Symbolic Model Checking for Probabilistic Timed Automata*

Marta Kwiatkowska[1], Gethin Norman[1] and Jeremy Sproston[2]
[1] School of Computer Science, University of Birmingham, Edgbaston,
Birmingham B15 2TT, United Kingdom
[2] Dipartimento di Informatica, Università di Torino, 10149 Torino, Italy

October 24, 2003

Abstract

Probabilistic timed automata are an extension of timed automata with discrete probability distributions, and can be used to model timed randomized protocols or fault-tolerant systems. We present symbolic model checking algorithms for probabilistic timed automata to verify qualitative properties, corresponding to satisfaction with probability 0 or 1, as well as quantitative properties, corresponding to satisfaction with arbitrary probability. The algorithms operate on zones, that is, sets of valuations of the probabilistic timed automaton's clocks, and therefore avoid an explicit construction of the state space. Our method considers only those system behaviours which guarantee the divergence of time with probability 1. The paper completes the symbolic framework for the verification of probabilistic timed automata against full PTCTL. We formulate new algorithms that can return the minimal probability with which a probabilistic timed automaton satisfies a property, thus extending a previously published result concerning the maximum probability.

1 Introduction

Systems exhibiting both *timed* and *probabilistic* characteristics are widespread, in application contexts as diverse as home entertainment, medical equipment and business infrastructures. For example, timing constraints are often vital to the correctness of embedded digital technology, whereas probability exhibits itself commonly in the form of statistical estimates regarding the environment in which a system is embedded. Similarly, protocol designers often exploit the combination of time and probability to design correct, efficient protocols, such as the IEEE1394 FireWire root contention protocol. The diffusion of such systems has led to methods for obtaining formal correctness guarantees, for instance, adaptations of model checking [CGP99]. *Symbolic model checking* refers to model-checking techniques in which implicit representations – such as BDDs in the finite-state case [BCM+90] – are used to represent both the transition relation of the system model and the state sets that are computed during the execution of the model-checking algorithm.

In this paper, we consider the modelling formalism of *probabilistic timed automata* [KNSS02], an extension of timed automata [AD94, HNSY94] with discrete probability distributions.

*Supported in part by the EPSRC grant GR/N22960, FORWARD and MIUR-FIRB Perf.

Technical Report CSR-03-10, School of Computer Science, University of Birmingham, October 2003.

Probabilistic timed automata have been shown as being suitable for the description of timed, randomized protocols, such as the aforementioned FireWire protocol [KNS03], the backoff strategy of the IEEE802.11 WLAN protocol [KNS02], and the link-local address selection protocol of the IPv4 standard [KNPS03]. As a requirement specification language for probabilistic timed automata we consider PTCTL (Probabilistic Timed Computation Tree Logic). The logic PTCTL combines the probabilistic threshold operator of the probabilistic temporal logic PCTL [HJ94] with the timing constraints of the timed temporal logic TCTL [ACD93, HNSY94], in order to express properties such as 'with probability 0.99 or greater, the system reaches a leader-elected state within 1 second'. Model checking of probabilistic timed automata against PTCTL was shown to be decidable in [KNSS02] via an adaptation of the classical region-graph construction [AD94, ACD93].

Unfortunately, the region-graph construction (and the integer-time semantics employed in [KNS03, KNS02, KNPS03]) can result in huge state spaces if the maximal constant used in the description of the automaton is large. Instead, the practical success of *symbolic*, *zone*-based techniques for non-probabilistic timed automata [BDL+01, DOTY96], suggests that a similar symbolic approach may also be employed for the verification of probabilistic timed automata. This hypothesis was answered affirmatively in [KNS01] for a subset of PTCTL with thresholds on maximal reachability probabilities. In this paper, we extend that result to arbitrary PTCTL formulae. In particular, a zone-based method for verification of properties which refer to the *minimum* probability of satisfaction is presented for the first time.

The technical contribution of this paper is the introduction of zone-based algorithms, both for the verification of *qualitative* PTCTL formulae, which refer to probabilistic thresholds 0 and 1 only, and *quantitative* PTCTL formulae, which feature thresholds on arbitrary probabilities. Note that the qualitative algorithms do not refer to exact transition probabilities, and therefore avoid potentially expensive computation of probabilities during the model-checking process.

We first consider the subset of PTCTL which requires the computation of maximal probabilities. For qualitative formulae, we show that model checking can be performed using analogues from the verification of finite-state probabilistic systems [dA97], while, in the quantitative case, we show that the previously published zone-based approach for calculating maximal probabilities [KNS01] can be employed. The quantitative algorithm works by constructing a finite-state system which has sufficient information to compute the maximum probability of interest using well-established finite-state model checking methods [BdA95].

Secondly, we consider algorithms for the subset of PTCTL which requires the computation of minimum reachability probabilities, a task which is more involved than computing maximum probabilities. For example, to compute the minimum probability of reaching a certain state set F, for any state other than those in F, the probabilistic timed automaton could exhibit behaviour in which the amount of time elapsed converges before F is reached, or even in which no time elapses at all. Clearly, such behaviours are pathological, and should be disregarded during model checking. We present both qualitative and quantitative algorithms for computing minimum reachability probabilities which consider only time-divergent behaviour, based on the non-probabilistic precedent of [HNSY94]. The algorithms are based on computing maximum probabilities for the dual formula while restricting attention to time-divergent behaviours.

Finally, again following the precedent of [HNSY94], we present an algorithm to check that a probabilistic timed automaton does not contain a state in which it is impossible for time to diverge with probability 1. The presence of such a state constitutes a modelling error, and

would invalidate the correctness of our model checking procedure.

2 Preliminaries

2.1 Distributions and Probabilistic Systems

A (discrete probability) *distribution* over a finite set Q is a function $\mu : Q \to [0,1]$ such that $\sum_{q \in Q} \mu(q) = 1$. Let $\mathsf{support}(\mu)$ be the subset of Q such that $q \in \mathsf{support}(\mu)$ if and only if $\mu(q) > 0$. For a possibly uncountable set Q', let $\mathsf{Dist}(Q')$ be the set of distributions over finite subsets of Q'. For any $q \in Q$, the point distribution μ_q denotes the distribution which assigns probability 1 to q.

2.2 Discrete Time Markov Chains

Definition 1 *A* DTMC *is a tuple* $\mathsf{DTMC} = (S, \mathbf{P}, \mathcal{L})$ *where:*

- S *is a finite set of* states*;*
- $\mathbf{P} : S \times S \to [0,1]$ *is a* transition probability matrix*, such that:* $\sum_{s' \in S} \mathbf{P}(s,s') = 1$ *for all states* $s \in S$*;*
- $\mathcal{L} : S \to 2^{AP}$ *is a* labelling function *assigning atomic propositions to states.*

Each element $\mathbf{P}(s,s')$ of the transition probability matrix gives the probability of making a transition from state s to state s'. An execution of a DTMC is represented by a *path* ω, that is, a non-empty sequence of states $s_0 s_1 s_2 \ldots$ where $s_i \in S$ and $\mathbf{P}(s_i, s_{i+1}) > 0$ for all $i \geqslant 0$. We denote by $\omega(i)$ the ith state of a path ω, $|\omega|$ the length of ω and if ω is finite, the last state by $last(\omega)$. We say that a finite path ω_{fin} of length n is a *prefix* of an infinite path ω if $\omega_{fin}(i) = \omega(i)$ for $0 \leqslant i \leqslant n$. The sets of all finite and infinite paths starting in state s are denoted $Path_{ful}(s)$ and $Path_{fin}(s)$, respectively.

In reason about the probabilistic behaviour of the DTMC, we need to determine the probability that certain paths are taken. This is achieved by defining, for each state $s \in S$, a probability measure $Prob_s$ over $Path_{ful}(s)$. Below, we give an outline of this construction. For further details, see [KSK76]. The probability measure is induced by the transition probability matrix $\mathbf{P}$ as follows. First, for any finite path $\omega_{fin} \in Path_{fin}(s)$, we define the probability $\mathbf{P}_s(\omega_{fin})$:

$$\mathbf{P}_s(\omega_{fin}) \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } n = 0 \\ \mathbf{P}(\omega_{fin}(0), \omega_{fin}(1)) \cdots \mathbf{P}(\omega_{fin}(n-1), \omega_{fin}(n)) & \text{otherwise} \end{cases}$$

where $n = |\omega_{fin}|$. Next, we define the *cylinder* of a finite path ω_{fin} as:

$$C(\omega_{fin}) \stackrel{\text{def}}{=} \{\omega \in Path_{ful}(s) \,|\, \omega_{fin} \text{ is a prefix of } \omega\},$$

and let Σ_s be the smallest σ-algebra on $Path_{ful}(s)$ which contains the cylinders $C(\omega_{fin})$ for $\omega_{fin} \in Path_{fin}(s)\}$ and set $Prob_s$ on Σ_s to be the unique measure such that

$$Prob_s(C(\omega_{fin})) = \mathbf{P}_s(\omega_{fin}) \text{ for all } \omega_{fin} \in Path_{fin}(s).$$

2.3 Probabilistic Systems

We next recall probabilistic systems which are essentially equivalent to Markov decision processes [Der70] and probabilistic-nondeterministic systems [BdA95].

Definition 2 *A* probabilistic system, PS, *is a tuple* $(S, \mathit{Steps}, \mathcal{L})$ *where*

- S *is a set of* states*;*
- $\mathit{Steps} \subseteq S \times \mathsf{Dist}(S)$ *is a* probabilistic transition relation*;*
- $\mathcal{L} : S \to 2^{AP}$ *is a* labelling function *assigning atomic propositions to states.*

A *probabilistic transition* $s \xrightarrow{\mu} s'$ is made from a state s by nondeterministically selecting a distribution $\mu \in \mathsf{Dist}(S)$ such that $(s, \mu) \in \mathit{Steps}$, and then making a probabilistic choice of target state s' according to μ, such that $\mu(s')>0$.

We consider two ways in which a probabilistic system's computation may be represented. A *path*, representing a particular resolution of both nondeterminism *and* probability, is a non-empty sequence of transitions:

$$\omega = s_0 \xrightarrow{\mu_0} s_1 \xrightarrow{\mu_1} s_2 \xrightarrow{\mu_2} \cdots .$$

We use the same notation for paths as presented in Section 2.2, in particular, the set of infinite (respectively, finite) paths starting in the state s are denoted by $\mathit{Path}_{\mathit{ful}}(s)$ (respectively, $\mathit{Path}_{\mathit{fin}}(s)$).

In contrast to a path, an *adversary* represents a particular resolution of nondeterminism *only*. Formally, an adversary A is a function mapping every finite path ω_{fin} to a distribution μ such that $(\mathit{last}(\omega_{\mathit{fin}}), \mu) \in \mathit{Steps}$. For any adversary A and state s, we let $\mathit{Path}^A_{\mathit{ful}}(s)$ (respectively, $\mathit{Path}^A_{\mathit{fin}}(s)$) denotes the subset of $\mathit{Path}_{\mathit{ful}}(s)$ (respectively, $\mathit{Path}_{\mathit{fin}}(s)$) which corresponds to A and, using classical techniques [KSK76], we can define the probability measure Prob^A_s over $\mathit{Path}^A_{\mathit{ful}}(s)$.

For a given adversary A and finite path ω, we define a new adversary A_ω as follows:

$$A_\omega(\omega') \stackrel{\text{def}}{=} \begin{cases} A(\omega \xrightarrow{\mu} \omega'') & \text{if } \omega' \text{ is of the form } \mathit{last}(\omega) \xrightarrow{\mu} \omega'' \\ A(\omega') & \text{otherwise.} \end{cases}$$

Whenever possible, the adversary A_ω acts essentially as A assuming that the path ω has already taken place.

For a probabilistic system $\mathsf{PS} = (S, \mathit{Steps}, \mathcal{L})$ and state $s \in S$, under a given adversary A, the behaviour from state s can be described with the infinite-state DTMC $\mathsf{DTMC}^A = (S^A, \mathbf{P}^A)$ where: $S^A = \mathit{Path}_{\mathit{fin}}(s)$ and for two finite paths $\omega_{\mathit{fin}}, \omega'_{\mathit{fin}} \in S^A$:

$$\mathbf{P}^A(\omega_{\mathit{fin}}, \omega'_{\mathit{fin}}) = \begin{cases} \mu(s') & \text{if } \omega'_{\mathit{fin}} \text{ is of the form } \omega_{\mathit{fin}} \xrightarrow{A(\omega_{\mathit{fin}})} s' \text{ and } A(\omega) = \mu \\ 0 & \text{otherwise.} \end{cases}$$

There is a one-to-one correspondence between the paths of DTMC^A and the set of paths $\mathit{Path}^A_{\mathit{ful}}(s)$, and hence using the construction given in Section 2.2 we can define a probability measure Prob^A_s over $\mathit{Path}^A_{\mathit{ful}}(s)$.

For a probabilistic system $\mathsf{PS} = (S, \mathit{Steps}, \mathcal{L})$, state $s \in S$, set $F \subseteq S$ of target states, and adversary $A \in \mathit{Adv}_{\mathsf{PS}}$, let:

$$\mathit{ProbReach}^A(s, F) \stackrel{\text{def}}{=} \mathit{Prob}_s^A\{\omega \in \mathit{Path}_{ful}^A(s) \mid \exists i \in \mathbb{N}.\, \omega(i) \in F\}\,,$$

then the *maximal reachability probabilities* of reaching the set of states F from s is defined as:

$$\mathit{MaxProbReach}(s, F) \stackrel{\text{def}}{=} \sup_{A \in \mathit{Adv}_{\mathsf{PS}}} \mathit{ProbReach}^A(s, F)\,.$$

2.4 Timed Probabilistic Systems

We now introduce *timed probabilistic systems*, an extension of probabilistic systems and a variant of Segala's probabilistic timed automata [Seg95].

Definition 3 *A* timed probabilistic system, TPS, *is a tuple* $(S, \mathit{Steps}, \mathcal{L})$ *where:*

- S *is a set of* states*;*
- $\mathit{Steps} \subseteq S \times \mathbb{R} \times \mathsf{Dist}(S)$ *is a* timed probabilistic transition relation*, such that, if* $(s, t, \mu) \in \mathit{Steps}$ *and* $t{>}0$*, then* μ *is a point distribution;*
- $\mathcal{L} : S \to 2^{AP}$ *is a labelling function.*

The component t of a tuple (s, t, μ) is called a *duration*. As for probabilistic systems, we can introduce paths and adversaries for timed probabilistic systems, except transitions are now labelled by duration-distribution pairs and an adversary maps each finite path to a duration-distribution pair.

We restrict attention to *time-divergent adversaries*; a common restriction imposed in real-time systems so that unrealisable behaviour (i.e. corresponding to time not advancing beyond a bound) is disregarded during analysis. For any path

$$\omega = s_0 \xrightarrow{t_0,\mu_0} s_1 \xrightarrow{t_1,\mu_1} s_2 \xrightarrow{t_2,\mu_2} \cdots$$

of a timed probabilistic system, the duration up to the $n{+}1$th state of ω, denoted $\mathcal{D}_\omega(n{+}1)$, equals $\sum_{i=0}^{n} t_i$, and we say that a path ω is *divergent* if for any $t \in \mathbb{R}$, there exists $j \in \mathbb{N}$ such that $\mathcal{D}_\omega(j){>}t$.

Definition 4 *An adversary* A *of a timed probabilistic system* TPS *is* divergent *if and only if for each state* s *of* TPS *the probability under* Prob_s^A *of the divergent paths of* $\mathit{Path}_{ful}^A(s)$ *is* 1*. Let* $\mathit{Adv}_{\mathsf{TPS}}$ *be the set of divergent adversaries of* TPS*.*

For motivation on why we consider *probabilistic* divergence, as opposed to the stronger notion where an adversary is divergent if and only if all its paths are divergent, see [KNSS02]. A restriction we impose on probabilistic timed systems is that of *non-zenoness*, which stipulates that there does not exist a state from which time cannot diverge, as we consider this situation to be a modelling error.

Definition 5 *A probabilistic timed system is* non-zeno *if and only if there exists a divergent adversary.*

3 Probabilistic Timed Automata

In this section we review the definition of probabilistic timed automata [KNSS02], a modelling framework for real-time systems exhibiting both nondeterministic and stochastic behaviour. The formalism is derived from classical timed automata [AD94, HNSY94] extended with discrete probability distributions over edges.

