in a location can reduce the time to reach the target. The target set is $\{l_2\}$ and the zone graph \mathbf{G} is given in Figure 10. For this example all states of the PTA can reach the target with maximum probability 1, and hence we find that $\mathbf{G}_{\min} = \mathbf{G}$. Starting from the constant 0 function f_0 and performing

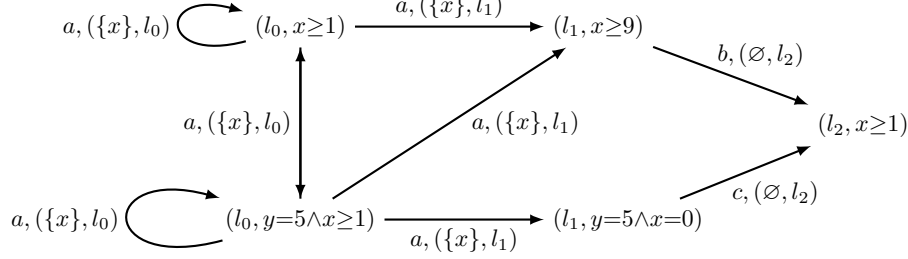


Figure 10: Backwards zone graph for PTA \mathcal{P}_4 and target set $\{l_2\}$

value iteration gives for $n \geq 2$:

$$\begin{aligned}
(T_{\mathbb{G}_{\min}}^{\min})^n(f_0)(\mathbf{z}_0^1)(l_0, v) &= \begin{cases} (1-v(x)) + \sum_{i=1}^{n-1} 0.5^n \cdot 9 & \text{if } v(x) \leq 1 \\ \sum_{i=1}^n 0.5^{n-1} \cdot 9 & \text{if } 1 \leq v(x) \leq 10 \end{cases} \\
(T_{\mathbb{G}_{\min}}^{\min})^n(f_0)(\mathbf{z}_0^2)(l_0, v) &= \begin{cases} (5-v(y)) + 0.5 \cdot (\sum_{i=1}^n 0.5^{n-1} \cdot 9) & \text{if } v(y) \leq 5 \\ 0.5 \cdot (\sum_{i=1}^{n-1} 0.5^{n-1} \cdot 9) & \text{if } 5 \leq v(y) \leq 10 \end{cases} \\
(T_{\mathbb{G}_{\min}}^{\min})^n(f_0)(\mathbf{z}_1^1)(l_1, v) &= \begin{cases} 9-v(x) & \text{if } v(x) \leq 9 \\ 0 & \text{if } 9 \leq v(x) \leq 10 \end{cases} \\
(T_{\mathbb{G}_{\min}}^{\min})^n(f_0)(\mathbf{z}_1^2) &= 0 \\
(T_{\mathbb{G}_{\min}}^{\min})^n(f_0)(\mathbf{z}_2) &= 0
\end{aligned}$$

where $\mathbf{z}_0^1 = (l_0, x \geq 1)$, $\mathbf{z}_0^2 = (l_0, y = 5 \wedge x \geq 1)$, $\mathbf{z}_1^1 = (l_1, x \geq 9)$, $\mathbf{z}_1^2 = (l_1, y = 5 \wedge x = 0)$ and $\mathbf{z}_2 = (l_2, x \geq 1)$. Therefore, value iteration converges to:

$$\begin{aligned}
f_{\min}(\mathbf{z}_0^1)(l_0, v) &= \begin{cases} (1-v(x)) + 9 & \text{if } v(x) \leq 1 \\ 9 & \text{if } 1 \leq v(x) \leq 10 \end{cases} \\
f_{\min}(\mathbf{z}_0^2)(l_0, v) &= \begin{cases} (5-v(y)) + 0.5 \cdot 9 & \text{if } v(y) \leq 5 \\ 0.5 \cdot 9 & \text{if } 5 \leq v(y) \leq 10 \end{cases}
\end{aligned}$$

and hence the minimum expected time for the initial state equals the minimum of $(1-0)+9$ and $(5-0)+0.5 \cdot 9$, yielding 9.5. Performing controller synthesis we find that this corresponds to waiting until $y=5$, then performing the action a . If l_1 is reached, we immediately perform the action c and reach the target. On the other hand, if l_0 is reached, we repeatedly immediately perform a and, if l_1 is reached, wait until $x=9$ and then perform the action b reaching the target. ■

5. Conclusions

We have proposed symbolic algorithms for PTAs to compute the minimum and maximum expected time to reach a target and synthesise the corresponding strategies. The algorithms are formulated as value iteration over the backwards

zone graph of the PTA. We also demonstrate that there is an effective representation of the value functions in terms of rational simple and rational nice functions. However, zones are not sufficient and convex polyhedra are required. Nevertheless, the Parma Polyhedra Library [39] offers efficient ways to manipulate convex polyhedra and is commonly used in a variety of real-time verification problems. For example, methods based on priced zones for TAs and PTAs, such as [17] and [10], also use convex polyhedra, where similarly zones do not suffice.

