A Decidable Probability Logic for Timed and Untimed Probabilistic Systems

Lanotte, R. and Beauquier, D.

Technical Report TR-2005-05
A Decidable Probability Logic for Timed an Untimed Probabilistic Systems *

Ruggero Lanotte\(^1\) and Danièle Beauquier\(^2\)

\(^1\) Dipartimento di Informatica, Università di Pisa via Buonarroti 2, 56127, Pisa, Italy
\(^2\) LACL, Dept. of Informatics, Université de Paris 12, 61 Avenue du Générale de Gaulle, 94010, Créteil Paris, France

Abstract. In this paper we extend the predicate logic introduced in [BRS02] in order to deal with Semi-Markov Processes. We prove that with respect to qualitative probabilistic properties, model checking is decidable for this logic applied to Semi-Markov Processes. Furthermore we apply our logic to Probabilistic Timed Automata considering classical and urgent semantics, and considering also predicates on clock. We prove that results on Semi Markov Processes are extendible to Probabilistic Timed Automata for both the two semantics considered. Moreover, we prove that results for Markov Processes shown in [BRS02] are extendible to Probabilistic Timed Automata where urgent semantics is considered.

1 Introduction

This work is in keeping with the general pattern of specification and verification of real time systems. Among the numerous existing frameworks within which a formal analysis can be carried out, the timed automata formalism [AD94] has received much attention.

Classically the timed properties to verify are expressed in terms of temporal logics. Moreover during the last years, a new feature has become of interest, namely the possible probabilistic behavior of a real system (there is a large field of application to fault tolerant systems for example). As a result, several models of probabilistic timed automata have been developed (see [KNSS02] and [Bea03]) as well as a lot of probabilistic temporal logics, and automatic verification methods for these models against the respective logics.

Recently, in [BRS02] a predicate logic of probabilities has been studied which leads to decidable model checking, when applied to Finite Probabilistic Processes (i.e. finite labelled Markov chains [KS60]). Finite Probabilistic Processes do not involve non determinism, contrary to Semi-Markov Processes which include both non determinism and probabilities. This model naturally implies the notion of adversary (or policy, strategy, depending on the authors). The adversary is used to resolve the non determinism.

In this paper we extend the predicate logic introduced in [BRS02] in order to deal with Semi-Markov Processes, by a modification of the probabilistic

\* Research partially supported by MIUR Progetto Cofinanziato MEFISTO.
operators. We prove that with respect to qualitative (probabilities are 0 or 1) probabilistic properties, model checking is decidable for this logic applied to Semi-Markov Processes.

Furthermore we apply our logic to Probabilistic Timed Automata giving to some predicates a fixed semantics; these predicates are clock predicates. They are of the form \( x_t - y_t + c \) where \( z(t) \) is the real value of clock \( z \) at step \( t \) (\( t \) is a natural), and \( c \) is an integer. We obtain two kinds of results: undecidability and decidability ones. In the general case, even without probabilistic operators, it turns out that the model checking is undecidable. If one restricts to clock predicates of the form \( x_t - y_t + c \), then, firstly, qualitative model checking is decidable, secondly, for urgent semantics, quantitative model checking is decidable for “almost” all values of the probabilistic parameters.

The structure of the paper is as follows. Section 2 gives basic definitions about labelled transition systems and weak second order monadic logic of order. In Section 3 we define a new logic and prove that the model checking is decidable for this logic applied to Semi-Markov Processes. Section 4 is devoted to undecidability and decidability results concerning model checking for Probabilistic Timed Automata with this logic enriched with clock predicates. The last section describes the future work and compares our logic with the existing ones.

2 Basic notion

A labelled transition system \( S \) is a tuple \((\Sigma, L, Q, q_{ini}, Tr, \lambda)\) such that: \( \Sigma \) is a set of symbols, \( L \) is a set of atomic propositions, \( Q \) is a set of states, \( q_{ini} \in Q \) is the initial state, \( Tr \subseteq Q \times \Sigma \times Q \) is a set of transitions, and \( \lambda \) is a function assigning to each state \( q \) a subset of atomic propositions \( \lambda(q) \). If \( q \) is a state and \( a \) is a symbol, then with \( S(q, a) \) we denote the set of transitions with source \( q \) and symbol \( a \) of the labelled transition system \( S \), more precisely it is the set \( \{(q, a, q') \mid (q, a, q') \in Tr\} \). We require that the set \( S(q, a) \) is finite, for each state \( q \) and symbol \( a \). Hence the set of state \( Q \) and the set of symbols \( \Sigma \) can be infinite sets, but the set of transitions \( S(q, a) \) must be finite.

A run of \( S \) is a possible infinite sequence of steps of the form \( \omega = q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} \ldots \) where \((q_i, a_i, q_{i+1})\) is in \( Tr \). The length of \( \omega \), denoted \( length(\omega) \), is equal to \( n \) if \( \omega \) is the finite run \( q_0 \xrightarrow{a_0} \ldots q_{n-1} \xrightarrow{a_{n-1}} q_n \), and \( \infty \) otherwise.

If \( length(\omega) = \infty \), then with \( inf(\omega) \) we denote the set of states of \( S \) crossed in \( \omega \) infinitely many times. Moreover, let \( k \leq length(\omega) \); with \( (\omega^k) \) we denote the state \( q_k \) and with \( \omega^{(k)} \) we denote the run \( q_0 \) if \( k = 0 \), and the run \( q_0 \xrightarrow{a_0} \ldots q_{k-1} \xrightarrow{a_{k-1}} q_k, \) otherwise.