3.1 Clocks and Zones

Let $\mathcal{X}$ be a finite set of variables called *clocks* which take values from the time domain $\mathbb{R}$ (non-negative reals). A point $v \in \mathbb{R}^{|\mathcal{X}|}$ is referred to as a *clock valuation*. For any clock $x \in \mathcal{X}$, we use $v(x)$ to denote the projection of v on the x-axis. For any $v \in \mathbb{R}^{|\mathcal{X}|}$ and $t \in \mathbb{R}$, we use $v{+}t$ to denote the clock valuation defined as $v(x){+}t$ for all $x \in \mathcal{X}$. We use $v[X{:=}0]$ to denote the clock valuation obtained from v by resetting all of the clocks in $X \subseteq \mathcal{X}$ to 0, and leaving the values of all other clocks unchanged.

The set of *zones* of $\mathcal{X}$, written $\mathit{Zones}(\mathcal{X})$, is defined inductively by the syntax:

$$\zeta ::= x \leqslant d \,|\, c \leqslant x \,|\, x + c \leqslant y + d \,|\, \neg\zeta \,|\, \zeta \vee \zeta$$

where $x, y \in \mathcal{X}$ and $c, d \in \mathbb{N}$. We only consider canonical zones ensuring equality between their syntactic and semantic (subsets of $\mathbb{R}^{|\mathcal{X}|}$) representations. This enables us to use the above syntax interchangeably with set-theoretic operations.

The clock valuation v *satisfies* the zone ζ, written $v \triangleright \zeta$, if and only if ζ resolves to true after substituting each clock $x \in \mathcal{X}$ with the corresponding clock value $v(x)$ from v. We require the following classical operations on zones [HNSY94, Tri98]. For any zones $\zeta, \zeta' \in \mathit{Zones}(\mathcal{X})$ and subset of clocks $X \subseteq \mathcal{X}$, let:

$$\begin{aligned}
\swarrow_{\zeta'} \zeta &\stackrel{\text{def}}{=} \{v \,|\, \exists t \geqslant 0.\, (v + t \triangleright \zeta \;\wedge\; \forall t' \leqslant t.\, (v + t' \triangleright \zeta \vee \zeta'))\} \\
[X := 0]\zeta &\stackrel{\text{def}}{=} \{v \,|\, v[X := 0] \triangleright \zeta\} \\
\zeta[X := 0] &\stackrel{\text{def}}{=} \{v[X := 0] \,|\, v \in \zeta\}\,.
\end{aligned}$$

3.2 Syntax and Semantics of Probabilistic Timed Automata

Definition 6 *A* probabilistic timed automaton *is a tuple* $(L, \mathcal{X}, \mathit{inv}, \mathit{prob}, \mathcal{L})$ *where:*

- L *is a finite set of* locations*;*
- *the function* $\mathit{inv} : L \to \mathit{Zones}(\mathcal{X})$ *is the* invariant condition*;*
- *the finite set* $\mathit{prob} \subseteq L \times \mathit{Zones}(\mathcal{X}) \times \mathsf{Dist}(2^{\mathcal{X}} \times L)$ *is the* probabilistic edge relation*;*
- $\mathcal{L} : L \to 2^{AP}$ *is a labelling function assigning atomic propositions to locations.*

A *state* of a probabilistic timed automaton PTA is a pair $(l, v) \in L \times \mathbb{R}^{|\mathcal{X}|}$ such that $v \triangleright \mathit{inv}(l)$. Informally, the behaviour of a probabilistic timed automaton can be understood as follows. In any state (l, v), there is a nondeterministic choice of either (1) making a *discrete transition* or (2) letting *time pass*. In case (1), a discrete transition can be made according to any $(l, g, p) \in \mathit{prob}$ with source location l which is *enabled*; that is, zone g is satisfied by the current clock valuation v. Then the probability of moving to the location l' and resetting all

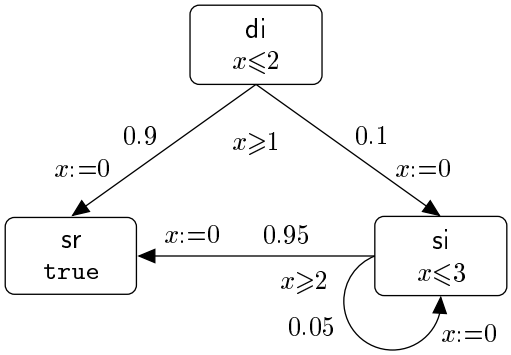

Figure 1: A probabilistic timed automaton modelling a probabilistic protocol.

of the clocks in X to 0 is given by $p(X, l')$. In case (2), the option of letting time pass is available only if the invariant condition $inv(l)$ is satisfied while time elapses.

An *edge* e of PTA is a tuple of the form (l, g, p, X, l') such that $(l, g, p) \in prob$ and $p(X, l') > 0$. Let edges denote the set of edges and $\mathsf{edges}(l, g, p)$ the set of edges corresponding to $(l, g, p) \in prob$.

Example. Consider the PTA modelling a simple probabilistic communication protocol given in Figure 1. The nodes represent the locations: di (sender has data, receiver idle); si (sender sent data, receiver idle); and sr (sender sent data, receiver received). The automaton starts in location di in which data has been received by the sender. After between 1 and 2 time units, the protocol makes a transition either to sr with probability 0.9 (data received), or to si with probability 0.1 (data lost). In si after 2 to 3 time units, the protocol will attempt to resend the data, which again can be lost, this time with probability 0.05.

We now give the semantics of probabilistic timed automata defined in terms of timed probabilistic systems.

Definition 7 *Let* PTA $= (L, \mathcal{X}, inv, prob, \mathcal{L})$ *be a probabilistic timed automaton. The* semantics of PTA *is defined as the timed probabilistic system* $\mathsf{TPS}_{\mathsf{PTA}} = (S, Steps, \mathcal{L}')$ *where:*

- $S \subseteq L \times \mathbb{R}^{|\mathcal{X}|}$ *and* $(l, v) \in S$ *if and only if* $v \triangleright inv(l)$*;*
- $((l, v), t, \mu) \in Steps$ *if and only if one of the following conditions holds*

  time transitions: $t \geqslant 0$, $\mu = \mu_{(l, v+t)}$ *and* $v + t' \triangleright inv(l)$ *for all* $0 \leqslant t' \leqslant t$

  discrete transitions: $t = 0$ *and there exists* $(l, g, p) \in prob$ *such that* $v \triangleright g$ *and for any* $(l', v') \in S$*:*

$$\mu(l', v') = \sum_{\substack{X \subseteq \mathcal{X} \,\&\, \\ v' = v[X:=0]}} p(X, l');$$

- $\mathcal{L}'(l, v) = \mathcal{L}(l)$ *for any* $(l, v) \in S$*.*

We say that PTA is non-zeno if and only if $\mathsf{TPS}_{\mathsf{PTA}}$ is non-zeno.

3.3 Probabilistic Timed Computation Tree Logic (PTCTL)

We now describe the probabilistic timed logic PTCTL which can be used to specify properties of probabilistic timed automata. PTCTL is a combination of two extensions of the temporal logic CTL, the timed logic TCTL [ACD93, HNSY94] and the probabilistic logic PCTL [HJ94]. The logic TCTL employs a set of *formula clocks*, $\mathcal{Z}$, disjoint from the clocks $\mathcal{X}$ of the probabilistic timed automaton. Formula clocks are assigned values by a *formula clock valuation* $\mathcal{E} \in \mathbb{R}^{|\mathcal{Z}|}$. The logic TCTL can express timing constraints and includes the reset quantifier $z.\phi$, used to reset the formula clock z so that ϕ is evaluated from a state at which $z = 0$. PTCTL is obtained by enhancing TCTL with the probabilistic quantifier $\mathcal{P}_{\sim\lambda}[\cdot]$.

Definition 8 *The syntax of* PTCTL *is defined as follows:*

$$\phi ::= a \mid \zeta \mid \neg\phi \mid \phi \vee \phi \mid z.\phi \mid \mathcal{P}_{\sim\lambda}[\phi \; \mathcal{U} \; \phi]$$

where $a \in AP$, $\zeta \in Zones(\mathcal{X} \cup \mathcal{Z})$, $z \in \mathcal{Z}$, $\sim \in \{\leqslant, <, >, \geqslant\}$ *and* $\lambda \in [0, 1]$.

In PTCTL we can express properties such as 'with probability at least 0.95, the system clock x does not exceed 3 before 8 time units elapse', which is represented as the formula $z.\mathcal{P}_{\geqslant 0.95}[(x \leqslant 3) \; \mathcal{U} \; (z{=}8)]$.

We write $v, \mathcal{E}$ to denote the composite clock valuation in $\mathbb{R}^{|\mathcal{X} \cup \mathcal{Z}|}$ obtained from $v \in \mathbb{R}^{|\mathcal{X}|}$ and $\mathcal{E} \in \mathbb{R}^{|\mathcal{Z}|}$. Given a state and formula clock valuation pair $(l, v), \mathcal{E}$, zone ζ and duration t, by abuse of notation we let $(l, v), \mathcal{E} \triangleright \zeta$ denote $v, \mathcal{E} \triangleright \zeta$, and $(l, v){+}t$ denote $(l, v{+}t)$.

Definition 9 *Let* $\mathsf{TPS} = (S, Steps, \mathcal{L}')$ *be the timed probabilistic system associated with the probabilistic timed automaton* PTA. *For any state* $s \in S$, *formula clock valuation* $\mathcal{E} \in \mathbb{R}^{|\mathcal{Z}|}$ *and* PTCTL *formula* θ, *the satisfaction relation* $s, \mathcal{E} \models \theta$ *is defined inductively as follows:*

$$\begin{array}{lcl}
s, \mathcal{E} \models a & \Leftrightarrow & a \in \mathcal{L}'(s) \\
s, \mathcal{E} \models \zeta & \Leftrightarrow & s, \mathcal{E} \triangleright \zeta \\
s, \mathcal{E} \models \phi \vee \psi & \Leftrightarrow & s, \mathcal{E} \models \phi \text{ or } s, \mathcal{E} \models \psi \\
s, \mathcal{E} \models \neg\phi & \Leftrightarrow & s, \mathcal{E} \not\models \phi \\
s, \mathcal{E} \models z.\phi & \Leftrightarrow & s, \mathcal{E}[z := 0] \models \phi \\
s, \mathcal{E} \models \mathcal{P}_{\sim\lambda}[\phi \; \mathcal{U} \; \psi] & \Leftrightarrow & p^A_{s,\mathcal{E}}(\phi \; \mathcal{U} \; \psi) \sim \lambda \text{ for all } A \in Adv_{\mathsf{TPS}}
\end{array}$$

where $p^A_{s,\mathcal{E}}(\phi \; \mathcal{U} \; \psi) = Prob^A_s\{\omega \in Path^A_{ful}(s) \mid \omega, \mathcal{E} \models \phi \; \mathcal{U} \; \psi\}$ *for any* $A \in Adv_{\mathsf{TPS}}$, *and, for any path* $\omega \in Path_{ful}(s)$, *we have that* $\omega, \mathcal{E} \models \phi \; \mathcal{U} \; \psi$ *if and only if there exists* $i \in \mathbb{N}$ *and* $t \leqslant \mathcal{D}_\omega(i{+}1){-}\mathcal{D}_\omega(i)$ *such that*

- $\omega(i){+}t, \mathcal{E}{+}\mathcal{D}_\omega(i){+}t \models \psi$;
- *if* $t' < t$, *then* $\omega(i){+}t', \mathcal{E}{+}\mathcal{D}_\omega(i){+}t' \models \phi \vee \psi$;
- *if* $j < i$ *and* $t' \leqslant \mathcal{D}_\omega(j{+}1){-}\mathcal{D}_\omega(j)$, *then* $\omega(j){+}t', \mathcal{E}{+}\mathcal{D}_\omega(j){+}t' \models \phi \vee \psi$.

In the following sections we will also consider the dual of the sub-formula $\phi \; \mathcal{U} \; \psi$, namely the release formula $\neg\phi \; \mathcal{V} \; \neg\psi$, where for any formulae ϕ, ψ, path ω and formula clock evaluation $\mathcal{E}$: $\omega, \mathcal{E} \models \phi \; \mathcal{V} \; \psi$ if and only if for all $i \in \mathbb{N}$ and $t \leqslant \mathcal{D}_\omega(i{+}1){-}\mathcal{D}_\omega(i)$, if

- $\omega(i){+}t', \mathcal{E}{+}\mathcal{D}_\omega(i){+}t' \not\models \phi \wedge \psi$ for all $t' < t$ and
- $\omega(j){+}t', \mathcal{E} + \mathcal{D}_\omega(j){+}t' \not\models \phi \wedge \psi$ for all $t' \leqslant \mathcal{D}_\omega(j{+}1){-}\mathcal{D}_\omega(j)$ and $j < i$,

then $\omega(i)+t, \mathcal{E}+\mathcal{D}_\omega(i)+t \models \psi$.

Furthermore, we use use the abbreviation $\Box\phi$ for the formula $\texttt{false}\ \mathcal{V}\ \phi$, that is, $\omega, \mathcal{E} \models \Box\psi$ if and only if $\omega(i)+t, \mathcal{E}+\mathcal{D}_\omega(i)+t \models \psi$ for all $i \in \mathbb{N}$ and $t \leqslant \mathcal{D}_\omega(i+1)-\mathcal{D}_\omega(i)$. In the standard manner, we refer to $\phi\ \mathcal{U}\ \psi$, $\phi\ \mathcal{V}\ \psi$ and $\Box\psi$ as *path formulae*.

We now present a number of lemmas concerning PTCTL that we will require in the remainder of the paper.

Lemma 10 *Let* PTA *be a timed probabilistic automaton,* $\mathsf{TPS} = (S, \mathit{Steps}, \mathcal{L}')$ *be the corresponding timed probabilistic system and* ϕ, ψ_1 *and* ψ_2 PTCTL *formulae. If* $s, \mathcal{E} \models \psi_1$ *implies* $s, \mathcal{E} \models \psi_2$ *for all state and formula clock valuation pairs* $s, \mathcal{E} \in S \times \mathbb{R}^{\mathcal{Z}}$, *then for any state and formula clock valuation pair* $s, \mathcal{E} \in S \times \mathbb{R}^{\mathcal{Z}}$:

- $s, \mathcal{E} \models \mathcal{P}_{\sim\lambda}[\phi\ \mathcal{U}\ \psi_2]$ *implies* $s, \mathcal{E} \models \mathcal{P}_{\sim\lambda}[\phi\ \mathcal{U}\ \psi_1]$,
- $s, \mathcal{E} \models z.\psi_1$ *implies* $s, \mathcal{E} \models z.\psi_2$.

Proof. The proof follows from the semantics of PTCTL (see Definition 9). □

Lemma 11 *Let* PTA *be a probabilistic timed automata,* $\mathsf{PS} = (S, \mathit{Steps}, \mathcal{L}')$ *be the corresponding timed probabilistic system and* ϕ *and* ψ *are PTCTL formulae. If* $s, \mathcal{E} \models \psi$ *implies* $s, \mathcal{E} \models \phi$ *for all state and formula clock valuation pairs* $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$, *then for any (infinite) path* ω *of* TPS *and formula clock valuation* $\mathcal{E}$:

$$\omega, \mathcal{E} \models \phi\ \mathcal{U}\ \psi \ \text{ if and only if } \ \omega, \mathcal{E} \not\models \neg\psi\ \mathcal{U}\ \neg\phi\,.$$

Proof. Let PTA be a probabilistic timed automaton, $\mathsf{TPS} = (S, \mathit{Steps}, \mathcal{L}')$ be the corresponding timed probabilistic system and ϕ and ψ be PTCTL formulae such that $s, \mathcal{E} \models \psi$ implies $s, \mathcal{E} \models \phi$ for all state and formula clock valuation pairs $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$. For the 'if' direction, consider any infinite path of TPS and formula clock valuation $\mathcal{E}$ such that $\omega, \mathcal{E} \models \phi\ \mathcal{U}\ \psi$. By Definition 9, there exists an $i \geqslant 0$ and $t \leqslant \mathcal{D}_\omega(i+1)-\mathcal{D}_\omega(i)$ such that:

- $\omega(i)+t, \mathcal{E}+\mathcal{D}_\omega(i)+t \models \psi$;
- if $t' < t$, then $\omega(i)+t', \mathcal{E}+\mathcal{D}_\omega(i)+t' \models \phi \vee \psi$;
- if $j < i$ and $t' \leqslant \mathcal{D}_\omega(j+1)-\mathcal{D}_\omega(j)$, then $\omega(j)+t', \mathcal{E}+\mathcal{D}_\omega(j)+t' \models \phi \vee \psi$.