An interesting topic for further study is a rigorous comparison with the digital clocks method [12]. As Example 4 demonstrates, we require rational bounds on polyhedra. This at first appears to contradict the digital clocks result, which demonstrates that integer bounds are sufficient for computing optimal expected time reachability values. However, it can be explained by the fact that in the digital clocks approach one restricts to the states of the PTA where all clocks take integer values and then computes expected values *individually*, while here we consider all states of the PTA and compute expected values *collectively* through rational simple and nice functions. More precisely, when employing the digital clocks approach, we build an MDP where the states of the MDP correspond to the states of the PTA where all clocks take integer values and compute an optimal expected reachability value for each state of the MDP which equals the optimal expected time value of the corresponding PTA state. On the other hand, here we build a zone graph where the (symbolic) states of the graph are location zone pairs representing (uncountable) sets of states of the PTA and then compute rational simple (and nice) functions over polyhedra, which represent the optimal expected time values for all the states of the PTA (for which the optimal expected value is finite). The fact that we consider all states introduces rational bounds which may yield inefficiencies, but this can potentially be outweighed by efficiency gains that may arise from computing these values collectively. For instance, in Example 5 using the approach of this paper rational bounds are introduced, but only eight separate location-polyhedron pairs are required for computing the optimal expected time values for all states. For comparison, using the digital clocks method, one is required to compute optimal expected values for 101 individual states. It is therefore unclear which approach will perform better in practice and a detailed evaluation of the efficiency of the approach is desirable.

Regarding future work, as well as working on an implementation, we note that optimisations to the backwards algorithm presented in [19], including first performing forwards reachability to restrict analysis to the reachable state space, could be considered here as well. Since policy iteration also converges (see Theorem 5 and Corollary 6), we plan to investigate this approach and compare with value iteration. In addition, we intend to consider linearly-priced PTAs and expected price reachability, which will require an extension of rational nice functions to encode the accumulation of prices as time passes.

Acknowledgements. This research is supported by ERC AdG-246967 VERIWARE. Part of this work was completed during a research internship at the University of Glasgow for Quentin Peyras as part of a Master of Research in

Computer Science at ENS Cachan. We also thank the anonymous referees for their helpful comments to improve the paper.

References

- [1] R. Alur, D. Dill, A theory of timed automata, *Theoretical Computer Science* 126 (1994) 183–235.
- [2] H. Gregersen, H. Jensen, Formal design of reliable real time systems, Master’s thesis, Department of Mathematics and Computer Science, Aalborg University (1995).
- [3] M. Kwiatkowska, G. Norman, R. Segala, J. Sproston, Automatic verification of real-time systems with discrete probability distributions, *Theoretical Computer Science* 282 (2002) 101–150.
- [4] D. Beauquier, On probabilistic timed automata, *Theoretical Computer Science* 292 (1) (2003) 65–84.
- [5] M. Kwiatkowska, G. Norman, D. Parker, Stochastic games for verification of probabilistic timed automata, in: J. Ouaknine, F. Vaandrager (Eds.), *Proc. 7th Int. Conf. Formal Modelling and Analysis of Timed Systems (FORMATS’09)*, Vol. 5813 of LNCS, Springer, 2009, pp. 212–227.
- [6] M. Kwiatkowska, G. Norman, D. Parker, PRISM 4.0: Verification of probabilistic real-time systems, in: G. Gopalakrishnan, S. Qadeer (Eds.), *Proc. 23rd Int. Conf. Computer Aided Verification (CAV’11)*, Vol. 6806 of LNCS, Springer, 2011, pp. 585–591.
- [7] M. Kwiatkowska, G. Norman, J. Sproston, F. Wang, Symbolic model checking for probabilistic timed automata, *Information and Computation* 205 (7) (2007) 1027–1077.
- [8] E. Asarin, O. Maler, As soon as possible: Time optimal control for timed automata, in: F. Vaandrager, J. van Schuppen (Eds.), *Proc. 2nd Int. Workshop Hybrid Systems: Computation and Control (HSCC’99)*, Vol. 1569 of LNCS, Springer, 1999, pp. 19–30.
- [9] G. Behrmann, A. Fehnker, T. Hune, K. Larsen, P. Pettersson, J. Romijn, F. Vaandrager, Minimum-cost reachability for priced timed automata, in: M. Di Benedetto, A. Sangiovanni-Vincentelli (Eds.), *Proc. 4th Int. Workshop Hybrid Systems: Computation and Control (HSCC’01)*, Vol. 2034 of LNCS, Springer, 2001, pp. 147–161.
- [10] K. Larsen, G. Berhmann, E. Brinksma, A. Fehnker, T. Hune, P. Pettersson, J. Romijn, As cheap as possible: Efficient cost-optimal reachability for priced timed automata, in: G. Berry, H. Comon, A. Finkel (Eds.), *Proc. 14th Int. Conf. Computer Aided Verification (CAV’02)*, Vol. 2102 of LNCS, Springer, 2001, pp. 493–505.