If \( k = length(\omega) \), then we say that \( \omega \) is a prefix of \( \omega' \) if and only if \( length(\omega') \geq k \) and \( \omega = (\omega')^{(k)} \).

If \( \omega \) is a run \( q_0 \xrightarrow{a_0} \ldots q_{n-1} \xrightarrow{a_{n-1}} q_n \) sometimes we will write \( \omega \xrightarrow{a} s \) to denote the run \( q_0 \xrightarrow{a_0} \ldots q_{n-1} \xrightarrow{a_{n-1}} q_n \xrightarrow{a} q \).

With \( Path_{fin}(S, q) \) (resp. \( Path_{inf}(S, q) \)) we denote the set of finite (resp. infinite) runs \( \omega = q_0 \xrightarrow{a_0} \ldots \xrightarrow{a_n} q_{n+1} \ldots \) of \( S \) such that \( q = q_0 \). More-
over, with \( \text{Path}_{\text{fin}}(S) \) and \( \text{Path}_{\text{ful}}(S) \) we denote the sets \( \text{Path}_{\text{fin}}(S,q_{\text{ini}}) \) and \( \text{Path}_{\text{ful}}(S,q_{\text{ini}}) \), namely the set of finite and infinite runs starting from the initial state \( q_{\text{ini}} \).

A labelled transition system \( S = (\Sigma, L, Q, q_{\text{ini}}, Tr, \lambda) \) is a Finite Automaton if \( \Sigma, L \) and \( Q \) are finite.

**Definition 1.** Let \( Z \) be a set of predicate symbols; the set \( \text{WMLO}(Z) \) is the set of formulae \( \phi \) on \( Z \) defined by the following grammar:

\[
\phi ::= B(t) \mid t < t' \mid t \in X \mid \exists t.\phi_1 \mid \exists X.\phi_1 \mid \neg \phi_1 \mid \phi_1 \lor \phi_2
\]

where \( B \) is a predicate symbol in \( Z \), \( t, t' \) are two natural variables, \( X \) is a variable representing a finite set of naturals. Conjunction, implication and universal quantification can be easily derived.

Let \( Z \) be a set of predicate symbols and \( S \) a labelled transition system. A valuation \( v \) on \( Z \) and \( S \) is a function that assigns to each variable \( t \) a natural, to each variable \( X \) a finite set of naturals, and to each predicate symbol \( B \in Z \) a subset of \( \text{Path}_{\text{ful}}(S) \cup \text{IN} \) (called interpretation of \( B \)). With \( v[n/t] \) and \( v[N/X] \) we denote the valuations that update the value of the variable \( t \) with \( n \) and the value of variable \( X \) with \( N \), respectively.

We give now the semantics of a formula \( \text{WMLO}(Z) \).

**Definition 2.** Let \( S \) be a labelled transition system, \( Z \) be a set of predicate symbols, \( \omega \) be in \( \text{Path}_{\text{ful}}(S) \), \( v \) be a valuation, and \( \phi \in \text{WMLO}(Z) \); we define when a formula \( \phi \) holds at \( \omega \) in \( S \) under a valuation \( v \), written \( S, v, \omega \models \phi \), by the following inductive clauses:

\[
\begin{align*}
S, v, \omega & \models B(t) \quad \text{iff} \quad (\omega, v(t)) \in v(B) \\
S, v, \omega & \models t < t' \quad \text{iff} \quad v(t) < v(t') \\
S, v, \omega & \models t \in X \quad \text{iff} \quad v(t) \in v(X) \\
S, v, \omega & \models \exists t.\phi_1 \quad \text{iff} \quad S, v[n/t], \omega \models \phi_1, \text{ for some } n \in \mathbb{N} \\
S, v, \omega & \models \exists X.\phi_1 \quad \text{iff} \quad S, v[N/X], \omega \models \phi_1, \text{ for some finite set of naturals } N \\
S, v, \omega & \models \neg \phi_1 \quad \text{iff} \quad \text{it is not the case } S, v, \omega \models \phi_1 \\
S, v, \omega & \models \phi_1 \lor \phi_2 \quad \text{iff} \quad \text{either } S, v, \omega \models \phi_1 \text{ or } S, v, \omega \models \phi_2
\end{align*}
\]

We note that, we can express constants \( 0, 1, \ldots \) and conditions of the form \( t \sim c \) and \( t \sim t' + c \), where \( t \) and \( t' \) are natural variables, \( c \) is a natural and \( \sim \in \{<, \leq, =, \geq, >\} \). For more details see [Büc62] and [Tho90].

**Definition 3.** Let \( L \) be the set of atomic propositions of \( S \). From now on, we restrict ourselves to valuations \( v \) on \( L \) and \( S \) that give the following interpretation to \( L \); for each \( B \in L \), it hold that \( (\omega, i) \in v(B) \) if and only if \( B \in \lambda(\omega(i)) \).

Hence \( v(B) \), with \( B \in L \), is the set of pair \( (\omega, i) \) such that the \( i \)th state of \( \omega \) is labelled with \( B \).
If $S$ is a Finite Automaton and $F$ is a subset of states of $S$, then with $\text{Word}(S, F)$, we denote the set of infinite words $a_0a_1\ldots$ such that there exists a infinite run $\omega = q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} \ldots$ with $\inf(\omega) \cap F \neq \emptyset$ (Büchi acceptance condition, see [Tho90]).