Therefore, using the fact that $s, \mathcal{E} \models \psi$ implies $s, \mathcal{E} \models \phi$ for all $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$, there exists an $i \geqslant 0$ and $t \leqslant \mathcal{D}_\omega(i+1)-\mathcal{D}_\omega(i)$ such that:

- $\omega(i)+t, \mathcal{E}+\mathcal{D}_\omega(i)+t \not\models \neg\psi \vee \neg\phi$;
- if $t' < t$, then $\omega(i)+t', \mathcal{E}+\mathcal{D}_\omega(i)+t' \not\models \neg\phi$;
- if $j < i$ and $t' \leqslant \mathcal{D}_\omega(j+1)-\mathcal{D}_\omega(j)$, then $\omega(j)+t', \mathcal{E}+\mathcal{D}_\omega(j)+t' \not\models \neg\phi$.

and hence $\omega, \mathcal{E} \not\models \neg\psi\ \mathcal{U}\ \neg\phi$. Since this was for any path ω of TPS the 'if' direction holds.

The 'only if' direction follows similarly, using the identity $\neg\neg\theta \equiv \theta$ and since, from the hypothesis, $s, \mathcal{E} \models \neg\phi$ implies $s, \mathcal{E} \models \neg\psi$ for all $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$. □

The lemma below use the measure construction for probabilistic systems given in Section 2.3 and recall that, the states of the DTMC corresponding to an adversary A and state s are the finite paths of A that start in state s. Furthermore, it follows from this construction that a finite path (state in the DTMC) satisfies a formula when the last state of the path satisfies the formula.

Lemma 12 *Let* PTA *be a probabilistic timed automaton and* $\mathsf{TPS} = (S, Steps, \mathcal{L}')$ *be the corresponding timed probabilistic system. For any PTCTL formulae* ϕ *and* ψ*, adversary* $A \in Adv_{\mathsf{TPS}}$ *and state and formula clock valuation pair* $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$*:*

$$p^A_{s,\mathcal{E}}(\psi \,\mathcal{U}\, (\phi \wedge \psi) \vee p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi))) = p^A_{s,\mathcal{E}}(\phi \;\mathcal{V}\; \psi)$$

where for any $\omega, \mathcal{E} \in Path^A_{fin}(s) \times \mathcal{E}^{|\mathcal{Z}|}$ *:*

$$\omega, \mathcal{E} \models p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi)) \quad \textit{if and only if} \quad p^{A_\omega}_{last(\omega),\mathcal{E}}(\Box(\neg\phi \wedge \psi)) = 1\,.$$

Proof. Consider any probabilistic timed automata PTA with associated timed probabilistic system $\mathsf{TPS} = (S, Steps, \mathcal{L}')$, adversary $A \in Adv_{\mathsf{TPS}}$ and PTCTL formulae ϕ and ψ. First, for any finite path ω of $Path^A_{fin}$ and formula clock valuation $\mathcal{E}$, if $\omega, \mathcal{E} \models p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi))$, then $\omega', \mathcal{E} \models \Box(\neg\phi \wedge \psi)$ for all (infinite) paths $\omega' \in Path^{A_\omega}_{ful}(last(\omega))$. Therefore, for any finite path ω of $Path^A_{fin}$ and formula clock valuation $\mathcal{E}$:

$$\omega, \mathcal{E} \in p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi)) \;\Rightarrow\; \omega', \mathcal{E} \not\models \neg\phi \,\mathcal{U}\, \neg\psi \;\text{ for all } \omega' \in Path^{A_\omega}_{ful}(last(\omega))\,. \tag{1}$$

Now by Definition 9, for any (infinite) path ω' of TPS and formula clock valuation $\mathcal{E} \in \mathbb{R}^{|\mathcal{Z}|}$:

$$\begin{aligned}\omega', \mathcal{E} \models \neg\phi \,\mathcal{U}\, \neg\psi \;\Leftrightarrow\; & \omega', \mathcal{E} \models (\neg\phi \vee \neg\psi) \,\mathcal{U}\, \neg\psi \\ \;\Leftrightarrow\; & \omega', \mathcal{E} \models \big((\neg\phi \vee \neg\psi) \wedge \neg p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi))\big) \,\mathcal{U}\, \neg\psi \qquad \text{by (1).}\end{aligned}$$

Therefore, by the duality $\phi \,\mathcal{U}\, \psi \equiv \neg(\neg\phi \;\mathcal{V}\; \neg\psi)$ and the definition of A_ω (sse Section 2.3), it follows that for any state and formula clock valuation pair $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$:

$$\begin{aligned}p^A_{s,\mathcal{E}}(\phi \;\mathcal{V}\; \psi) = \;& 1 - p^A_{s,\mathcal{E}}(((\neg\phi \vee \neg\psi) \wedge \neg p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi))) \,\mathcal{U}\, \neg\psi) \\ = \;& 1 - p^A_{s,\mathcal{E}}(\neg((\phi \wedge \psi) \vee p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi))) \,\mathcal{U}\, \neg\psi)\end{aligned} \tag{2}$$

where the last step follows from the following derivation:

$$\begin{aligned}((\neg\phi \vee \neg\psi) \wedge \neg p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi))) \;&\equiv\; (\neg(\phi \wedge \psi) \wedge \neg p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi))) \\ &\equiv\; \neg((\phi \wedge \psi) \vee p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi)))\,.\end{aligned}$$

Finally, since for any finite path ω and formula clock valuation $\mathcal{E}$ we have $\omega, \mathcal{E} \models \neg\psi$ implies $\omega, \mathcal{E} \models \neg((\phi \wedge \psi) \vee p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi)))$, applying Lemma 11 to (2) we have that for any state and formula clock valuation pair $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$:

$$\begin{aligned}p^A_{s,\mathcal{E}}(\phi \;\mathcal{V}\; \psi) = \;& 1 - (1 - p^A_{s,\mathcal{E}}(\psi \,\mathcal{U}\, (\phi \wedge \psi) \vee p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi)))) \\ = \;& p^A_{s,\mathcal{E}}(\psi \,\mathcal{U}\, (\phi \wedge \psi) \vee p^A_{\geqslant 1}(\Box(\neg\phi \wedge \psi)))\end{aligned}$$

as required. □

algorithm PTCTLModelCheck(PTA, θ)

output: set of symbolic states $[\![\theta]\!]$ **such that**
$[\![a]\!] := \{(l, inv(l)) \mid l \in L \text{ and } l \in \mathcal{L}(a)\}$;
$[\![\zeta]\!] := \{(l, inv(l) \wedge \zeta) \mid l \in L\}$;
$[\![\neg\phi]\!] := \{(l, inv(l) \wedge \neg \bigvee_{(l,\zeta)\in[\![\phi]\!]} \zeta) \mid l \in L\}$;
$[\![\phi \vee \psi]\!] := [\![\phi]\!] \vee [\![\psi]\!]$;
$[\![z.\phi]\!] := \{(l, [\{z\}{:=}0]\zeta) \mid (l, \zeta) \in [\![\phi]\!]\}$;
$[\![\mathcal{P}_{\sim\lambda}[\phi \mathcal{U} \psi]]\!] :=$ Until$([\![\phi]\!], [\![\psi]\!], \sim\lambda)$;

Figure 2: Symbolic PTCTL model checking algorithm

4 Symbolic PTCTL Model Checking

In this section, we show how a probabilistic timed automaton may be model checked against PTCTL formulae. In order to represent the state sets computed during the model checking process, we use the concept of *symbolic state*: a symbolic state is a pair (l, ζ) comprising a location and a zone over $\mathcal{X} \cup \mathcal{Z}$. The set of state and formula clock valuation pairs corresponding to a symbolic state (l, ζ) is $\{(l, v), \mathcal{E} \mid v, \mathcal{E} \triangleright \zeta\}$, while the state set corresponding to a set of symbolic states is the union of those corresponding to each individual symbolic state. In the manner standard for model checking, we progress up the parse tree of a PTCTL formula, from the leaves to the root, recursively calling the algorithm PTCTLModelCheck, shown in Figure 2, to compute the set of symbolic states which satisfy each subformula. Handling observables and Boolean operations is classical, and we therefore reduce our problem to computing Until$([\![\phi_1]\!], [\![\phi_2]\!], \sim\lambda)$, which arises when we check probabilistically quantified formula.

Our technique depends on the following, which is a direct consequence of the semantics of PTCTL (Definition 9):

$$\{s, \mathcal{E} \mid s, \mathcal{E} \models \mathcal{P}_{\sim\lambda}[\phi \mathcal{U} \psi]\} = \begin{cases} \{s, \mathcal{E} \mid p^{\max}_{s,\mathcal{E}}(\phi \mathcal{U} \psi) \sim \lambda\} & \text{if } \sim \in \{<, \leqslant\} \\ \{s, \mathcal{E} \mid p^{\min}_{s,\mathcal{E}}(\phi \mathcal{U} \psi) \sim \lambda\} & \text{if } \sim \in \{\geqslant, >\} \end{cases} \tag{3}$$

where for any PTCTL path formula φ:

$$p^{\max}_{s,\mathcal{E}}(\varphi) \stackrel{\text{def}}{=} \sup_{A \in Adv_{\mathsf{TPS}}} p^{A}_{s,\mathcal{E}}(\varphi) \quad \text{and} \quad p^{\min}_{s,\mathcal{E}}(\varphi) \stackrel{\text{def}}{=} \inf_{A \in Adv_{\mathsf{TPS}}} p^{A}_{s,\mathcal{E}}(\varphi)\,.$$

We begin by introducing operations on symbolic states. In Section 4.2, we review algorithms for calculating maximum probabilities, while in Section 4.3 we present new algorithms for calculating minimum probabilities. In each case we include specialised algorithms for qualitative formulae ($\lambda \in \{0, 1\}$), as, for such formulae, verification can be performed through only an analysis of the underlying graph [HSP83, Pnu83]. Then in Section 4.4 we show how to ensure that the probabilistic timed automaton is non-zeno and, finally, in Section 4.5, we apply our approach to the example given in Figure 1.

Note that the cases $\mathcal{P}_{\geqslant 0}[\cdot]$ and $\mathcal{P}_{\leqslant 1}[\cdot]$ are trivially satisfied by all states, while the cases $\mathcal{P}_{<0}[\cdot]$ and $\mathcal{P}_{>1}[\cdot]$ are trivially not satisfied by any state, and therefore we omit these cases in our analysis.

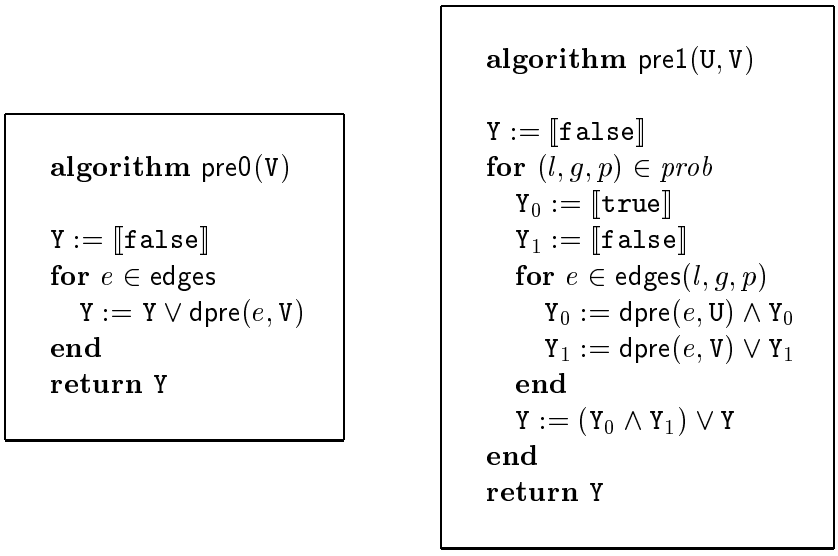

Figure 3: The functions pre0 and pre1

4.1 Operations on Symbolic States

In this section we extend the *time predecessor* and *discrete predecessor* functions tpre and dpre of [HNSY94, Tri98] to probabilistic timed automata. For any sets of symbolic states $\mathtt{U}, \mathtt{V} \subseteq L \times Zones(\mathcal{X} \cup \mathcal{Z})$, clock $x \in \mathcal{X} \cup \mathcal{Z}$ and edge (l, g, p, X, l'):

$$\begin{aligned} x.\mathtt{U} &\stackrel{\text{def}}{=} \{(l, [\{x\}{:=}0]\zeta^l_{\mathtt{U}}) \mid l \in L\} \\ \mathsf{tpre}_{\mathtt{U}}(\mathtt{V}) &\stackrel{\text{def}}{=} \{(l, \swarrow_{\zeta^l_{\mathtt{U}} \wedge inv(l)} (\zeta^l_{\mathtt{V}} \wedge inv(l)) \mid l \in L\} \\ \mathsf{dpre}((l, g, p, X, l'), \mathtt{U}) &\stackrel{\text{def}}{=} \{(l, g \wedge ([X := 0]\zeta^{l'}_{\mathtt{U}}))\}\,. \end{aligned}$$

where $\zeta^l_{\mathtt{U}} = \bigvee\{\zeta \mid (l, \zeta) \in \mathtt{U}\}$, i.e $\zeta^l_{\mathtt{U}}$ is the zone such that $v, \mathcal{E} \triangleright \zeta^l_{\mathtt{U}}$ if and only if $(l, v), \mathcal{E} \in \mathtt{u}$ for some $\mathtt{u} \in \mathtt{U}$. Furthermore, we define the conjunction and disjunction of sets of symbolic states as follows:

$$\mathtt{U} \wedge \mathtt{V} \stackrel{\text{def}}{=} \{(l, \zeta^l_{\mathtt{U}} \wedge \zeta^l_{\mathtt{V}}) \mid l \in L\} \text{ and } \mathtt{U} \vee \mathtt{V} \stackrel{\text{def}}{=} \{(l, \zeta^l_{\mathtt{U}} \vee \zeta^l_{\mathtt{V}}) \mid l \in L\}\,.$$

Finally, let $[\![\mathtt{false}]\!] = \varnothing$ and $[\![\mathtt{true}]\!] = \{(l, inv(l)) \mid l \in L\}$, the sets of symbolic states representing the empty and full state sets respectively.

4.2 Computing Maximum Probabilities

In this section we review the methods for calculating the set of states satisfying a formula of the form $\mathcal{P}_{\sim\lambda}[\phi\ \mathcal{U}\ \psi]$ which, from (3), reduces to the computation of $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi)$ for all state and formula clock valuation pairs $s, \mathcal{E}$. Note that, since we consider only non-zeno automata, when calculating these sets we can ignore the restriction to divergent adversaries. This is similar to verifying the same type of properties against (finite state) probabilistic systems with *fairness* constraints [BK98] and verifying (non-probabilistic) non-zeno timed automata against formulae of the form $\phi\ \exists\mathcal{U}\ \psi$ ('there exists a divergent path which satisfies $\phi\ \mathcal{U}\ \psi$') [HNSY94].

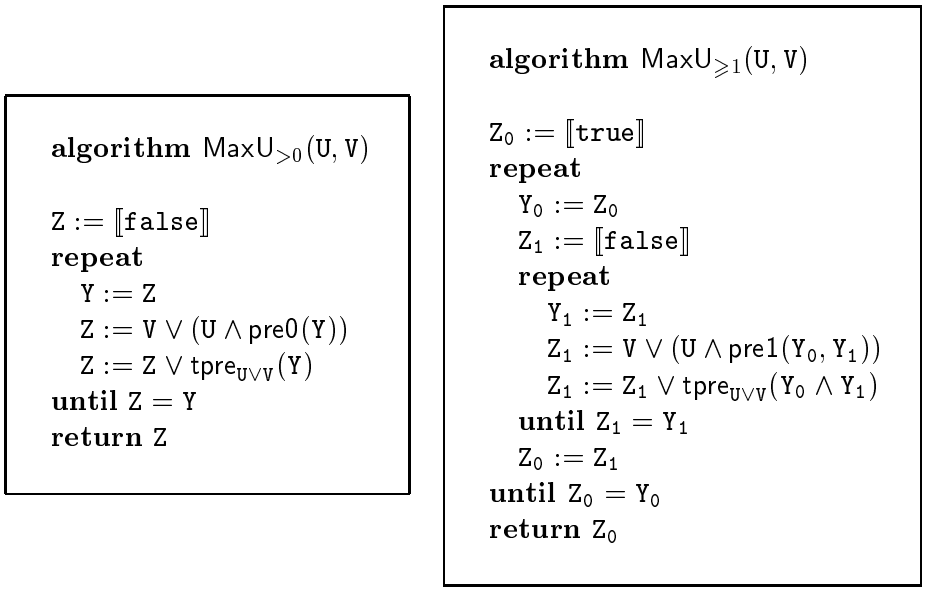

Figure 4: $\mathsf{MaxU}_{>0}$ and $\mathsf{MaxU}_{\geqslant 1}$ algorithms

We first recall the results for computing maximum *qualitative* probabilities of finite state probabilistic systems, which requires the introduction of the following functions. For a probabilistic system $\mathsf{PS} = (S, Steps, \mathcal{L}')$ and $X, Y \subseteq S$ let:

$$\begin{aligned} pre_0(X) &= \{s \in S \mid \exists (s,p) \in Steps.\, \exists s' \in X.\, p(s') > 0\} \\ pre_1(Y,X) &= \{s \in S \mid \exists (s,p) \in Steps.\, (\forall s' \in S.\, (p(s') > 0 \rightarrow s' \in Y) \wedge \exists s' \in X.\, p(s') > 0)\}\ . \end{aligned}$$

Intuitively, $s \in pre_0(X)$ if one can go from s to X with positive probability and $s \in pre_1(Y,X)$ if one can go from s to X with positive probability and with probability 1 reach Y. Using these functions we have the following proposition[1].