- [11] K. Larsen, P. Pettersson, W. Yi, UPPAAL in a Nutshell, *International Journal on Software Tools for Technology Transfer* 1 (1997) 134–152.
- [12] M. Kwiatkowska, G. Norman, D. Parker, J. Sproston, Performance analysis of probabilistic timed automata using digital clocks, *Formal Methods in System Design* 29 (2006) 33–78.
- [13] M. Dufflot, M. Kwiatkowska, G. Norman, D. Parker, A formal analysis of Bluetooth device discovery, *Int. Journal on Software Tools for Technology Transfer* 8 (6) (2006) 621–632.
- [14] G. Norman, D. Parker, X. Zou, Verification and control of partially observable probabilistic real-time systems, in: S. Sankaranarayanan, E. Vicario (Eds.), *Proc. 13th Int. Conf. Formal Modelling and Analysis of Timed Systems (FORMATS'15)*, Vol. 9268 of LNCS, Springer, 2015, pp. 240–255.
- [15] A. David, P. Jensen, K. Larsen, A. Legay, D. Lime, M. Sørensen, J. Taankvist, On time with minimal expected cost!, in: F. Cassez, J. Raskin (Eds.), *Proc. 12th Int. Symp. Automated Technology for Verification and Analysis (ATVA'14)*, Vol. 8837 of LNCS, Springer, 2014, pp. 129–145.
- [16] P. Bulychev, A. David, K. Larsen, M. Mikučionis, D. Poulsen, A. Legay, Z. Wang, UPPAAL-SMC: Statistical model checking for priced timed automata, in: *Proc. 10th Workshop Quantitative Aspects of Programming Languages (QAPL'12)*, Vol. 85 of EPTCS, Open Publishing Association, 2012, pp. 1–16.
- [17] J. Berendsen, D. Jansen, J.-P. Katoen, Probably on time and within budget – On reachability in priced probabilistic timed automata, in: *Proc. 3rd Int. Conf. Quantitative Evaluation of Systems (QEST'06)*, IEEE Press, 2006, pp. 311–322.
- [18] J. Berendsen, T. Chen, D. Jansen, Undecidability of cost-bounded reachability in priced probabilistic timed automata, in: J. Chen, S. Cooper (Eds.), *Proc. 6th Int. Conf. Theory and Applications of Models of Computation (TAMC'09)*, Vol. 5532 of LNCS, Springer, 2009, pp. 128–137.
- [19] J. Berendsen, D. Jansen, F. Vaandrager, Fortuna: Model checking priced probabilistic timed automata, in: *Proc. 7th Int. Conf. Quantitative Evaluation of Systems (QEST'10)*, IEEE Press, 2010, pp. 273–281.
- [20] A. Jovanovic, M. Kwiatkowska, G. Norman, Symbolic minimum expected time controller synthesis for probabilistic timed automata, in: S. Sankaranarayanan, E. Vicario (Eds.), *Proc. 13th Int. Conf. Formal Modeling and Analysis of Timed Systems (FORMATS'15)*, Vol. 9268 of LNCS, Springer, 2015, pp. 140–155.
- [21] J. Kemeny, J. Snell, A. Knapp, *Denumerable Markov Chains*, Springer, 1976.