Let $L = \{B_1, \ldots, B_k\}$ be a set of atomic propositions, and $\phi \in \text{WMLO}(L)$ with free variables in $X = \{t_1, \ldots, t_n, X_1, \ldots, X_m\}$. We define the set $\text{Word}(\phi)$ of strings $a_0a_1\ldots$, where $a_i = (b_{i1}^k, \ldots, b_{i(k+n+m)})$ and $b_{ij}^k \in \{0, 1\}$, such that $\phi$ is true when valuating:

- Each variable $t_j$ as the unique $i \in \mathbb{N}$ such that $b_{i(k+j)}^k = 1$;
- Each variable $X_j$ as the finite set $\{i \in \mathbb{N} \mid b_{i(k+n+j)}^k = 1\}$;
- Each formula $B_j(t)$ as the formula $b_{ij}^k = 1$ where $t$ is valuated as $i$.

We note that $\text{Word(true)}$ is not the set of all strings, but it is the set of strings where each variable $t$ is valuated as a natural value, and each variable $X$ is valuated as a finite set of naturals.

**Definition 4.** A Finite Automaton $S$ with a subset of states $F$ of $S$ represent a set of valuations for $L \cup X$ if $S$ has the symbols in the set $\{0, 1\}^{(k+n+m)}$ and $\text{Word}(S, F) \subseteq \text{Word(true)}$.

The following theorem can be derived from the results in [Büc62] and [Tho90].

**Theorem 1.** For each $\phi \in \text{WMLO}(L)$ with free variables in $X$, one can compute a Finite Automaton $S$ and a set of states $F$ that represent a set of valuations for $L \cup X$, such that $\text{Word}(S, F) = \text{Word(\phi)}$. Moreover, the vice versa also holds.

**Proof.** The algorithm proposed in [Büc62] and [Tho90] works for valuations that assign to a variable $X$ a set possibly infinite of naturals. Hence to use this algorithm, it is sufficient to consider the formula $\phi \land \exists t \forall t' > t. \bigwedge_{X \in X} t' \notin X$. Actually, since $v(X)$ is a finite set, then there exists an upper bound to the values in $\bigcup_{X \in X} v(X)$. For the vicevera, we have required that $S$ and $F$ represent a set of valuations and hence each string in $\text{Word}(S, F)$ valuates each variable $t$ as a natural and each variable $X$ as a set of finite naturals. Hence the formula derived from the results in [Büc62] and [Tho90] is a $\text{WMLO}(L)$. \qed

### 3 Probabilistic extensions

In this section, we consider Probabilistic Structures and a Probabilistic Logic $\text{PMLO}$ for them. We recall the definition of Semi Markov Process as a Probabilistic Structure, we recall known results about decidability for Markov Process (a sub class of the class of Semi Markov Processes). Moreover, we prove a decidability result for the class of Semi Markov Processes.
3.1 Probabilistic Structure

Definition 5. A Probabilistic Structure is a pair \( M = (S^M, \rho^M) \), where \( S^M = (\Sigma, L, Q, q_0, \text{Tr}, \lambda) \) is a labelled transition system and \( \rho^M : \text{Tr} \to [0, 1] \) is a probability transition function such that for each state \( q \in Q \) and each symbol \( a \in \Sigma \) we have that \( \sum_{e \in S^M(q, a)} \rho^M(e) = 1 \). We note that a transition cannot have a probability equal to 0.

From now on, for simplicity, we do not make distinction between \( M \) and the labelled transition system \( S^M \) of \( M \). As an example we will write \( \text{Path}_{\text{fin}}(M) \) to denote the set \( \text{Path}_{\text{fin}}(S^M) \), and \( M(q, a) \) to denote \( S^M(q, a) \).

Definition 6. An adversary \( A \) of a Probabilistic Structure \( M \) is a function from \( \text{Path}_{\text{fin}}(M) \) to \( \Sigma \) such that if \( A(\omega) = a \), then there exists a state \( q' \) such that \( \omega \xrightarrow{a} q' \) is in \( \text{Path}_{\text{fin}}(M) \).

If \( A \) is an adversary of \( M \), then with \( \text{Path}_{\text{fin}}^A(M) \) (resp. \( \text{Path}_{\text{ful}}^A(M) \)) we denote the set of finite (resp. infinite) runs \( \omega = q_0 \xrightarrow{a_0} \ldots \xrightarrow{a_n} q_{n+1} \ldots \) of \( M \) such that \( a_i = A(\omega(i)) \), for any \( 0 \leq i < \text{length}(\omega) \).

Definition 7. If \( \omega \) is a finite run \( q_0 \xrightarrow{a_0} \ldots \xrightarrow{a_{n-1}} q_n \), then with \( \overline{\pi}(\omega) \) we denote the probability computed as follows

\[
\overline{\pi}(\omega) = \begin{cases} 
1 & \text{if } n = 0 \\
\frac{1}{\overline{\pi}(\omega^{(n-1)})} \cdot \rho^M((q_{n-1}, a, q_n)) & \text{if } n > 0
\end{cases}
\]

If \( A \) is an adversary of \( M \), then with \( F^A_{\text{path}}(M) \) we denote the smallest \( \sigma \)-algebra on \( \text{Path}_{\text{ful}}^A(M) \) that contains the sets

\[
\{ \omega \mid \omega \in \text{Path}_{\text{ful}}^A(M) \land \omega' \text{ is a prefix of } \omega \}
\]

for any \( \omega' \in \text{Path}_{\text{fin}}^A(M) \).