Proposition 13 *[dA97] If $\mathsf{PS} = (S, Steps, \mathcal{L})$ is a finite state probabilistic system and ϕ, ψ are PCTL formulae, then*

- $\{s \in S \mid p_s^{\max}(\phi\ \mathcal{U}\ \phi) > 0\}$ *equals the fixpoint* $\mu X.(\psi \vee (\phi \wedge pre_0(X)))$*;*
- $\{s \in S \mid p_s^{\max}(\phi\ \mathcal{U}\ \psi) \geqslant 1\}$ *equals the double fixpoint* $\nu Y.\mu X.(\psi \vee (\phi \wedge pre_1(Y,X)))$*.*

We adapt this approach to probabilistic timed automata. First, using the function dpre, the analogues of pre_0 and pre_1 for the *discrete* transitions of a PTA are given in Figure 3. It therefore remains to consider the *time* transitions of a PTA. For such transitions, we must take into account the state and formula clock valuation pairs that are passed through as time elapses. More precisely, for PTCTL, when using the time predecessor function we must ensure that we remain in the set of symbolic states satisfying $\phi \vee \psi$, that is, take the time predecessor $\mathsf{tpre}_{[\![\phi]\!] \vee [\![\psi]\!]}(\cdot)$. Following this observation, Figure 4 presents the algorithms for computing $\{s, \mathcal{E} \mid p_{s,\mathcal{E}}^{\max}(\phi\ \mathcal{U}\ \psi) > 0\}$ and $\{s, \mathcal{E} \mid p_{s,\mathcal{E}}^{\max}(\phi\ \mathcal{U}\ \psi) \geqslant 1\}$.

In the case of computing *quantitative* maximum probabilities we use the approach described in [KNS01]. The algorithm is given in Figure 5. The key observation is that to

[1] See [BdA95] for the definitions of PCTL, $p_s^{\max}(\phi\ \mathcal{U}\ \psi)$ and $p_s^{\min}(\phi\ \mathcal{U}\ \psi)$.

preserve the probabilistic branching one must take the conjunctions of symbolic states generated by edges from the same distribution. Lines 1–4 deal with the initialisation of $\mathtt{Z}$, which is set equal to the set of time predecessors of $\mathtt{V}$, and the set of edges $E_{(l,g,p)}$ associated with each probabilistic edge $(l,g,p) \in \mathit{prob}$. Lines 5–20 generate a finite-state graph, the nodes of which are symbolic states, obtained by iterating timed and discrete predecessor operations (line 8), and taking conjunctions (lines 12–17). The edges of the graph are partitioned into the sets $E_{(l,g,p)}$ for $(l,g,p) \in \mathit{prob}$, with the intuition that $(\mathtt{z},(X,l'),\mathtt{z}') \in E_{(l,g,p)}$ corresponds to a transition from any state in $\mathtt{z}$ to some state in $\mathtt{z}'$ when the outcome (X,l') of the probabilistic edge (l,g,p) is chosen. The graph edges are added in lines 11 and 15. The termination of lines 5–20 is guaranteed (see [KNS01]). Line 21 describes the manner in which the probabilistic edges of the probabilistic timed automaton are used in combination with the computed edge sets to construct the probabilistic transition relation *Steps*. Finally, in line 22, model checking is performed on the resulting finite-state probabilistic system PS to obtain the maximum probability of reaching $\mathsf{tpre}_{\mathtt{U}\vee\mathtt{V}}(\mathtt{V})$ for each $\mathtt{z} \in \mathtt{Z}$. Note that we write $\mathtt{z} \neq \varnothing$ if and only if $\mathtt{z}$ encodes at least one state and formula clock valuation pair. The following proposition states the correctness of this algorithm.

Proposition 14 *For any probabilistic timed automaton* PTA*, corresponding timed probabilistic system* $\mathsf{TPS} = (S, \mathit{Steps}, \mathcal{L}')$ *and* PTCTL *formula* $\mathcal{P}_{\lesssim\lambda}[\phi\ \mathcal{U}\ \psi]$*, if* $\mathsf{PS} = (\mathtt{Z}, \mathit{Steps})$ *is the probabilistic system generated by* $\mathsf{MaxU}([\![\phi]\!],[\![\psi]\!],\gtrsim\lambda)$ *then for any* $s,\mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$*:*

- $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi){>}0$ *if and only if* $s,\mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathtt{Z})$*;*
- *if* $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi){>}0$*, then* $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi)$ *equals*

$$\max\left\{ \mathit{MaxProbReach}(\mathtt{z}, \mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!]) \;\middle|\; \mathtt{z} \in \mathtt{Z} \text{ and } s,\mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathtt{z}) \right\}.$$

Proof. See Appendix A. □

Combining the above results we set $\mathsf{Until}([\![\phi]\!],[\![\psi]\!],\lesssim\lambda)$ equal to:

- $[\![\mathtt{true}]\!] \setminus \mathsf{MaxU}_{>0}([\![\phi]\!],[\![\psi]\!])$ if $\lesssim = \leqslant$ and $\lambda = 0$;
- $[\![\mathtt{true}]\!] \setminus \mathsf{MaxU}_{\geqslant 1}([\![\phi]\!],[\![\psi]\!])$ if $\lesssim = <$ and $\lambda = 1$;
- $[\![\mathtt{true}]\!] \setminus \mathsf{MaxU}([\![\phi]\!],[\![\psi]\!],\not\gtrsim\lambda)$ otherwise.

As in the case of finite state probabilistic model checking, we can use the qualitative algorithms as *precomputation* algorithms when computing quantitative probabilities. In particular, we can set $\mathsf{Until}([\![\phi]\!],[\![\psi]\!],\lesssim\lambda)$, for $\lambda \in (0,1)$, equal to:

$$[\![\mathtt{true}]\!] \setminus \mathsf{MaxU}(\mathsf{MaxU}_{>0}([\![\phi]\!],[\![\psi]\!]) \setminus \mathsf{MaxU}_{\geqslant 1}([\![\phi]\!],[\![\psi]\!]), \mathsf{MaxU}_{\geqslant 1}([\![\phi]\!],[\![\psi]\!]), \not\gtrsim \lambda).$$

4.3 Computing Minimum Probabilities

We now consider the problem of verifying formulae of the form $\mathcal{P}_{\gtrsim\lambda}[\phi\ \mathcal{U}\ \psi]$ which, using (3), reduces to computing $p^{\min}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi)$ for all state and formula clock valuation pairs $s,\mathcal{E}$. As in the cases for (non-probabilistic) timed automata and (finite-state) probabilistic systems with fairness constraints, when considering properties which have universal quantification over paths/adversaries the standard algorithm can no longer be applied. For example, for

algorithm $\mathsf{MaxU}(\mathtt{U}, \mathtt{V}, \gtrsim \lambda)$

1. $\mathtt{Z} := \mathsf{tpre}_{\mathtt{U} \vee \mathtt{V}}(\mathtt{V})$
2. **for** $(l, g, p) \in \mathit{prob}$
3. $\quad E_{(l,g,p)} := \varnothing$
4. **end for**
5. **repeat**
6. $\quad \mathtt{Y} := \mathtt{Z}$
7. $\quad$ **for** $\mathtt{y} \in \mathtt{Y} \wedge (l, g, p) \in \mathit{prob} \wedge e = (l, g, p, X, l') \in \mathsf{edges}(l, g, p)$
8. $\quad\quad \mathtt{z} := \mathtt{U} \wedge \mathsf{dpre}(e, \mathsf{tpre}_{\mathtt{U} \vee \mathtt{V}}(\mathtt{y}))$
9. $\quad\quad$ **if** $(\mathtt{z} \neq \varnothing) \wedge (\mathtt{z} \notin \mathsf{tpre}_{\mathtt{U} \vee \mathtt{V}}(\mathtt{V}))$
10. $\quad\quad\quad \mathtt{Z} := \mathtt{Z} \cup \{\mathtt{z}\}$
11. $\quad\quad\quad E_{(l,g,p)} := E_{(l,g,p)} \cup \{(\mathtt{z}, (X, l'), \mathtt{y})\}$
12. $\quad\quad\quad$ **for** $(\bar{\mathtt{z}}, (\bar{X}, \bar{l}'), \bar{\mathtt{y}}) \in E_{(l,g,p)}$
13. $\quad\quad\quad\quad$ **if** $(\mathtt{z} \wedge \bar{\mathtt{z}} \neq \varnothing) \wedge ((\bar{X}, \bar{l}') \neq (X, l')) \wedge (\mathtt{z} \wedge \bar{\mathtt{z}} \notin \mathsf{tpre}_{\mathtt{U} \vee \mathtt{V}}(\mathtt{V}))$
14. $\quad\quad\quad\quad\quad \mathtt{Z} := \mathtt{Z} \cup \{\mathtt{z} \wedge \bar{\mathtt{z}}\}$
15. $\quad\quad\quad\quad\quad E_{(l,g,p)} := E_{(l,g,p)} \cup \{(\mathtt{z} \wedge \bar{\mathtt{z}}, (X, l'), \bar{\mathtt{y}}), (\mathtt{z} \wedge \bar{\mathtt{z}}, (\bar{X}, \bar{l}'), \mathtt{y})\}$
16. $\quad\quad\quad\quad$ **end if**
17. $\quad\quad\quad$ **end for**
18. $\quad\quad$ **end if**
19. $\quad$ **end for**
20. **until** $\mathtt{Z} = \mathtt{Y}$
21. **construct** $\mathsf{PS} = (\mathtt{Z}, \mathit{Steps})$ **where** $(\mathtt{z}, \rho) \in \mathit{Steps}$ **if and only if**
    there exists $(l, g, p) \in \mathit{prob}$ **and** $E \subseteq E_{(l,g,p)}$ **such that**
    - $(\mathtt{z}', \mathtt{e}, \mathtt{z}'') \in E \;\Rightarrow\; \mathtt{z}' = \mathtt{z}$
    - $(\mathtt{z}, \mathtt{e}, \mathtt{z}') \neq (\mathtt{z}, \mathtt{e}', \mathtt{z}'') \in E \;\Rightarrow\; \mathtt{e} \neq \mathtt{e}'$
    - E **is maximal**
    - $\rho(\mathtt{z}') = \sum\{\!|\, p(X, l') \mid (\mathtt{z}, (X, l'), \mathtt{z}') \in E \,|\!\}\ \ \forall \mathtt{z}' \in \mathtt{Z}$
22. **return** $\bigvee\{\mathsf{tpre}_{\mathtt{U} \vee \mathtt{V}}(\mathtt{z}) \mid \mathtt{z} \in \mathtt{Z} \wedge \mathit{MaxProbReach}(\mathtt{z}, \mathsf{tpre}_{\mathtt{U} \vee \mathtt{V}}(\mathtt{V})) \gtrsim \lambda\}$

Figure 5: Algorithm $\mathsf{MaxUntil}(\cdot, \cdot, \gtrsim \lambda)$

any formula clock $z \in \mathcal{Z}$, under divergent adversaries the minimum probability of reaching $z{>}1$ is 1; however, if we remove the restriction to time divergent adversaries the minimum probability is 0.

The techniques we introduce here are based on those for non-probabilistic timed automata [HNSY94], which we now recall. In [HNSY94], it is shown that verifying $\phi\, \forall\mathcal{U}\, \psi$ ('all divergent paths satisfy $\phi\, \mathcal{U}\, \psi$') reduces to computing the fixpoint:

$$\mu X.(\psi \vee \neg z.((\neg X)\ \exists\mathcal{U}\ (\neg(\phi \vee X) \vee (z{>}c))) \tag{4}$$

for any $c \in \mathbb{N}$ greater than 0. The important point is that the universal quantification over paths has been replaced by a existential quantification, allowing one to ignore the restriction to time divergence in the verification procedure.

For the analysis of probabilistic timed automata it is convenient to consider, instead of

until, the dual, release, formula $\phi\ \exists\mathcal{V}\ \psi$ ('there exists a divergent path satisfying $\phi\ \mathcal{V}\ \psi$'). Using (4), it follows that verifying the formula $\phi\ \exists\mathcal{V}\ \psi$ can be performed by computing the fixpoint:

$$\nu X.(\psi \wedge z.(X\ \exists\mathcal{U}\ ((\phi \wedge X) \vee (z{>}c)))) \,. \tag{5}$$

Now, from the semantics of PTCTL and the duality $\phi\,\mathcal{U}\,\psi \equiv \neg(\neg\phi\ \mathcal{V}\ \neg\psi)$, we have, for any state s of $\mathsf{TPS}_{\mathsf{PTA}}$ and formula clock valuation $\mathcal{E}$:

$$\begin{aligned} p^{\min}_{s,\mathcal{E}}(\phi\,\mathcal{U}\,\psi) &= \inf_{A\in Adv_{\mathsf{TPS}}} p^{A}_{s,\mathcal{E}}(\neg(\neg\phi\ \mathcal{V}\ \neg\psi)) \\ &= \inf_{A\in Adv_{\mathsf{TPS}}} 1 - p^{A}_{s,\mathcal{E}}(\neg\phi\ \mathcal{V}\ \neg\psi) \\ &= 1 - \sup_{A\in Adv_{\mathsf{TPS}}} p^{A}_{s,\mathcal{E}}(\neg\phi\ \mathcal{V}\ \neg\psi) \\ &= 1 - p^{\max}_{s,\mathcal{E}}(\neg\phi\ \mathcal{V}\ \neg\psi) \,. \end{aligned}$$

Therefore, to verify $\mathcal{P}_{\gtrsim\lambda}[\phi\,\mathcal{U}\,\psi]$, it suffices to calculate $p^{\max}_{s,\mathcal{E}}(\neg\phi\ \mathcal{V}\ \neg\psi)$ for all state and formula clock valuation pairs. In the case when $\lambda{=}1$, by replacing the $\exists$ operator with $\neg\mathcal{P}_{<1}[\cdot]$ in (5), we arrive at the following proposition.