- [22] V. Forejt, M. Kwiatkowska, G. Norman, D. Parker, Automated verification techniques for probabilistic systems, in: M. Bernardo, V. Issarny (Eds.), *Formal Methods for Eternal Networked Software Systems (SFM'11)*, Vol. 6659 of LNCS, Springer, 2011, pp. 53–113.
- [23] R. Bellman, *Dynamic Programming*, Princeton University Press, 1957.
- [24] D. Bertsekas, J. Tsitsiklis, An analysis of stochastic shortest path problems, *Mathematics of Operations Research* 16 (3) (1991) 580–595.
- [25] D. Bertsekas, *Dynamic Programming and Optimal Control*, Volumes 1 and 2, Athena Scientific, 1995.
- [26] S. Haddad, B. Monmege, Reachability in MDPs: Refining convergence of value iteration, in: J. Ouaknine, I. Potapov, J. Worrell (Eds.), *Proc. 8th Int. Workshop Reachability Problems (RP'14)*, Vol. 8762, Springer, 2014, pp. 125–137.
- [27] H. James, E. Collins, An analysis of transient Markov decision processes, *Journal of Applied Probability* 43 (3) (2006) 603–621.
- [28] T. Henzinger, X. Nicollin, J. Sifakis, S. Yovine, Symbolic model checking for real-time systems, *Information and Computation* 111 (2) (1994) 193–244.
- [29] S. Tripakis, *The analysis of timed systems in practice*, Ph.D. thesis, Université Joseph Fourier, Grenoble (1998).
- [30] G. Behrmann, A. Fehnker, T. Hune, K. Larsen, P. Pettersson, J. Romijn, F. Vaandrager, Minimum-cost reachability for linearly priced timed automata, in: M. Di Benedetto, A. Sangiovanni-Vincentelli (Eds.), *Proc. 4th Int. Workshop Hybrid Systems: Computation and Control (HSCC'01)*, Vol. 2034 of LNCS, Springer, 2001, pp. 147–162.
- [31] V. Forejt, M. Kwiatkowska, G. Norman, A. Trivedi, Expected reachability-time games, *Theoretical Computer Science* 631 (2016) 139–160.
- [32] M. Jurdziński, M. Kwiatkowska, G. Norman, A. Trivedi, Concavely-priced probabilistic timed automata, in: M. Bravetti, G. Zavattaro (Eds.), *Proc. 20th Int. Conf. Concurrency Theory (CONCUR'09)*, Vol. 5710 of LNCS, Springer, 2009, pp. 415–430.
- [33] S. Tripakis, Verifying progress in timed systems, in: J.-P. Katoen (Ed.), *Proc. 5th Int. AMAST Workshop Real-Time and Probabilistic Systems (ARTS'99)*, Vol. 1601 of LNCS, Springer, 1999, pp. 299–314.
- [34] S. Tripakis, S. Yovine, A. Bouajjan, Checking timed Büchi automata emptiness efficiently, *Formal Methods in System Design* 26 (3) (2005) 267–292.
- [35] J. Ouaknine, J. Worrell, Revisiting digitization, robustness, and decidability for timed automata, in: *Proc. 18th Annual IEEE Symp. Logic in Computer Science (LICS'03)*, IEEE Press, 2003, pp. 198–207.

- [36] R. Alur, C. Courcoubetis, D. Dill, Model checking in dense real time, *Information and Computation* 104 (1) (1993) 2–34.
- [37] A. Bianco, L. de Alfaro, Model checking of probabilistic and nondeterministic systems, in: P. Thiagarajan (Ed.), *Proc. 15th Conf. Foundations of Software Technology and Theoretical Computer Science (FSTTCS'95)*, Vol. 1026 of LNCS, Springer, 1995, pp. 499–513.
- [38] L. de Alfaro, Computing minimum and maximum reachability times in probabilistic systems, in: J. Baeten, S. Mauw (Eds.), *Proc. 10th Int. Conf. Concurrency Theory (CONCUR'99)*, Vol. 1664 of LNCS, Springer, 1999, pp. 66–81.
- [39] R. Bagnara, P. Hill, E. Zaffanella, The Parma Polyhedra Library: Toward a complete set of numerical abstractions for the analysis and verification of hardware and software systems, *Science of Computer Programming* 72 (1–2) (2008) 3–21.