Definition 8. The measure \( \mu \) on the \( \sigma \)-algebra \( F^A_{\text{path}}(M) \) is the unique measure such that

\[
\mu(\{ \omega \mid \omega \in \text{Path}_{\text{ful}}^A(M) \land \omega' \text{ is a prefix of } \omega \}) = \overline{\pi}(\omega')
\]

for any \( \omega' \in \text{Path}_{\text{fin}}^A(M) \).

From now on, we suppose that a valuation gives to a predicate \( B \) an interpretation that is measurable, more precisely, if \( v \) is a valuation, then we require that for each adversary \( A \) of \( M \), for each predicate symbol \( B \), and for each finite set of naturals \( N \), it holds that the set

\[
\{ \omega \in \text{Path}_{\text{fin}}^A(M) \mid (\omega, N) \in v(B) \}
\]

is measurable. We note that the interpretation given in the previous section to the set of atomic proposition \( L \) respects this requirement.
Definition 9. Let $Z$ be a set of predicate symbols; the set $\text{PMLO}(Z)$ is the set of formulae $\phi$ on $Z$ defined by the following grammar:

$$
\phi ::= B(t) \mid t < t' \mid t \in X \mid \exists P_{\sim p}(\phi_1 | \phi_2) \mid \exists \forall \phi_1 \mid \exists X. \phi_1 \mid \neg \phi_1 \mid \phi_1 \lor \phi_2
$$

where $B$ is a predicate symbol in $Z$, $t, t'$ are two natural variables, $X$ is a variable representing a finite set of naturals, $\sim \in \{<, \leq, =, \geq, >\}$ and $p \in \mathbb{Q}$.

Conjunction, implication and universal quantification can be easily derived. We call $\exists P_{\sim p}(\_)$ the probabilistic operator. With $\exists P_{\sim p}(\phi)$ we will denote the formula $\exists P_{\sim p}(\phi|\text{true})$.

The probabilistic operator $\exists P_{\sim p}(\phi_1 | \phi_2)$ means that there exists an adversary of $M$ such that the probability that $\phi_1$ holds when $\phi_2$ holds is related with the rational $p$ with the relation $\sim \in \{<, \leq, =, \geq, >\}$. A formula in $\text{PMLO}(Z)$ is closed if and only if it has no variable that is no quantified.

We give now the semantics of a formula in $\text{PMLO}(Z)$.

Definition 10. Let $Z$ be a set of predicate symbols, $M$ be a Probabilistic Structure, $\omega$ be in Path$_{fail}(M)$, $v$ be a valuation, and $\phi \in \text{PMLO}(Z)$. We define when a formula $\phi$ holds at $\omega$ in $M$ under a valuation $v$, written $M, v, \omega \models \phi$, by the following inductive clauses:

- $M, v, \omega \models B(t)$ \iff $(\omega, v(t)) \in v(B)$
- $M, v, \omega \models t < t'$ \iff $v(t) < v(t')$
- $M, v, \omega \models t \in X$ \iff $v(t) \in v(X)$
- $M, v, \omega \models \exists P_{\sim p}(\phi_1 | \phi_2)$ \iff if $m_i = \mu(\{\omega' \mid \omega' \in \text{Path}^A_{\sim p}(M) \land M, v, \omega' \models \phi_i\})$, for $i = 1, 2$, then $m_1 \sim p \cdot m_2$, for some adversary $A$
- $M, v, \omega \models \exists \forall \phi_1$ \iff $M, v[n/t], \omega \models \phi_1$, for some $n \in \mathbb{N}$
- $M, v, \omega \models \exists X. \phi_1$ \iff $M, v[N/X], \omega \models \phi_1$, for some finite set of naturals $N$
- $M, v, \omega \models \neg \phi_1$ \iff it is not the case $M, v, \omega \models \phi_1$
- $M, v, \omega \models \phi_1 \lor \phi_2$ \iff either $M, v, \omega \models \phi_1$ or $M, v, \omega \models \phi_2$

It is classical fact that the set $\{\omega \mid \omega \in \text{Path}^A_{\sim p}(M) \land M, v, \omega \models \phi\}$ is measurable (see [BRS02]).

3.2 Semi Markov Processes

Definition 11. A Probabilistic Structure $M = (S^M, \rho^M)$ is a Semi Markov Process if and only if $S^M$ is a Finite Automaton. If for each state $q$ there exists at most one label $a$ such that $M(q, a) \neq \emptyset$, then we call $M$ a Markov Process.

Form now on we consider the model checking problem on closed formulae in $\text{PMLO}(L)$, where $L$ is the set of atomic propositions of $M$. Hence, we say that $M$ satisfies $\phi \in \text{PMLO}(L)$, written $M \models \phi$, if and only if $M, v, \omega \models \phi$, for each valuation $v$ satisfying the Definition 3 and $\omega \in \text{Path}_{\text{fail}}(M)$. 