Proposition 15 *For any positive $c \in \mathbb{N}$ and PTCTL formulae ϕ, ψ, if $z \in \mathcal{Z}$ does not appear in either ϕ or ψ, then the set $\{s, \mathcal{E} \,|\, p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{V}\ \psi) \geqslant 1\}$ is given by the fixpoint $\nu X.(\psi \wedge z.\neg\mathcal{P}_{<1}[X\ \mathcal{U}\ ((X \wedge \phi) \vee z{>}c)])$.*

Proof. Consider any positive $c \in \mathbb{N}$, PTCTL formulae ϕ, ψ, and $z \in \mathcal{Z}$ such that z does not appear in either ϕ or ψ. To ease notation we let:

$$p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi) = \{s, \mathcal{E} \,|\, p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{V}\ \psi) \geqslant 1\} \,,$$

and prove the proposition by showing:

1. the set $p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi)$ is a fixpoint of $G_1(\cdot, c)$;
2. if $G_1(Y, c) = Y$, then $Y \subseteq p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi)$

where $G_1(X, c) = \psi \wedge z.\neg\mathcal{P}_{<1}[X\ \mathcal{U}\ ((X \wedge \phi) \vee z{>}c)]$. First, since, for any $X \subseteq S \times \mathbb{R}^{|\mathcal{Z}|}$, $X \supseteq [\![z.\neg\mathcal{P}_{<1}[X\ \mathcal{U}\ ((X \wedge \phi) \vee z{>}c)]]\!]$ it follows that: $X \supseteq G_1(X, c)$ for all $X \subseteq S \times \mathbb{R}^{|\mathcal{Z}|}$. Therefore, to prove that $p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi)$ is a fixpoint it is sufficient to show that:

$$G_1\big(p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi), c\big) \supseteq p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi) \,.$$

Now, by definition of $\phi\ \mathcal{V}\ \psi$ (see Section 3.3) we have that:

- For any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$, if $s, \mathcal{E} \models \phi \wedge \psi$, then $\omega, \mathcal{E} \models \phi\ \mathcal{V}\ \psi$ for all paths $\omega \in Path_{ful}(s)$, and hence $s, \mathcal{E} \models \phi \wedge \psi$ implies $s, \mathcal{E} \in p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi)$. It follows that $s, \mathcal{E} \models (\phi \wedge \psi) \vee z{>}c$ implies $s, \mathcal{E} \models \big(p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi) \wedge \phi\big) \vee z{>}c$ and therefore, using Lemma 10, for any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$:

$$\begin{aligned} &s, \mathcal{E} \models z.\neg\mathcal{P}_{<1}\left[\,p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi)\ \mathcal{U}\ ((\phi \wedge \psi) \vee z{>}c)\,\right] \\ &\Rightarrow s, \mathcal{E} \models z.\neg\mathcal{P}_{<1}\left[\,p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi)\ \mathcal{U}\ \left(\big(p^{\max}_{\geqslant 1}(\phi\ \mathcal{V}\ \psi) \wedge \phi\big) \vee z{>}c\right)\,\right] . \end{aligned} \tag{6}$$

- For any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$ and $\omega \in Path_{ful}(s)$, if $\omega, \mathcal{E} \models \phi \ \mathcal{V} \ \psi$, then $s, \mathcal{E} \models \psi$, and hence:

$$s, \mathcal{E} \in p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) \ \Rightarrow \ s, \mathcal{E} \models \psi \, . \tag{7}$$

- As the satisfaction of PTCTL is with respect to divergent adversaries, for any $s, \mathcal{E} \in p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi)$, there exists an adversary A such that, from $s, \mathcal{E}$ with probability 1, one remains in $p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi)$ until either a state satisfying $\phi \wedge \psi$ is reached or more than c time units pass. Therefore, since the clock z does not appear in ϕ or ψ, for any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$:

$$s, \mathcal{E} \in p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) \quad \Rightarrow \quad s, \mathcal{E}[z := 0] \models \neg \mathcal{P}_{<1} \left[p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) \, \mathcal{U} \, ((\phi \wedge \psi) \vee z{>}c) \right] . \tag{8}$$

By definition of G_1:

$$\begin{aligned}
& G_1(p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi), c) = \psi \wedge z.\neg \mathcal{P}_{<1} \left[p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) \, \mathcal{U} \ \left(\left(\phi \wedge p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) \right) \vee z{>}c \right) \right] \\
& \supseteq \ \psi \wedge z.\neg \mathcal{P}_{<1} \left[p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) \, \mathcal{U} \ ((\phi \wedge \psi) \vee z{>}c) \right] && \text{by (6)} \\
& \supseteq \ \psi \wedge p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) && \text{by (8) and Definition 9} \\
& = \ p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) && \text{by (7)}
\end{aligned}$$

and hence $p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi)$ is a fixpoint of $G_1(X, c)$.

To complete the proof it remains to show that, if $G_1(Y, c) = Y$, then $Y \subseteq p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi)$ which we prove by contradiction. Therefore, suppose that there exists a of states Y such that $G_1(Y, c) = Y$ and $p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi) \subset Y$. Now for any $s, \mathcal{E} \in Y \setminus p^{\max}_{\geqslant 1}(\phi \ \mathcal{V} \ \psi)$, and adversary A, under A starting from $s, \mathcal{E}$ the probability of satisfying $\phi \ \mathcal{V} \ \psi$ is less than 1, and therefore the probability of satisfying the dual formula $\neg\phi \, \mathcal{U} \, \neg\psi$ is greater than 0. It then follows that there exists a path $\omega \in Path^{A}_{ful}(s)$ such that $\omega, \mathcal{E} \models \neg\phi \, \mathcal{U} \, \neg\psi$, and since z does not appear in either ϕ or ψ, $\omega, \mathcal{E}[z := 0] \models \neg\phi \, \mathcal{U} \, \neg\psi$. Hence, there exists some duration t_A such that at some point along this path $\neg\psi \wedge (z = t_A)$ is true and at all preceding points $\neg\phi \vee \neg\psi$ is true.

However, since $s, \mathcal{E} \in Y$, and therefore $s, \mathcal{E} \in G_1(Y, c)$ it follows that there exists an adversary such that with probability 1 from $s, \mathcal{E}[z := 0]$ one remains in Y while $z \leqslant c$ unless a state in Y which satisfies ϕ is reached. Since the above holds for any $s', \mathcal{E}' \in Y$ and z does not appear in ϕ or ψ, iterating the above result, we have that for any $n \in \mathbb{N}$ we can construct an adversary A' such that, for any $n \in \mathbb{N}$, under A' from $s, \mathcal{E}$ one remains in Y while $z \leqslant n \cdot c$ unless a state in Y which satisfies ϕ is reached. Furthermore, since $Y = G_1(Y, c)$ it follows that $Y \subseteq \psi$, and hence under A', for any $n \in \mathbb{N}$, with probability 1, from $s, \mathcal{E}$ one remains in states satisfying ψ while $z \leqslant n \cdot c$ unless a state satisfying $\phi \wedge \psi$ is reached. From above, there exists some duration $t_{A'}$ and path $\omega' \in Path^{A'}_{ful}(s)$ such that at some point along this path $\neg\psi \wedge (z = t_A)$ is true and at all preceding points $\neg\phi \vee \neg\psi$ is true. However, considering any n such that $n \cdot c > t_A$ (which exists since $c > 0$) leads to a contradiction. □

The algorithm for calculating the set $\{s, \mathcal{E} \mid p^{\max}_{s,\mathcal{E}}(\phi \ \mathcal{V} \ \psi){\geqslant}1\}$ follows from Proposition 15 and is given in Figure 6. Note that we cannot use the same approach for calculating the set $\{s, \mathcal{E} \mid p^{\max}_{s,\mathcal{E}}(\phi \ \mathcal{V} \ \psi){>}0\}$, i.e. in (5) replace $\exists$ with $\neg\mathcal{P}_{\leqslant 0}[\cdot]$. This is because the greatest fixpoint in this case yields the set of state and formula clock valuation pairs for which, under some divergent adversary, there exists a path which satisfies $\phi \ \mathcal{V} \ \psi$, which does not imply that the probability of satisfying $\phi \ \mathcal{V} \ \psi$ is greater than zero.

Instead, we employ the following proposition, which together with Proposition 15 provides us with a method for verifying $\mathcal{P}_{\gtrsim \lambda}[\phi \, \mathcal{U} \, \psi]$ when $\lambda \in [0, 1)$.

algorithm $\mathsf{MaxV}_{\geqslant 1}(c, \mathtt{U}, \mathtt{V})$

$\mathtt{Z}{:=}[\![\mathtt{true}]\!]$
repeat
  $\mathtt{Y}{:=}\mathtt{Z}$
  $\mathtt{Z}{:=}\mathtt{V} \wedge z.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, (\mathtt{U} \wedge \mathtt{Y}) \vee [\![z{>}c]\!])$
until $\mathtt{Z} = \mathtt{Y}$
return $\mathtt{Z}$

algorithm NonZeno

$\mathtt{Z}{:=}[\![\mathtt{true}]\!]$
repeat
  $\mathtt{Y}{:=}\mathtt{Z}$
  $\mathtt{Z}{:=}z.\mathsf{MaxU}_{\geqslant 1}([\![\mathtt{true}]\!], \mathtt{Y} \wedge [\![z{=}1]\!])$
until $\mathtt{Z} = \mathtt{Y}$
return $\mathtt{Z}_0$

Figure 6: $\mathsf{MaxV}_{\geqslant 1}(c, \mathtt{U}, \mathtt{V})$ and NonZeno algorithms

Proposition 16 *For any probabilistic timed automata* PTA*, corresponding timed probabilistic system* $\mathsf{TPS} = (S, Steps, \mathcal{L}')$*,* $s \in S$*, formula clock valuation* $\mathcal{E} \in \mathbb{R}^{|\mathcal{Z}|}$ *and PTCTL formulae* ϕ, ψ*:*

$$p^{\max}_{s,\mathcal{E}}(\phi \ \mathcal{V} \ \psi) = p^{\max}_{s,\mathcal{E}}(\psi \ \mathcal{U} \ \neg\mathcal{P}_{<1}[\phi \ \mathcal{V} \ \psi])\,.$$

Proof. Consider any PTCTL formulae ϕ and ψ and let $A_{\max}$ be an adversary such that for any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$:

$$p^{A_{\max}}_{s,\mathcal{E}}(\psi \ \mathcal{U} \ (\psi \wedge \phi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]) = p^{\max}_{s,\mathcal{E}}(\psi \ \mathcal{U} \ (\psi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)])$$

and if one reaches any $s', \mathcal{E}'$ satisfying $\neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]$, then $A_{\max}$ behaves like the adversary A for which:

$$p^{A}_{s',\mathcal{E}'}(\Box\,(\neg\phi \wedge \psi)) = \ p^{\max}_{s,\mathcal{E}}(\Box(\neg\phi \wedge \psi)) = 1$$

since $s', \mathcal{E}' \models \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]$. Note that, this adversary is well defined (and divergent) since for any adversary A, once a state satisfying $\neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]$ is reached, the behaviour of A has no influence on the probability of satisfaction of the formula $\psi \ \mathcal{U} \ (\psi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]$. Furthermore, for any $s', \mathcal{E}'$ such that $p^{\max}_{s,\mathcal{E}}(\Box(\neg\phi \wedge \psi)) = 1$, the fact that there exists an adversary A such that $p^{A}_{s,\mathcal{E}}(\Box(\neg\phi \wedge \psi)) = 1$ follows from Proposition 15.

Now, since for any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$ and $A \in Adv_{\mathsf{TPS}}$ we have $p^{A}_{s,\mathcal{E}}(\Box\,(\neg\phi \wedge \psi)) = 1$ if and only if $\omega, \mathcal{E} \models \Box\,(\neg\phi \wedge \psi)$ for all $\omega \in Path^{A}_{ful}(t)$, it follows from the construction of $A_{\max}$ that, for any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$ and path $\omega \in Path^{A_{\max}}_{ful}(s)$, if $\omega, \mathcal{E} \models \psi \ \mathcal{U} \ (\phi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]$, then $\omega, \mathcal{E} \models \phi \ \mathcal{V} \ \psi$. Therefore, for all $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$:

$$p^{A_{\max}}_{s,\mathcal{E}}(\psi \ \mathcal{U} \ (\phi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]) \leqslant p^{A_{\max}}_{s,\mathcal{E}}(\phi \ \mathcal{V} \ \psi)\,,$$

and hence, by the definition of $A_{\max}$, it follows that

$$p^{\max}_{s,\mathcal{E}}(\psi \ \mathcal{U} \ (\phi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]) \leqslant p^{\max}_{s,\mathcal{E}}(\phi \ \mathcal{V} \ \psi) \ \ \forall s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}. \tag{9}$$

On the other hand, from Lemma 12 and the fact that for any $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$ and adversary A: $p^{A}_{s,\mathcal{E}}(\Box(\neg\phi \wedge \psi)) = 1$ implies $s, \mathcal{E} \models \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]$ we have for any adversary A and $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$: $p^{A}_{s,\mathcal{E}}(\psi \ \mathcal{U} \ (\psi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]) \geqslant p^{A}_{s,\mathcal{E}}(\phi \ \mathcal{V} \ \psi)$, and hence it follows that:

$$p^{\max}_{s,\mathcal{E}}(\psi \ \mathcal{U} \ (\phi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]) \geqslant p^{\max}_{s,\mathcal{E}}(\phi \ \mathcal{V} \ \psi) \ \ \forall s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}. \tag{10}$$

Combining (9) and (10) we have:

$$p^{\max}_{s,\mathcal{E}}(\psi \,\mathcal{U}\, (\phi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]) = p^{\max}_{s,\mathcal{E}}(\phi \,\mathcal{V}\, \psi) \;\; \forall s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}. \tag{11}$$

Now using (11) we have $s, \mathcal{E} \models \neg\mathcal{P}_{<1}[\psi \,\mathcal{U}\, (\phi \wedge \psi) \vee \neg\mathcal{P}_{<1}[\Box(\neg\phi \wedge \psi)]]$ if and only if $s, \mathcal{E} \models \neg\mathcal{P}_{<1}[\phi \,\mathcal{V}\, \psi]$, and since for any formulae θ_1, θ_2 and $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$:

$$p^{\max}_{s,\mathcal{E}}(\theta_1 \,\mathcal{U}\, \theta_2) = p^{\max}_{s,\mathcal{E}}(\theta_1 \,\mathcal{U}\, \neg\mathcal{P}_{<1}[\theta_1 \,\mathcal{U}\, \theta_2])\,,$$

using (11) again, we have:

$$p^{\max}_{s,\mathcal{E}}(\psi \,\mathcal{U}\, \neg\mathcal{P}_{<1}[\phi \,\mathcal{V}\, \psi]) = p^{\max}_{s,\mathcal{E}}(\phi \,\mathcal{V}\, \psi)\,.$$

as required. □

Combining the above results, we set $\mathsf{Until}(\llbracket\phi\rrbracket, \llbracket\psi\rrbracket, \gtrsim \lambda)$ to:

- $\llbracket\mathtt{true}\rrbracket \setminus \mathsf{MaxV}_{\geqslant 1}(c, \llbracket\neg\phi\rrbracket, \llbracket\neg\psi\rrbracket)$ if $\gtrsim \,= >$ and λ=0;
- $\llbracket\mathtt{true}\rrbracket \setminus \mathsf{MaxU}_{>0}(\llbracket\neg\psi\rrbracket, \mathsf{MaxV}_{\geqslant 1}(c, \llbracket\neg\phi\rrbracket, \llbracket\neg\psi\rrbracket))$ if $\gtrsim \,= \geqslant$ and λ=1;
- $\llbracket\mathtt{true}\rrbracket \setminus \mathsf{MaxU}(\llbracket\neg\psi\rrbracket, \mathsf{MaxV}_{\geqslant 1}(c, \llbracket\neg\phi\rrbracket, \llbracket\neg\psi\rrbracket), \geqslant 1{-}\lambda)$ if $\gtrsim \,= >$ and $\lambda \in (0, 1)$;
- $\llbracket\mathtt{true}\rrbracket \setminus \mathsf{MaxU}(\llbracket\neg\psi\rrbracket, \mathsf{MaxV}_{\geqslant 1}(c, \llbracket\neg\phi\rrbracket, \llbracket\neg\psi\rrbracket), > 1{-}\lambda)$ if $\gtrsim \,= \geqslant$ and $\lambda \in (0, 1)$.

4.4 Checking Non-Zenoness

We now consider how to check that the probabilistic timed automaton under study is non-zeno. In the non-probabilistic case checking non-zenoness corresponds to finding the greatest fixpoint $\nu X.(z.(\mathtt{true}\ \exists\mathcal{U}\ ((z{=}1) \wedge X)))$. For probabilistic timed automata, we can replace $\exists$ with $\neg\mathcal{P}_{<1}[\cdot]$, i.e replace 'there exists a path that reaches $(z{=}1) \wedge X$' with 'there exists an adversary which reaches $(z{=}1) \wedge X$ with probability 1'. Following this approach, the algorithm for calculating the set of non-zeno states is given in Figure 6. A probabilistic timed automata is then non-zeno if and only if the algorithm $\mathsf{NonZeno}$ returns the set of symbolic states $\llbracket\mathtt{true}\rrbracket$. Formally, we have the following proposition.

Proposition 17 *A probabilistic timed automaton* PTA *is non-zeno if and only if* $\{(l, inv(l)) \mid l \in L\}$ *equals the fixpoint* $\nu X.(z.\neg\mathcal{P}_{<1}[\mathtt{true}\ \mathcal{U}\ ((z{=}1) \wedge X)])$.