6
The set of parametrized formulae is defined similarly to the set $PMLO(Z)$ except that probabilistic operators $\exists p_{-\alpha}$ with $p \in \mathbb{Q}$ are replaced with $\exists P_{-\alpha}$, where $\alpha$ is a parameter name. Let $\phi$ be a parametrized formula with parameters $\alpha_1, \ldots, \alpha_n$, and $\mathbf{p} = (p_1, \ldots, p_n)$ be a vector of rational in $\mathbb{Q}^n$. With $\phi_{\mathbf{p}}$ we denote the formula in $PMLO(Z)$ replacing in $\phi$ each parameter $\alpha_i$ with $p_i$. By abuse of terminology we say that $\phi \in PMLO(Z)$ if each instance $\phi_{\mathbf{p}}$ of $\phi$ is in $PMLO(Z)$.

A known result for Markov Processes that states the decidability of the model checking for a rather large class of formula, is the following (see [BRS02]).

**Theorem 2.** Let $M$ be a Markov Process, $\epsilon > 0$ be a rational, and $\phi \in PMLO(L)$ be a parametrized formula where each probabilistic operator is of the form $\exists P_{-\alpha}(\phi')$ where in $\phi'$ free variables are natural variables and no probabilistic operators appear. One can compute, for each parameter $\alpha_i$ in $\phi$ ($i = 1, \ldots, n$), a finite set of intervals $H_i$ not containing zero and with length less than $\epsilon$ such that, if $\mathbf{p} \notin H_1 \times \cdots \times H_n$, then one can compute a formula $\phi' \in WMLO(L)$ such that $M \models \phi_{\mathbf{p}}$ if and only if $M \models \phi'$.

Now since we are interested in Semi Markov Processes, the previous theorem cannot be used. More precisely, the previous theorem strongly depends on the determinism of Markov Processes. We give now a result for Semi Markov Processes on qualitative formulae (i.e. formulae that have probabilistic operators $\exists P_{-\alpha}(\phi_1|\phi_2)$ where $p \in \{0, 1\}$). Hence we restrict the values of probabilities appearing in probabilistic operators but we extend the formulae by considering second order free variables in the scope of probabilistic operators.

Let $M$ be a Semi Markov Process with $L = \{B_1, \ldots, B_k\}$ as set of atomic propositions, and $\phi \in WMLO(L)$ with free variables $\{t_1, \ldots, t_n, X_1, \ldots, X_m\}$. If $S$ and $F$ are respectively the Finite Automaton and the set of states of Theorem 1, then we can construct a Semi Markov Process $M(\phi)$ with symbols in $\Sigma \times \{0, 1\}^{(n+m)}$ such that $M(\phi)$ is the cartesian product of $M$ and $S$. The transition $((q_1, q_2), (a, \beta), (q'_1, q'_2))$ is a transition of $M(\phi)$ if and only if $(q_1, a, q'_1)$ is a transition of $M$, $(q_2, (b_1, \ldots, b_k), \beta, q'_2)$ is a transition of $S$, and $b_i = 1$ if and only if $B_i$ labels $q_1$ in $M$, for $i = 1, \ldots, k$. Moreover, we denote with $P^M$ the set of states $(q, q')$ of $M(\phi)$ such that $q' \in F$.

An adversary is Markovian if it depends not on the past but only on the current state of the Semi Markov Process.

**Lemma 1 ([Bea02]).** Let $M$ be a Semi Markov Process and $F$ be a set of subset of its states. If $A$ is an adversary for $M$ and $q$ is a state, then the set $P_M(A,q) = \{\omega \in Path^{\Delta}_w(M,q) \mid \inf(\omega) \cap F \neq \emptyset\}$ is measurable. Moreover, one can compute in polynomial time for each state $q$, the minimal value of $\mu(P_M(A,q))$ for all adversaries $A$, as well as a Markovian adversary realizing this maximal value.

**Proposition 1.** One can compute a formula $\phi' \in WMLO(L)$ with the same free variables of $\phi$ such that, for each infinite run $\omega$ and valuation $v$, it holds that $M, v, \omega \models \exists P_{>0}(\phi)$ if and only if $M, v, \omega \models \phi'$. The formula $\phi'$ is computed...
in polynomial time on the size of $M(\phi)$, and hence, the size of $\phi'$ is polynomial on the size of $M(\phi)$.

Proof. Observe that, for each run $\omega = q_0 a_0 b_0 \ldots$ in $Path_{f\omega}(M(\phi), q)$ such that $inf(\omega) \cap F^M \neq \emptyset$, there exists a natural $n_1$ such that, for each $n_2 > n_1$, $\beta_{n_2} = (0, \ldots, 0)$. Let $M'$ be the Semi Markov Process obtained from $M(\phi)$ in the following way: for each symbol $a \in \Sigma$, if there exists a transition $q \xrightarrow{a, \beta} q'$ with $\beta \neq (0, \ldots, 0)$, then remove all transition starting from $q$ where label has a first component equal to $a$. Now the transitions have labels of the form $(a, (0, \ldots, 0))$. Using Lemma 1, one can compute the set $F^{>0}$ of states of $M'$ such that

$$
\max \{ \mu(P_M(A, q)) \mid A \text{ is an adversary of } M' \} > 0.
$$

Notice that $q \in F^{>0}$ iff there exists a Markovian adversary $A$ for $M'$ such that $\mu(P_M(A, q)) > 0$. Let $X = \{ t_1, \ldots, t_n, X_1, \ldots, X_m \}$ be the free variables in $\phi$, and $v$ be a valuation for these variables. Clearly, $M, v, \omega \models \exists P_{>0}(\phi)$ iff there is in $M(\phi)$ a finite run from the initial state to a state of $F^{>0}$ such that, for $i = 1, \ldots, n$, $h \leq n_1$, and $j = 1, \ldots, m$, we have that $v(t_i) = h$ if $b^{i}_{h} = 1$, and $h \in v(X_j)$ if $b^{m+j}_{h} = 1$. Hence, it is easy to construct a finite automaton $S(\phi)$ representing exactly the set of valuations $v$ of $X$ such that $M, v, \omega \models \exists P_{>0}(\phi)$.