Proof. Consider any probabilistic timed automata PTA and suppose that $\mathsf{PS}_{\mathsf{PTA}} = (S, Steps, \mathcal{L})$ is the corresponding timed probabilistic system. To ease notation we let:

$$S_{\mathrm{nz}} = \left\{ s \in S \;\middle|\; Prob^{A}_{s}\{\omega \in Path^{A}_{ful}(s) \mid \omega \text{ is divergent}\} = 1 \text{ for some adversary } A \right\}.$$

We prove the proposition by showing that:

1. the set S_{nz} is a fixpoint of $G_{\mathrm{nz}}(\cdot)$;
2. if $G_{\mathrm{nz}}(Y) = Y$, then $Y \subseteq S_{\mathrm{nz}}$

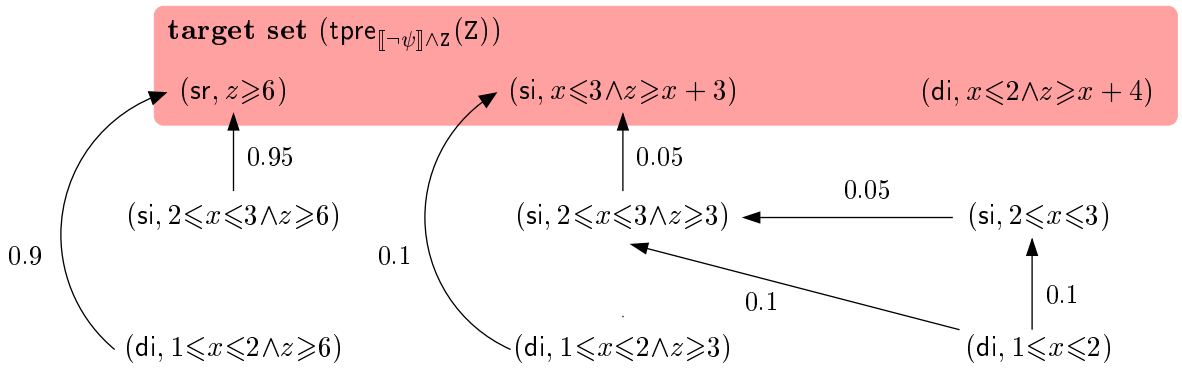

Figure 7: Probabilistic system PS generated by the algorithm MaxU

where $G_{\mathrm{nz}}(X) = z.\neg\mathcal{P}_{\geqslant 1}[\mathtt{true}\ \mathcal{U}\ (z{=}1) \wedge X]$. First, $S_{\mathrm{nz}} \subseteq G_{\mathrm{nz}}(S_{\mathrm{nz}})$, and therefore to prove S_{nz} is a fixpoint of G_{nz} it remains to show that $S_{\mathrm{nz}} \supseteq G_{\mathrm{nz}}(S_{\mathrm{nz}})$. Considering any $s \in G_{\mathrm{nz}}(S_{\mathrm{nz}})$, by definition of G_{nz} there exists an adversary A under which, with probability 1, from s one reaches a state in S_{nz} after 1 time unit. Therefore considering the adversary which behaves as A except that when a state in S_{nz} is reached, and in such a case the adversary lets time diverge with probability 1 (such choices exists by the definition of S_{nz}). It follows that, under this adversary, time diverges from s with probability 1, and hence $s \in S_{\mathrm{nz}}$ as required.

It therefore remains to show that, if $G_{\mathrm{nz}}(Y) = Y$, then $Y \subseteq S_{\mathrm{nz}}$ which we prove by contradiction. Therefore, suppose that there exists Y such that $G_{\mathrm{nz}}(Y) = Y$ and $Y \supset S_{\mathrm{nz}}$. Now, by definition of S_{nz}, if $s \in Y \setminus S_{\mathrm{nz}}$ there does not exist an adversary for which time diverges from s with probability 1. However, since $G_{\mathrm{nz}}(Y) = Y$, there exists an adversary for which with probability 1 one reaches a state in Y after 1 time unit. Iterating this fact, we have that for any $n \in \mathbb{N}$, there exists an adversary which with probability 1 reaches a state in Y after n time units. Therefore $s \in S_{\mathrm{nz}}$ which is a contradiction. □

Similarly to [HNSY94], the algorithm can be used to convert a 'zeno' probabilistic timed automaton into a non-zeno automaton by strengthening invariants. More precisely, supposing NonZeno returns Z, we can construct a new invariant condition by letting $inv_{\mathrm{nz}}(l) = \zeta^l_{\mathsf{Z}}$ for each location l of the automaton under study.

4.5 Example

We now return to the PTA in Figure 1 and verify the property $z.\mathcal{P}_{>\lambda}[\phi\ \mathcal{U}\ \psi]$, where $\phi = \mathtt{true}$ and $\psi = (\mathsf{sr} \wedge z{<}6)$, which involves computing the set of states for which minimal probability of a message being correctly delivered before 6 time units have elapsed is greater than λ. In particular, we consider this minimum probability when starting from the location di with the clock x equal to 0. In particular, we consider the minimum probability of correctly delivering before 6 time units have elapsed starting from the location di with the clock x equal to 0. In this example, we do not distinguish between the name of a location and the atomic proposition with which it is labelled. According to our methodology, the set of states satisfying $\mathcal{P}_{>\lambda}[\phi\ \mathcal{U}\ \psi]$ is given by the following set of symbolic states:

$$[\![\mathtt{true}]\!] \setminus \mathsf{MaxU}([\![\neg\psi]\!], \mathsf{MaxV}_{\geqslant 1}(c, [\![\neg\phi]\!], [\![\neg\psi]\!]), \geqslant 1{-}\lambda)\,.$$

Next, applying $\mathsf{MaxU}([\![\neg\psi]\!], \mathsf{Z}, \geqslant 1-\lambda)$ returns the probabilistic system PS given in Figure 7 (for details on the computations performed by MaxU see Appendix D), from which we find that, using Proposition 14, starting from di with x equal to 0, the maximum probability of satisfying $\neg\psi\,\mathcal{U}\,(\neg\mathcal{P}_{<1}[\neg\phi\,\mathcal{V}\,\neg\psi])$ is 0.005 (corresponding to the maximum probability for $(\mathsf{di}, 1\leqslant x\leqslant 2)$ in PS). Therefore, using Proposition 16, starting from di with x equal to 0, the minimum probability of correctly delivering before 6 time units have elapsed is $1-0.005 = 0.995$.

5 Conclusions

We have presented the theoretical foundations for the symbolic model-checking of probabilistic timed automata and PTCTL. For quantitative formulas, our algorithm is expensive, as, in the worst case, the MaxU algorithm constructs a powerset of the region graph, which itself is exponential in the largest constant used in zones and number of clocks. However, we expect this case to arise rarely in practice. Note that we do not construct a partition of the state space, but rather a set of overlapping symbolic states to avoid potentially expensive disjunction operations on zones within MaxU. Future work will address the implementation of the presented algorithms, and adaptations to probabilistic polyhedral hybrid automata and symbolic probabilistic systems [KNS01].

Finally, observe that many of the results are relevant to other verification methods for probabilistic timed automata, for example algorithms based on a *quantitative predecessor* operation [dA03]; in particular, the key result on nesting a qualitative operator inside a quantitative operator to deal with time divergence when computing minimum probabilities also holds in this context.

References

[ACD93] R. Alur, C. Courcoubetis, and D. Dill. Model checking in dense real time. *Information and Computation*, 104(1):2–34, 1993.

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

[BCM+90] J. Burch, E. Clarke, K. McMillan, D. Dill, and J. Hwang. Symbolic model checking: 10^{20} states and beyond. In *Proc. 5th Annual IEEE Symposium on Logic in Computer Science (LICS'90)*, pages 428–439. IEEE, 1990.

[BdA95] A. Bianco and L. de Alfaro. Model checking of probabilistic and nondeterministic systems. In P. Thiagarajan, editor, *Proc. 15th Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 1026 of *Lecture Notes in Comuter Science*, pages 499–513, 1995.

[BDL+01] G. Behrmann, A. David, K. Larsen, O. Möller, P. Pettersson, and W. Yi. Uppaal - present and future. In *Proceedings of the 40th IEEE Conference on Decision and Control (CDC'2001)*, volume 3, pages 2881–2886. IEEE Computer Society Press, 2001.

[BK98] C. Baier and M. Kwiatkowska. Model checking for a probabilistic branching time logic with fairness. *Distributed Computing*, 11(3):125–155, 1998.

[CGP99] E. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 1999.

[dA97] L. de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, 1997.

[dA03] L. de Alfaro. Quantitative verification and control via the mu-calculus. In *Proc. 14th International Conference on Concurrency Theory (CONCUR 2003)*, volume 2761 of *Lecture Notes in Comuter Science*, pages 103–127. Springer-Verlag, 2003.

[Der70] C. Derman. *Finite-State Markovian Decision Processes*. New York: Academic Press, 1970.

[DOTY96] C. Daws, A. Olivero, S. Tripakis, and S. Yovine. The tool Kronos. In R. Alur and E. Sontag T. Henzinger, editors, *Hybrid Systems III: Verification and Control*, volume 1066 of *Lecture Notes in Comuter Science*, pages 208–219. Springer-Verlag, 1996.

[HJ94] H. Hansson and B. Jonsson. A logic for reasoning about time and reliability. *Formal Aspects of Computing*, 6(4):512–535, 1994.

[HNSY94] T. Henzinger, X. Nicollin, J. Sifakis, and S. Yovine. Symbolic model checking for real-time systems. *Information and Computation*, 111(2):193–244, 1994.

[HSP83] S. Hart, M. Sharir, and A. Pnueli. Termination of probabilistic concurrent programs. *ACM Transactions on Programming Languages and Systems*, 5(3):356–380, 1983.

[KNPS03] M. Kwiatkowska, G. Norman, D. Parker, and J. Sproston. Performance analysis of probabilistic timed automata using digital clocks. In K. Larsen and P. Niebert, editors, *Proc. Formal Modeling and Analysis of Timed Systems (FORMATS'03)*, Lecture Notes in Comuter Science. Springer-Verlag, 2003. To appear.

[KNS01] M. Kwiatkowska, G. Norman, and J. Sproston. Symbolic computation of maximal probabilistic reachability. In K. Larsen and M. Nielsen, editors, *Proc. 13th International Conference on Concurrency Theory (CONCUR'01)*, volume 2154 of *Lecture Notes in Comuter Science*, pages 169–183. Springer-Verlag, 2001.

[KNS02] M. Kwiatkowska, G. Norman, and J. Sproston. Probabilistic model checking of the IEEE 802.11 wireless local area network protocol. In H. Hermanns and R. Segala, editors, *Proc. 2nd Joint International Workshop on Process Algebra and Probabilistic Methods, Performance Modeling and Verification (PAPM/PROBMIV'02)*, volume 2399 of *Lecture Notes in Comuter Science*, pages 169–187. Springer-Verlag, 2002.

[KNS03] M. Kwiatkowska, G. Norman, and J. Sproston. Probabilistic model checking of deadline properties in the IEEE 1394 FireWire root contention protocol. *Formal Aspects of Computing*, 14:295–318, 2003.

[KNSS02] M. Kwiatkowska, G. Norman, R. Segala, and J. Sproston. Automatic verification of real-time systems with discrete probability distributions. *Theoretical Computer Science*, 282:101–150, 2002.

[KSK76] J. Kemeny, J. Snell, and A. Knapp. *Denumerable Markov Chains*. Springer-Verlag, 2nd edition, 1976.

[Pnu83] A. Pnueli. On the extremely fair treatment of probabilstic algorithms. In *Proc. 15th Annual ACM Symposium on Theory of Computing*, pages 278–290, 1983.

[Seg95] R. Segala. *Modelling and Verification of Randomized Distributed Real Time Systems*. PhD thesis, Massachusetts Institute of Technology, 1995.

[Tri98] S. Tripakis. *L'Analyse Formelle des Systèmes Temporisés en Pratique*. PhD thesis, Université Joseph Fourier, 1998.

A Proof of Proposition 14

Before we give the proof we require a number of definitions and lemmas. First, for any adversary A of a probabilistic timed automaton PTA we introduce the sequence of functions $(\mathbf{U}^A_n)_{n\in\mathbb{N}}$. Intuitively, for $s,\mathcal{E} \in S\times\mathbb{R}^{|\mathcal{Z}|}$, the value $\mathbf{U}^A_n(\phi,\psi,s,\mathcal{E})$ equals the probability of reaching from $s,\mathcal{E}$, under the adversary A, a state which satisfies ψ in *at most* n discrete transitions, while passing through only ϕ states. Since adversaries can choose on the basis of history, we define $\mathbf{U}^A_n$ over paths, then restrict to the case of states (paths of length 0).

Definition 18 *Let* PTA *be a probabilistic timed automaton and* TPS *the corresponding timed probabilistic system. For any* PTCTL *formulae* ϕ,ψ*, adversary* $A \in Adv_{\mathsf{TPS}}$*,* $\mathcal{E}\in\mathbb{R}^{|\mathcal{Z}|}$ *and finite path* $\omega \in Path^A_{fin}$ *where* $last(\omega)=(l,v)$ *and* $A(\omega)=(t,\mu)$*:*

- *if* $t \geqslant 0$*,* $\mu = \mu_{(l,v+t)}$ *and there exists* $t' \leqslant t$ *such that* $(l,v+t'),\mathcal{E}+t' \models \psi$ *and* $(l,v+t''),\mathcal{E}+t'' \models \phi\vee\psi$ *for all* $t''\leqslant t'$*, then* $\mathbf{U}^A_0(\phi,\psi,(l,v),\mathcal{E})=1$*;*
- *else if* $t=0$ *and* $(l,v),\mathcal{E}\models\psi$*, then* $\mathbf{U}^A_0(\phi,\psi,\omega,\mathcal{E})=1$*;*
- *otherwise,* $\mathbf{U}^A_0(\phi,\psi,\omega,\mathcal{E})=0$*.*

and for any $n\geqslant 0$*:*

- *if* $t \geqslant 0$*,* $\mu = \mu_{(l,v+t)}$ *and there exists* $t' \leqslant t$ *such that* $(l,v+t'),\mathcal{E}+t' \models \psi$ *and* $(l,v+t''),\mathcal{E}+t'' \models \phi\vee\psi$ *for all* $t''\leqslant t'$*, then* $\mathbf{U}^A_{n+1}(\phi,\psi,\omega,\mathcal{E})=1$*;*
- *else if* $t\geqslant 0$*,* $\mu=\mu_{(l,v+t)}$ *and* $(l,v+t'),\mathcal{E}+t'\models\phi\wedge\neg\psi$ *for all* $t'\leqslant t$*, then*

$$\mathbf{U}^A_{n+1}(\phi,\psi,\omega,\mathcal{E}) = \mathbf{U}^A_n(\phi,\psi,\omega\xrightarrow{t,\mu}(l,v+t),\mathcal{E}+t)$$

- *else if* $t=0$ *and* $(l,v),\mathcal{E}\models\phi\wedge\neg\psi$*, then*

$$\mathbf{U}^A_{n+1}(\phi,\psi,\omega,\mathcal{E}) = \sum_{(l',v')\in Q}\mu(l',v')\cdot\mathbf{U}^A_n(\phi,\psi,\omega\xrightarrow{t,p}(l',v'),\mathcal{E})$$

- *otherwise, let* $\mathbf{U}^A_{n+1}(\phi,\psi,\omega,\mathcal{E})=0$*.*

Lemma 19 *For any probabilistic timed automaton* PTA*, corresponding timed probabilistic system* PS*,* $A\in Adv_{\mathsf{TPS}}$*,* $s,\mathcal{E}\in S\times\mathbb{R}^{|\mathcal{Z}|}$ *and* PTCTL *formulae* ϕ *and* ψ*: the sequence* $\langle\mathbf{U}^A_n(\phi,\psi,s,\mathcal{E})\rangle_{n\in\mathbb{N}}$ *is an increasing in* $[0,1]$ *and converges to* $p^A_{s,\mathcal{E}}(\phi\,\mathcal{U}\,\psi)$*.*

Next, for any adversary B of a probabilistic system PS, we define a sequence of functions $(\mathbf{R}^B_n)_{n\in\mathbb{N}}$, where $\mathbf{R}^B_n(F,s)$ equals the probability, of reaching, from s under the adversary B, a state in F in at most n steps.