Let $S(\phi)$ be the Finite Automaton with symbols in $\{0, 1\}^{(n+m)}$, with the same states of $M(\phi)$ and such that $(q, q', \phi')$ is a transition of $S(\phi)$ if and only if either $q = q' \in F^{>0}$ and $\beta = (0, \ldots, 0)$, or $q \notin F^{>0}$ and $(q, (a, \beta), q')$ is a transition of $M(\phi)$. Now $S(\phi)$ and $F^{>0}$ represent a set of valuations for the free variables of $\phi$ since has labels in $\{0, 1\}^{(n+m)}$ and each run starting from a state in $F^{>0}$ has labels of the form $(0, \ldots, 0)$.

Let $S(\phi)$ be the formula computed following Theorem 1 such that $Word(\phi') = Word(S(\phi), F^{>0})$. We prove that $M, v, \omega \models \exists P_{>0}(\phi)$ if and only if $M, v, \omega' \models \exists P_{>0}(\phi')$. If it holds, then, by Lemma 1 and Theorem 1, the thesis holds.

Since for each $\omega$ and $\omega'$ it holds that $M, v, \omega \models \exists P_{>0}(\phi)$ iff $M, v, \omega' \models \exists P_{>0}(\phi')$, the satisfiability of $\exists P_{>0}(\phi)$ depends only on $M$ and $v$. Moreover, the finite Automaton $S(\phi)$ once entered in a state in $F^{>0}$ loops. Actually, by definition of $M(\phi)$, by Theorem 1 and by Lemma 1, it is sufficient to enter in a state of $F^{>0}$ to have a word that describes a valuation satisfying $\exists P_{>0}(\phi)$. Hence, $\exists P_{>0}(\phi)$ is satisfied by exactly valuation $v$ such that there exists a string $a_0 a_1 \ldots$ in $Word(S(\phi), F^{>0})$, where $a_j = (b^j_1, \ldots, b^j_{n+m})$, and such that $v(t_i) = j$ if $b^{i}_{1} = 1$, and $v(X_i) = \{ j \mid b^{m+i}_{1} = 1 \}$. This implies that $M, v, \omega \models \exists P_{>0}(\phi)$ iff $M, v, \omega \models \phi'$.

The problem to compute $E^{>0}$ is polynomial in $M(\phi)$ (see Lemma 1). The formula $\phi'$ is computed in polynomial time in the size of $M(\phi)$ (see [Tho90]), and hence, has polynomial size in the size of $M(\phi)$.

The proof of $\exists P_{>1}(\phi)$ is similar to the previous one. But, since we must guarantee a probability equal to 1, we must consider subset of states of $M(\phi)$. This because, for the previous case, to guarantee a probability greater than zero, it is sufficient that there exists a path reaching a state in $F^{>0}$. Now, since the
probability must be equal to 1, we must guarantee that there exist paths leading exactly to a certain subset of states in $F^{-1}$.

**Proposition 2.** One can compute a formula $\phi' \in WMLO(L)$ with the same free variables of $\phi$ and such that, for each infinite run $\omega$ and valuation $v$, it holds that $M, v, \omega \models \exists P_{-1}(\phi)$ if and only if $M, v, \omega \models \phi'$. The formula $\phi'$ is computed in exponential time on the size of $M(\phi)$, and moreover, the size of $\phi'$ is exponential on the size of $M(\phi)$.

**Proof.** In the same way as in the proof of Proposition 1, one can compute the set $F_{-1}$ of states $q$ of $M'$ such that $\max(\{\mu(P_{M'}(A, q) \mid A \text{ is an adversary of } M')\}) = 1$. Notice that $q \in F_{-1}$ if there exists a Markovian adversary $A$ such that $\mu(P_{M'}(A, q)) = 1$. We define the Finite Automaton $S(\phi)$ where:

- The set of symbols is $\{0, 1\}^{(n+m)}$.
- States of $S(\phi)$ is the set of subsets of states of $M(\phi)$.
- The initial state is the set containing only the initial state of $M(\phi)$.
- The transition $(G, \beta, G')$ is a transition of $S(\phi)$ if and only if either $G = G' \subseteq F^{-1}$ and $\beta = (0, \ldots, 0)$, or there exists a function $f : G \rightarrow \Sigma$ such that $G'$ is the set

$$\{q' \mid \text{there exists } q \in G \text{ s.t. } (q, (f(q), \beta), q') \text{ is a transition of } M(\phi)\}.$$  

The idea is that a run $\omega$ of $S(\phi)$ represents the fact that there exists an adversary $A$ and a valuation $v$ such that the possible states reachable at step $i$ are $\omega(i)$, and the string that labels $\omega$ represents the valuation $v$. The function $f$ represents the choice of an adversary at a certain step. The finite automaton $S(\phi)$, once entered in a subset of states in $F_{-1}$, loops since, by Lemma 1 and by definition of $F_{-1}$, it is sufficient to enter in a subset of $F_{-1}$ to have a word that describes a valuation satisfying $\exists P_{-1}(\phi)$. Hence $\phi'$ is the formula such that $\text{Word}(\phi') = \text{Word}(S(\phi), 2F_{-1})$. The proof is similar to that of Proposition 1 by using Lemma 1 and since, if $(G, \beta, G')$ is a transition of $S(\phi)$, then there exists an adversary such that the probability to reach state $G'$ from $G$ is equal to 1.

The time to compute $F_{-1}$ is polynomial in $M(\phi)$ (see Lemma 1). The formula $\phi'$ is computed in exponential time in the size of $M(\phi)$, and moreover, has exponential size in the size of $M(\phi)$. □

Hence, since $\exists P_{-0}(\phi_1 \mid \phi_2)$ is equivalent to $\exists P_{-0}(\phi_1 \land \phi_2)$, by applying repeatedly Propositions 1 and 2, we have the following Theorem.

**Theorem 3.** Let $M$ be a Semi Markov Process with atomic propositions in $L$ and $\phi \in PMLO(L)$ be a qualitative formula. One can compute a formula $\phi' \in WMLO(L)$ such that $M \models \phi$ if and only if $M \models \phi'$. Hence the model checking problem for Semi Markov Processes of qualitative formulae in PMLO(L) is decidable.
4 Timed Probabilistic Automata

In this section, we define the class of Probabilistic Timed Automata and the Probabilistic Logic PMLO($L_C$) for Probabilistic Timed Automata.

4.1 Timed Probabilistic Automata

We assume a set $C$ of variables, called clocks. A clock valuation $\xi$ for a set of clocks $C$ is a function that assigns a non-negative real value to each clock. For a clock valuation $\xi$ and a time value $\tau$, $\xi + \tau$ denotes the clock valuation such that $(\xi + \tau)(x) = \xi(x) + \tau$, for any $x \in C$. Moreover, for a given set of clocks $C' \subseteq C$, with $\xi[C']$ we denote the clock assignment which sets each clock in $C'$ to 0, more precisely, $\xi[C'](x) = 0$ if $x \in C'$, and $\xi[C'](x) = \xi(x)$ otherwise.

The most general set of clock constraints over a set of clocks $C$, denoted $\Psi(C)$, is defined by the following grammar, where $1, 2$ ranges over $\Psi(C)$, $x, y \in X$, $c \in \mathbb{Z}$ and $<, \leq, =, \neq, >, \geq$.

$e ::= x \sim c | x - y \sim c | \psi_1 \land \psi_2 | \psi_1 \lor \psi_2 | \text{true}$

Definition 12. A tuple $T = (C, S, \text{trap}, \gamma)$ is a Probabilistic Timed Automaton if the following requirements are satisfied:

- $C$ is a finite set of clocks.
- $S = (\Sigma, L, Q, q_0, Tr, \lambda)$ is a Finite Automaton. We will write $T(q,a)$ to denote the set of transitions $S(q,a)$.
- trap is a trap state such that trap $\notin Q$, hence, no transitions in $Tr$ argue on trap.
- $\gamma : Tr \rightarrow \Psi(C) \times 2^C \times (0,1]$ is a probability condition function. If $\gamma(e) = (\psi, C', p)$, then with cond(e), res(e) and prob(e) we denote $\psi$, $C'$ and $p$, respectively. Moreover, we require that, for all states $q \in Q$ and symbols $a \in \Sigma$, we have that $\sum_{e \in T(q,a)} \text{prob}(e) = 1$.

We note that the previous definition is equivalent to that given in [Bea03]. In [Bea03] a transition is of the form $(q, a, \psi, C', p, q')$. The case in which between $q$ and $q'$, with the symbol $a$, we have $k > 1$ transitions can be modelled with our formalism by replicating the state $q'$ (each new state can be target only with transitions with a certain fixed pair $(\psi, C')$). We note that this transformation has a polynomial cost. In similar way we can simulate the definition of [KNSS02].

A configuration of $T$ is a pair $(q, \xi)$ where $q \in Q \cup \text{trap}$ and $\xi$ is a clock valuation on $C$. With $C(T)$, we denote the set of configurations of $T$.

The initial configuration is the configuration $(q_0, \xi_0)$, where for each clock $x$ it holds that $\xi_0(x) = 0$.

We consider now the Probabilistic Structure defined by $T$.

Definition 13. The Probabilistic Timed Automaton $T$ defines the Probabilistic Structure $(S^T, \rho^T)$ such that the labelled transition system $S^T$ is equal to
the tuple \( (\mathbb{R}^{\geq 0} \times \Sigma), L, C(T), (q_0, v_0), Tr^T, \lambda^T \) where \( \lambda^T(q, \xi) = \lambda(q) \), and

\((q, \xi), (\tau, a), (q', \xi')\) is in \( Tr^T \) if and only if one of the following requirements holds:

1. \( e = (q, a, q') \) is in \( Tr \), \( (\xi + \tau) \models \text{cond}(e) \) and \( \xi' = (\xi + \tau)[\text{res}(e)] \).
2. \( \xi' = \xi + \tau \), \( a = \text{trap} \), and there exists a transition \( e = (q, a, q'') \in Tr \), for

   some \( q'' \), such that \( (\xi + \tau) \not\models \text{cond}(e) \). 
3. \( \xi' = \xi + \tau \) and \( q = q' = \text{trap} \).