Definition 20 *Let* $\mathsf{PS}=(S,Steps,\mathcal{L}')$ *be a probabilistic system* PS*. For any subset of states* F*, adversary* $B\in Adv_{\mathsf{PS}}$ *and* $\pi\in Path^B_{fin}$*, if* $last(\pi)=s$ *and* $B(\pi)=\rho$*, let:*

$$\mathbf{R}^B_0(F,\pi)=\begin{cases}1 & \text{if } s\in F\\ 0 & \text{otherwise}\end{cases}$$

and for any $n\geqslant 0$*:*

$$\mathbf{R}^B_{n+1}(F,\pi) \;=\; \begin{cases}1 & \text{if } s\in F\\ \sum_{s'\in S}\rho(s')\cdot\mathbf{R}^B_n(F,\pi\xrightarrow{\rho}s') & \text{otherwise.}\end{cases}$$

Lemma 21 *For any probabilistic system* $\mathsf{PS} = (S, Steps, \mathcal{L})$*, adversary* $B \in Adv_{\mathsf{PS}}$*, state* $s \in S$ *and subset of states* $F \subseteq S$*:* $\langle \mathbf{R}^B_n(F, s)\rangle_{n\in\mathbb{N}}$ *is an increasing sequence in* $[0, 1]$ *which converges to* $MaxProbReach(s, F)$*.*

We are now in a position to prove Proposition 14 which states:

For any probabilistic timed automaton PTA *and* PTCTL *formula* $\mathcal{P}_{\lesssim\lambda}[\phi\ \mathcal{U}\ \psi]$*, if* $\mathsf{PS} = (\mathsf{Z}, Steps)$ *is the probabilistic system generated by* $\mathsf{MaxU}([\![\phi]\!], [\![\psi]\!], \gtrsim \lambda)$ *then for any* $s, \mathcal{E} \in S \times \mathbb{R}^{|\mathcal{Z}|}$*:*

- $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi){>}0$ *if and only if* $s, \mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathsf{Z})$*;*
- *if* $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi){>}0$*, then* $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi)$ *equals*

$$\max\left\{ MaxProbReach(\mathsf{z}, \mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!]) \;\middle|\; \mathsf{z} \in \mathsf{Z} \text{ and } s, \mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathsf{z}) \right\}.$$

Proof of Proposition 14. We split the proof into proving a sequence of properties: (a), (b), (c) and (d). First consider the following:

(a) If $(\mathsf{z}, (X, l'), \mathsf{z}') \in E_{(l,g,p)}$ and $(l, v), \mathcal{E} \in \mathsf{z}$, then $(l, v), \mathcal{E} \models \phi \vee \psi$, $v \triangleright inv(l)$, $v \triangleright g$, and $(l', v[X{:=}0]), \mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathsf{z}')$.

The result follows from the definition of dpre and tpre. Next we prove that the following condition holds.

(b) For any $s, \mathcal{E} \in S{\times}\mathbb{R}^{|\mathcal{Z}|}$, $p^{\max}_{s,\mathcal{E}}(\phi\ \mathcal{U}\ \psi) > 0$ if and only if $s, \mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathsf{Z})$.

The proof follows by induction on the shortest path to reach a state satisfying ψ passing through only ϕ states.

The main step in the proof involves showing the following correspondence between the values of $\mathbf{U}^A_n$ for $A \in Adv_{\mathsf{TPS}}$ and $\mathbf{R}^B_n$ for $B \in Adv_{\mathsf{PS}_\mathsf{z}}$ for all $n \in \mathbb{N}$.

(c) For any $B \in Adv_{\mathsf{PS}_\mathsf{z}}$, $\mathsf{z} \in \mathsf{Z}$ and $(l, v), \mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathsf{z})$, there exists $A \in Adv_{\mathsf{TPS}}$ such that: $\mathbf{U}^A_{2n}(\phi, \psi, (l, v), \mathcal{E}) \geqslant \mathbf{R}^B_n(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!], \mathsf{z})$.

(d) For any $A \in Adv_{\mathsf{TPS}}$ and $(l, v), \mathcal{E} \in S{\times}\mathbb{R}^{|\mathcal{Z}|}$, if $p^{\max}_{(l,v),\mathcal{E}}(\phi\ \mathcal{U}\ \psi) > 0$, then $\mathsf{z} \in \mathsf{Z}$ and $B \in Adv_{\mathsf{PS}_\mathsf{z}}$ such that $(l, v), \mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathsf{z})$ and $\mathbf{R}^B_n(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!], \mathsf{z}) \geqslant \mathbf{U}^A_n(\phi, \psi, (l, v), \mathcal{E})$.

It follows from (b), Lemma 19 and Lemma 21 that to prove Proposition 14 it is sufficient to show that (c) and (d) hold. We now prove (c) and (d) by induction on $n \in \mathbb{N}$.

Proof of (c). Consider any $B \in Adv_{\mathsf{PS}_\mathsf{z}}$, $\mathsf{z} \in \mathsf{Z}$ and $(l, v), \mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathsf{z})$. If $n = 0$, then by Definition 20 we have the following two cases to consider.

- If $\mathbf{R}^B_0(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!], \mathsf{z}) = 1$, then $\mathsf{z} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!]$ and by definition of tpre there exists $t \in \mathbb{R}$ such that $(l, v{+}t), \mathcal{E}{+}t \models \psi$ and $(l, v{+}t'), \mathcal{E}{+}t' \models \phi \vee \psi$ for all $t' \leqslant t'$, therefore letting A be the adversary such that $A(l, v) = (t, \delta_{(l,v+t)})$, it follows that:

$$\mathbf{U}^A_{2\cdot 0}(\phi, \psi, (l, v), \mathcal{E}) = \mathbf{U}^A_0(\phi, \psi, (l, v), \mathcal{E}) = 1 = \mathbf{R}^B_0(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!], \mathsf{z}).$$

- If $\mathbf{R}_0^B(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z})=0$, then choosing any $A\in Adv_{\mathsf{TPS}}$ we have:

$$\mathbf{U}_{2\cdot 0}^A(\phi,\psi,(l,v),\mathcal{E})=\mathbf{U}_0^A(\phi,\psi,(l,v),\mathcal{E})\geqslant 0=\mathbf{R}_0^B(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z})\,.$$

Next, suppose that (c) holds for some $n\in\mathbb{N}$ and consider $\mathbf{U}_{2(n+1)}^A(\phi,\psi,(l,v),\mathcal{E})$. If $\mathtt{z}\in\mathsf{tpre}_{[\![\phi\vee\psi]\!]}([\![\psi]\!])$ the result follows as in the case for $n=0$. We are therefore left to consider the case when $\mathtt{z}\notin\mathsf{tpre}_{[\![\phi\vee\psi]\!]}([\![\psi]\!])$.

By construction, $B(\mathtt{z})=\rho$ for some $(\mathtt{z},\rho)\in \mathit{Steps}$, and from the construction of $\mathsf{PS}_{\mathtt{Z}}$, there exists $(l,g,p)\in \mathit{prob}$ and set of edges $E_\rho\subseteq E_{(l,g,p)}$ such that for any $\mathtt{z}'\in\mathtt{Z}$:

$$\rho(\mathtt{z}')=\sum_{(\mathtt{z},(X,l'),\mathtt{z}')\in E_\rho} p(X,l').$$

If B' is the adversary such that $\mathbf{R}_n^{B'}(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z}')=\mathbf{R}_n^B(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z}\xrightarrow{\rho}\mathtt{z}')$, then from Definition 20 and the construction of ρ we have:

$$\begin{aligned}\mathbf{R}_{n+1}^B(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z}) &= \sum_{\mathtt{z}'\in\mathtt{Z}}\rho(\mathtt{z}')\cdot\mathbf{R}_n^{B'}(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z}')\\ &= \sum_{(\mathtt{z},(X,l'),\mathtt{z}')\in E_\rho} p(X,l')\cdot\mathbf{R}_{n+1}^B(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z}')\,. \qquad (12)\end{aligned}$$

Since $(l,v),\mathcal{E}\in\mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathtt{z})$, it follows that there exists $t\in\mathbb{R}$ such that $(l,v+t),\mathcal{E}+t\in\mathtt{z}$ and $((l,v),(t,\delta_{(l,v+t)}))\in \mathit{prob}$. Now, for any $(\mathtt{z},(X,l'),\mathtt{z}')\in E_\rho$ using (a) we have that $(l',(v+t)[X:=0]),\mathcal{E}+t\in\mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathtt{z}')$. Therefore, by induction, for any $e=(\mathtt{z},(X,l'),\mathtt{z}')\in E_\rho$ there exists $A_e\in Adv_{\mathsf{TPS}}$ such that:

$$\mathbf{U}_{2n}^{A_e}(\phi,\psi,(l',(v+t)[X:=0]),\mathcal{E}+t)\geqslant\mathbf{R}_n^{B'}(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathtt{z}')\,. \qquad (13)$$

Let $A\in Adv_{\mathsf{TPS}}$ be the adversary such that

- $A(l,v)=(t,\delta_{(l,v+t)})$;
- $A\left((l,v)\xrightarrow{t,\delta_{(l,v+t)}}(l,v+t)\right)=(0,\mu_p)$ where for any $(l',v')\in S$:

$$\mu_p(l',v')=\sum_{\substack{X\subseteq\mathcal{X}\ \&\\ v'=(v+t)[X:=0]}} p(X,l');$$

- for any $e=(\mathtt{z},(X,l'),\mathtt{z}')\in E_\rho$:

$$A\left((l,v)\xrightarrow{t,\delta_{(l,v+t)}}(l,v+t[X:=0])\xrightarrow{0,\mu}(l',(v+t)[X:=0])\right)=A^e(l',(v+t)[X:=0])\,.$$

Note that, the existence of the above distributions follows from Definition 7. It then follows from Definition 18 and the construction of A that:

$$\begin{aligned}
\mathbf{U}^{A}_{2(n+1)}(\phi,\psi,(l,v),\mathcal{E}) &= \sum_{(X,l')\in \mathsf{edges}(p)} p(X,l')\cdot \mathbf{U}^{A^{(X,l')}}_{2n}(\phi,\psi,(l',(v{+}t)[X{:=}0]),\mathcal{E}{+}t)\\
&\geqslant \sum_{(\mathbf{z},(X,l'),\mathbf{z}')\in E_\rho} p(X,l')\cdot \mathbf{U}^{A^{(X,l')}}_{2n}(\phi,\psi,(l',(v{+}t)[X{:=}0]),\mathcal{E}{+}t) \qquad \text{by definition of } E_\rho\\
&\geqslant \sum_{(\mathbf{z},(X,l'),\mathbf{z}')\in E_\rho} p(X,l')\cdot \mathbf{R}^{B}_{n}(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathbf{z}') \qquad \text{by (13)}\\
&= \mathbf{R}^{B}_{n+1}(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathbf{z}) \qquad \text{by (12)}
\end{aligned}$$

and since $\mathbf{z}$ and B are arbitrary, **(c)** holds by induction.

Proof of (d). Consider any $A \in Adv_{\mathsf{TPS}}$ and $(l,v),\mathcal{E} \in S{\times}\mathbb{R}^{|\mathcal{Z}|}$ such that $p^{\max}_{(l,v),\mathcal{E}}(\phi\,\mathcal{U}\,\psi) > 0$. If $n = 0$, then by Definition 18 we have the following two cases to consider.

- If $\mathbf{U}^{A}_{0}(\phi,\psi,(l,v),\mathcal{E}) = 1$, then there exists $t \in \mathbb{R}$ such that $(l,v{+}t),\mathcal{E}{+}t \models \psi$ and $(l,v{+}t'),\mathcal{E}{+}t' \models \phi\vee\psi$ for all $t' \leqslant t$. By definition of tpre it follows that $(l,v),\mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!]$, and hence

$$\mathbf{R}^{B}_{0}(\mathsf{tpre}_{[\![\phi\vee\psi]\!]}[\![\psi]\!],\mathbf{z}) = 1 = \mathbf{U}^{A}_{0}(\phi,\psi,(l,v),\mathcal{E})\,.$$

- If $\mathbf{U}^{A}_{0}(\phi,\psi,(l,v),\mathcal{E}) = 0$, then the result follows by choosing any $B \in Adv_{\mathsf{PS}_\mathbf{z}}$ and $\mathbf{z} \in Z$ such that $(l,v),\mathcal{E} \in \mathsf{tpre}_{[\![\phi\vee\psi]\!]}(\mathbf{z})$ (the existence of $\mathbf{z}$ follows from (b)).

Now suppose that (d) holds from some $n \in \mathbb{N}$. If $\mathbf{U}^{A}_{n+1}(\phi,\psi,(l,v),\mathcal{E}) = 0$, then the result follows as in the case when $n = 0$. It therefore remains to consider the case when $\mathbf{U}^{A}_{n+1}(\phi,\psi,(l,v),\mathcal{E}) > 0$ and from Definition 18 we have the following cases to consider.

- If $A(l,v) = (t,\delta_{(l,v+t)})$ such that $(l,v{+}t),\mathcal{E}{+}t \models \psi$ and $(l,v{+}t'),\mathcal{E}{+}t' \models \phi\vee\psi$ for all $t' \leqslant t$ the result follows similarly to when $n = 0$.

- If $A(l,v) = (t,\delta_{(l,v+t)})$ such that $(l,v{+}t'),\mathcal{E}{+}t' \models \phi\wedge\neg\psi$ for all $t' \leqslant t$, then

$$\mathbf{U}^{A}_{n+1}(\phi,\psi,(l,v),\mathcal{E}) = \mathbf{U}^{A}_{n}(\phi,\psi,(l,v{+}t),\mathcal{E}{+}t)\,.$$

  and the result follows by induction and Lemma 21.

- If $A(l,v) = (0,\mu)$, then by Definition 18 we have:

$$\mathbf{U}^{A}_{n+1}(\phi,\psi,(l,v),\mathcal{E}) = \sum_{(l',v')\in S} \mu(l',v')\cdot \mathbf{U}^{A}_{n}(\phi,\psi,(l,v) \xrightarrow{t,p} (l',v'),\mathcal{E})$$

  and $(l,v),\mathcal{E} \models \phi\wedge\psi$. Now, from Definition 7, there exists $(l,g,p) \in \mathit{prob}$ such that $v \triangleright g$ and for any $(l',v') \in S$:

$$\mu(l',v') = \sum_{\substack{X\subseteq\mathcal{X}\ \&\\ v'=v[X:=0]}} p(X,l')\,.$$

Letting $A^{(l',X)}$ be the adversary such that $A^{(l',X)}(l', v[X{:=}0]) = A((l,v) \xrightarrow{0,\mu} (l', v[X{:=}0]))$, it follows from above that:

$$\mathbf{U}^{A}_{n+1}(\phi, \psi, (l,v), \mathcal{E}) \quad = \sum_{(X,l') \in \mathsf{support}(p)} p(X, l') \cdot \mathbf{U}^{A^{(l',X)}}_{n}(\phi, \psi, (l', v[X{:=}0]), \mathcal{E}) \quad (14)$$

Now consider any $(l', X) \in \mathsf{support}(p)$ such that $\mathbf{U}^{A^{(l',X)}}_{n}(l', v[X{:=}0]), \mathcal{E}) > 0$, then by definition $(l, g, p, X, l') \in \mathsf{edges}$ and by induction and Lemma 19 there exists $(l', \zeta'_{l',X}) \in \mathtt{Z}$ and adversary $B^{(l',X)}$ such that

$$\mathbf{R}^{B^{(l',X)}}_{n}(\mathsf{tpre}_{[\![\phi \vee \psi]\!]}[\![\psi]\!], (l', \zeta'_{l',X})) \geqslant \mathbf{U}^{A^{(l',X)}}_{n}(\phi, \psi, (l', v[X{:=}0]), \mathcal{E}) \quad (15)$$

and $(l', v[X{:=}0]), \mathcal{E} \in \mathsf{tpre}_{[\![\phi \vee \psi]\!]}(l', \zeta'_{l',X})$. Letting:

$$(l, \zeta_{l',X}) = \mathsf{dpre}((l, g, p, X, l'), \mathsf{tpre}_{[\![\phi \vee \psi]\!]}(l', \zeta'_{l',X})),$$

since $(l,v), \mathcal{E} \models \phi \wedge \psi$, it follows that $((l, \zeta_{l',X}), (X, l'), (l', \zeta'_{l',X})) \in E_{(l,g,p)}$, $(l, \zeta_{l',X}) \in \mathtt{Z}$ and $(l,v), \mathcal{E} \in (l, \zeta_{l',X})$. Therefore, from the construction of PS setting $\mathtt{z}$ equal to:

$$\Big(l, \wedge\{\zeta_{l',X} \mid (l',X) \in \mathsf{support}(p) \text{ and } p^{\max}_{(l',v[X:=0]),\mathcal{E}}(\phi \,\mathcal{U}\, \psi) > 0\}\Big)$$

we have $\mathtt{z} \in \mathtt{Z}$ and $(l,v) \in \mathtt{z}$. Furthermore, by construction of PS there exists $(\mathtt{z}, \rho) \in \mathit{Steps}$ such that for any $\mathtt{z}' \in \mathtt{Z}$:

$$\rho(\mathtt{z}') \geqslant \sum_{\substack{(l',X) \in \mathsf{support}(p),\ \zeta' = \zeta_{l',X} \\ \&\ \mathbf{U}^{A'}_{n}(\phi,\psi,(l',v[X:=0]),\mathcal{E}) > 0}} p(X, l'). \quad (16)$$

Now, setting B to be the adversary of PS such that $B(\mathtt{z}) = \rho$ and $B(\mathtt{z} \xrightarrow{\rho} (l', \zeta'_{l',X})) = B^{(l',X)}(l', \zeta'_{l',X})$, by Definition 20 we have:

$$\begin{aligned}
&\mathbf{R}^{B}_{n+1}(\mathsf{tpre}_{[\![\phi \vee \psi]\!]}[\![\psi]\!], \mathtt{z}) = \sum_{\mathtt{z}' \in \mathtt{Z}} \rho(\mathtt{z}') \cdot \mathbf{R}^{B}_{n}(\mathsf{tpre}_{[\![\phi \vee \psi]\!]}[\![\psi]\!], \mathtt{z} \xrightarrow{\rho} \mathtt{z}') \\
&\geqslant \sum_{\substack{(l',X) \in \mathsf{support}(p)\,\& \\ \mathbf{U}^{A'}_{n}(l',v[X:=0]),\mathcal{E}) > 0}} p(s, X) \cdot \mathbf{R}^{B}_{n}(\mathsf{tpre}_{[\![\phi \vee \psi]\!]}[\![\psi]\!], \mathtt{z} \xrightarrow{\rho} (l', \zeta'_{l',X})) && \text{by (16)} \\
&= \sum_{\substack{(l',X) \in \mathsf{support}(p)\,\& \\ \mathbf{U}^{A'}_{n}(l',v[X:=0]),\mathcal{E}) > 0}} p(s, X) \cdot \mathbf{R}^{B^{(l',X)}}_{n}(\mathsf{tpre}_{[\![\phi \vee \psi]\!]}[\![\psi]\!], (l', \zeta'_{l',X})) && \text{by construction} \\
&\geqslant \sum_{\substack{(l',X) \in \mathsf{support}(p)\,\& \\ \mathbf{U}^{A'}_{n}(l',v[X:=0]),\mathcal{E}) > 0}} p(s, X) \cdot \mathbf{U}^{A^{(l',X)}}_{n}(\phi, \psi, (l', v[X{:=}0]), \mathcal{E}) && \text{by (15)} \\
&= \sum_{(l',X) \in \mathsf{support}(p)} p(s, X) \cdot \mathbf{U}^{A^{(l',X)}}_{n}(\phi, \psi, (l, v[X{:=}0]), \mathcal{E}) && \text{rearranging} \\
&= \mathbf{U}^{A}_{n+1}(\phi, \psi, (l,v), \mathcal{E}) && \text{by (14).}
\end{aligned}$$

Since these are all the cases to consider (d) holds by induction as required. □

B $\mathsf{MaxV}_{\geqslant 1}(c, [\![\mathtt{false}]\!], [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!])$

$\mathtt{Z} := [\![\mathtt{true}]\!]$
repeat

$\mathtt{Y} := \mathtt{Z}$
$\mathtt{Z} := [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, ([\![\mathtt{false}]\!] \wedge \mathtt{Y})\vee[\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}([\![\mathtt{true}]\!], [\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!]$

$\mathtt{Y} := \mathtt{Z}$
$\mathtt{Z} := [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, ([\![\mathtt{false}]\!] \wedge \mathtt{Y})\vee[\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, [\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\{(\mathsf{sr}, z\geqslant 6\vee y>c), (\mathsf{si}, x\leqslant 3\wedge(z\geqslant x+3\vee y>x+c-3)), (\mathsf{di}, x\leqslant 2\wedge(z\geqslant x+4\vee y>x+c-2))\}$
$= \{(\mathsf{sr}, z\geqslant 6), (\mathsf{si}, x\leqslant 3\wedge(z\geqslant x+3\vee x<3-c)), (\mathsf{di}, x\leqslant 2\wedge(z\geqslant x+4\vee x<2-c))\}$

$\mathtt{Y} := \mathtt{Z}$
$\mathtt{Z} := [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, ([\![\mathtt{false}]\!] \wedge \mathtt{Y})\vee[\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, [\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\{(\mathsf{sr}, z\geqslant 6\vee y>c), (\mathsf{si}, x\leqslant 3\wedge(z\geqslant x+3\vee y>c\vee y>x+2\cdot c-3)), (\mathsf{di}, x\leqslant 2\wedge(z\geqslant x+4\vee y>c\vee y>x+2\cdot c-2))\}$
$= \{(\mathsf{sr}, z\geqslant 6), (\mathsf{si}, x\leqslant 3\wedge(z\geqslant x+3\vee x<3-2\cdot c)), (\mathsf{di}, x\leqslant 2 \wedge (z\geqslant x+4\vee x<2-2\cdot c))\}$

repeating $n-2$ times such that $n\cdot c \geqslant 3$ and $(n-1)\cdot c < 3$ (which exists as $c> 0$)

$\mathtt{Y} := \mathtt{Z}$
$\mathtt{Z} := [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, ([\![\mathtt{false}]\!] \wedge \mathtt{Y})\vee[\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\mathsf{MaxU}_{\geqslant 1}(\mathtt{Y}, [\![y>c]\!])$
$= [\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge y.\{(\mathsf{sr}, z\geqslant 6\vee y>c), (\mathsf{si}, x\leqslant 3\wedge(z\geqslant x+3\vee y>c\vee y>x+n\cdot c-3)), (\mathsf{di}, x\leqslant 2\wedge(z\geqslant x+4\vee y>c\vee y>x+n\cdot c-2))\}$
$= \{(\mathsf{sr}, z\geqslant 6), (\mathsf{si}, x\leqslant 3\wedge(z\geqslant x+3\vee x<3-n\cdot c)), (\mathsf{di}, x\leqslant 2\wedge(z\geqslant x+4\vee x<2-n\cdot c))\}$
$= \{(\mathsf{sr}, z\geqslant 6), (\mathsf{si}, x\leqslant 3\wedge z\geqslant x+3), (\mathsf{di}, x\leqslant 2\wedge z\geqslant x+4)\}$

endrepeat

C $\quad \mathsf{MaxU}_{\geqslant 1}([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!], [\![y>c]\!])$

$\mathtt{Z}_0 := [\![\mathtt{true}]\!]$
repeat

$\mathtt{Y}_0 := \mathtt{Z}_0$
$\mathtt{Z}_1 := [\![\mathtt{false}]\!]$
repeat

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1)) = [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge [\![\mathtt{false}]\!])$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee [\![\mathtt{false}]\!] = [\![y>c]\!]$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1)) = [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3 \wedge y>c), (\mathsf{di}, 1\leqslant x\leqslant 2 \wedge y>c)\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge y>x+(c-3)), (\mathsf{di}, x\leqslant 2 \wedge y>x+(c-2))\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge y>x+(c-3)), (\mathsf{di}, x\leqslant 2 \wedge y>x+(c-2))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3 \wedge (y>c-3 \vee z\geqslant 6)), (\mathsf{di}, 1\leqslant x\leqslant 2 \wedge (y>c-3 \vee z\geqslant 6))\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge y>x+(c-3)), (\mathsf{di}, x\leqslant 2 \wedge y>x+(c-2))\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, (2\leqslant x\leqslant 3 \wedge (y>c-3 \vee z\geqslant 6)) \vee (x\leqslant 3 \wedge y>x+(c-3))),$
$(\mathsf{di}, (1\leqslant x\leqslant 2 \wedge (y>c-3 \vee z\geqslant 6)) \vee (x\leqslant 2 \wedge y>x+(c-2)))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3 \wedge (y>c-3 \vee z\geqslant 6)), (\mathsf{di}, 1\leqslant x\leqslant 2 \wedge (y>c-3 \vee z\geqslant 6))\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1)$
$= \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge (y>x+c-6 \vee z\geqslant x+3)), (\mathsf{di}, x\leqslant 2 \wedge (y>x+c-5 \vee z\geqslant x+4))\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge (y>x+c-6 \vee z\geqslant x+3)), (\mathsf{di}, x\leqslant 2 \wedge (y>x+c-5 \vee z\geqslant x+4))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3 \wedge (y>c-6 \vee z\geqslant 3)), (\mathsf{di}, 1\leqslant x\leqslant 2 \wedge (y>c-6 \vee z\geqslant 3))\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1)$
$= \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge (y>x+c-6 \vee z\geqslant x+3)), (\mathsf{di}, x\leqslant 2 \wedge (y>x+c-5 \vee z\geqslant x+4))\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, (2\leqslant x\leqslant 3 \wedge (y>c-6 \vee z\geqslant 3))) \vee (x\leqslant 3 \wedge (y>x+c-6 \vee z\geqslant x+3))),$
$(\mathsf{di}, (1\leqslant x\leqslant 2 \wedge (y>c-6 \vee z\geqslant 3)) \vee (x\leqslant 2 \wedge (y>x+c-5 \vee z\geqslant x+4)))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3 \wedge (y>c-6 \vee z\geqslant 3)), (\mathsf{di}, 1\leqslant x\leqslant 2 \wedge (y>c-6 \vee z\geqslant 3))\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1)$
$= \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge (y>x+c-9 \vee z\geqslant x)), (\mathsf{di}, x\leqslant 2 \wedge (y>x+c-8 \vee z\geqslant x+1))\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge (y>x+c-9 \vee z\geqslant x)), (\mathsf{di}, x\leqslant 2 \wedge (y>x+c-8 \vee z\geqslant x+1))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3), (\mathsf{di}, 1\leqslant x\leqslant 2)\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1)$
$= \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3 \wedge (y>x+c-9 \vee z\geqslant x)), (\mathsf{di}, x\leqslant 2 \wedge (y>x+c-8 \vee z\geqslant x+1))\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, (2\leqslant x\leqslant 3) \vee (x\leqslant 3 \wedge (y>x+c-9 \vee z\geqslant x))), (\mathsf{di}, (1\leqslant x\leqslant 2) \vee (x\leqslant 2 \wedge (y>x+c-8 \vee z\geqslant x+1)))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1)) = [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3), (\mathsf{di}, 1\leqslant x\leqslant 2)\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3), (\mathsf{di}, x\leqslant 2)\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3), (\mathsf{di}, x\leqslant 2)\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1)) = [\![y>c]\!] \vee ([\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \wedge \{(\mathsf{si}, 2\leqslant x\leqslant 3), (\mathsf{di}, 1\leqslant x\leqslant 2)\})$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si}\vee\mathsf{di}\vee z\geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3), (\mathsf{di}, x\leqslant 2)\}$
$= \{(\mathsf{sr}, z\geqslant 6 \vee y>c), (\mathsf{si}, x\leqslant 3), (\mathsf{di}, x\leqslant 2)\}$

endrepeat

$\mathtt{Z}_0 := \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3), (\mathsf{di}, x \leqslant 2)\}$

$\mathtt{Y}_0 := \mathtt{Z}_0$
$\mathtt{Z}_1 := [\![\mathtt{false}]\!]$
repeat

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1)) = [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge [\![\mathtt{false}]\!])$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee [\![\mathtt{false}]\!] = [\![y>c]\!]$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \{(\mathsf{si}, 2 \leqslant x \leqslant 3 \wedge y > c) \vee (\mathsf{di}, 1 \leqslant x \leqslant 2 \wedge y > c)\}$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge y > x + c - 3), (\mathsf{di}, x \leqslant 2 \wedge y > x + c - 2)\}$
$= \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge y > x + c - 3), (\mathsf{di}, x \leqslant 2 \wedge y > x + c - 2)\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \{(\mathsf{si}, 2 \leqslant x \leqslant 3 \wedge (z \geqslant 6 \vee y > c)) \vee (\mathsf{di}, 1 \leqslant x \leqslant 2 \wedge (z \geqslant 6 \vee y > c))\}$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge y > x + c - 3), (\mathsf{di}, x \leqslant 2 \wedge y > x + c - 2)\}$
$= \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, (2 \leqslant x \leqslant 3 \wedge z \geqslant 6) \vee (x \leqslant 3 \wedge y > x + c - 3)), (\mathsf{di}, (1 \leqslant x \leqslant 2 \wedge z \geqslant 6) \vee (x \leqslant 2 \wedge y > x + c - 2))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \{(\mathsf{si}, 2 \leqslant x \leqslant 3 \wedge (z \geqslant 6 \vee y > c)) \vee (\mathsf{di}, 1 \leqslant x \leqslant 2 \wedge (z \geqslant 6 \vee y > c))\}$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1)$
$= \mathtt{Z}_1 \vee \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$
$= \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \{(\mathsf{si}, 2 \leqslant x \leqslant 3 \wedge (z \geqslant 6 \vee y > c)) \vee (\mathsf{di}, 1 \leqslant x \leqslant 2 \wedge (z \geqslant 6 \vee y > c))\}$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1)$
$= \mathtt{Z}_1 \vee \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$
$= \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$

end repeat

$\mathtt{Z}_0 := \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$

$\mathtt{Y}_0 := \mathtt{Z}_0$
$\mathtt{Z}_1 := [\![\mathtt{false}]\!]$
repeat

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1)) = [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge [\![\mathtt{false}]\!])$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee [\![\mathtt{false}]\!] = [\![y>c]\!]$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \{(\mathsf{si}, 2 \leqslant x \leqslant 3 \wedge y > c) \vee (\mathsf{di}, 1 \leqslant x \leqslant 2 \wedge y > c)\}$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge y > x + c - 3), (\mathsf{di}, x \leqslant 2 \wedge y > x + c - 2)\}$
$= \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge y > x + c - 3), (\mathsf{di}, x \leqslant 2 \wedge y > x + c - 2)\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \{(\mathsf{si}, 2 \leqslant x \leqslant 3 \wedge (z \geqslant 6 \vee y > c)) \vee (\mathsf{di}, 1 \leqslant x \leqslant 2 \wedge (z \geqslant 6 \vee y > c))\}$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1) = \mathtt{Z}_1 \vee \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$
$= \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, (2 \leqslant x \leqslant 3 \wedge z \geqslant 6) \vee (x \leqslant 3 \wedge y > x + c - 3)), (\mathsf{di}, (1 \leqslant x \leqslant 2 \wedge z \geqslant 6) \vee (x \leqslant 2 \wedge y > x + c - 2))\}$

$\mathtt{Y}_1 := \mathtt{Z}_1$
$\mathtt{Z}_1 := [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \mathsf{pre1}(\mathtt{Y}_0, \mathtt{Y}_1))$
$= [\![y>c]\!] \vee ([\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \wedge \{(\mathsf{si}, 2 \leqslant x \leqslant 3 \wedge (z \geqslant 6 \vee y > c)) \vee (\mathsf{di}, 1 \leqslant x \leqslant 2 \wedge (z \geqslant 6 \vee y > c))\}$
$\mathtt{Z}_1 := \mathtt{Z}_1 \vee \mathsf{tpre}_{[\![\mathsf{si} \vee \mathsf{di} \vee z \geqslant 6]\!] \vee [\![y>c]\!]}(\mathtt{Y}_0 \wedge \mathtt{Y}_1)$
$= \mathtt{Z}_1 \vee \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$
$= \{(\mathsf{sr}, z \geqslant 6 \vee y > c), (\mathsf{si}, x \leqslant 3 \wedge (z \geqslant x + 3 \vee y > x + c - 3)), (\mathsf{di}, x \leqslant 2 \wedge (z \geqslant x + 4 \vee y > x + c - 2))\}$

end repeat

$\mathtt{Z}_0 := \mathtt{Z}_1$

end repeat

D $\mathsf{MaxU}([\![\mathsf{si} \vee z \geqslant 6]\!], \{(\mathsf{sr}, z \geqslant 6), (\mathsf{si}, x \leqslant 3 \wedge z \geqslant x+3), (\mathsf{di}, x \leqslant 2 \wedge z \geqslant x+4)\})$