Moreover, the probabilistic function \( \rho^T \) is such that if \( e = ((q, \xi), (\tau, a), (q', \xi')) \)

is a transition in \( Tr^T \), then

\[
\rho^T(e) = \begin{cases}
\text{prob}((q, a, q')) & \text{if } q, q' \not\text{ trap} \\
\sum_{e' \in T(q, a)} \text{s.t. } \xi + \tau \not\models \text{cond}(e') \text{prob}(e') & \text{if } q \not\text{ trap and } q' = \text{trap} \\
1 & \text{if } q = q' = \text{trap}
\end{cases}
\]

Hence, if there exists a transition \( e \) with label \( a \) such that \( \text{cond}(e) \) holds at
time \( \tau \), then there exists a step with label \( (\tau, a) \). The new values of clocks are
incremented by time \( \tau \), and the clocks in \( \text{res}(e) \) are reset to 0. Moreover, there
exists a step that reaches the trap state at time \( \tau \) if there exists a transition \( e \)
with target \( q'' \) such that \( \text{cond}(e) \) does not hold at time \( \tau \). Finally, once entered
in a trap state we must loop on it.

As done before, we will not make distinction between \( T \) and the Probabilistic
structures \((S^T, \rho^T)\) and the labelled transition system \( S^T \) that \( T \) defines. Hence,
as an example, we will write \( T \models \phi \) to denote that \((S^T, \rho^T) \models T \) and \( \text{Path}_{ful}(T) \)
to denote the set \( \text{Path}_{ful}(S^T) \).

We define now the set of of predicates \( L_C \).

**Definition 14.** Let \( T \) be a Probabilistic Timed Automaton with set of atomic
propositions \( L \) and set of clocks \( C \). We define the set of predicate symbols \( L_C \) as follows:

\[
L_C = L \cup \{ \sim_{c, y}, \sim_{c, +}, \sim_{c} \}
\]

where \( x, y \in C \), \( c \in \mathbb{N} \) and \( \sim \in \{ <, <=, =, >, \\} \). From now on we suppose that
each valuation \( v \) gives to predicate symbols in \( L_C \) the following interpretation:

- The set of atomic propositions \( L \) has the interpretation given in Section 2.
- \( (\omega, i) \in v(\sim_{c, y}) \) if and only if \( \xi_i(x) \sim \xi_i(y) + c \) where \( \omega(i) = (q_i, \xi_i) \), namely
  the values of \( x \) and \( y \) in step \( i \) are related by \( \sim \) and \( c \). The case \( \sim_{c} \) requires
  that \( \xi_i(x) \sim c \).
- \( (\omega, i) \in v(\sim_{c, +}) \) if and only if \( \xi_i(x) + \tau_i \sim c \) where \( \omega = (q_0, \xi_0) \xleftarrow{\left(q_0, \tau_0 \right)} \), \ldots ,
  namely the values of \( x \) before the reset of step \( i \) is related with \( c \) by \( \sim \).

It is obvious that, if \( B \in L_C \), then the set \( \{ \omega \in \text{Path}_{ful}^A(M) \mid M, v, \omega \models B(t) \} \)
is measurable for all Semi Markov Processes \( M \), valuations \( v \), and adversaries
\( A \) of \( M \). Moreover, let \( t \) be a natural variable and \( x, y \) be two clocks, we will
write \( x_i \sim y_i + c \), \( x_i \sim c \) and \( x_i^+ \sim y_i^+ + c \) for the formulae \( \sim_{c, y} (t) \), \( \sim_{c} (t) \) and
We have not defined a formula of the form $x_1^+ \sim y_1^+ + c$ since this holds if and only if $x_1 \sim y_1 + c$ holds.

As for Semi Markov Processes, from now on we consider the model checking problem $T \models \phi$ such that $\phi$ is a closed formula in $PMLO(L_C)$. The decidability result on Semi Markov Processes cannot be directly used for Timed Probabilistic Automata since the Probabilistic Structure defined by a Probabilistic Timed Automaton has an infinite set of states and symbols. The fact that we have not considered relations between the values of variables in different steps is because a formula of the form $x_1 \sim y_1 + c$ makes the model checking problem undecidable (even if we consider formulae without probabilistic operators). Actually, let $L_{diff}$ be the set of predicates symbols $x_1 \sim y_1 + c$ with the semantics $(\omega, i_1, i_2) \in v(x_1 \sim y_1 + c)$ if and only if $\xi_{i_1} \sim \xi_{i_2} + c$, where $\omega(i_1) = (q_{i_1}, \xi_{i_1})$ and $\omega(i_2) = (q_{i_2}, \xi_{i_2})$. The following theorem states that the model checking problem for formulae in $WMLO(L_C \cup L_{diff})$ (hence formulae without probabilistic operators) is undecidable.

**Theorem 4.** It is undecidable to check whether $T \models \phi$ for a given Probabilistic Timed Automaton $T$ and a formula $\phi \in WMLO(L_C \cup L_{diff})$.

**Proof.** We translate the reachability problem of a 2-counter machine (that is undecidable) into the problem of checking whether $T \models \phi$.

A 2-counter machine consists of two counters $J$ and $K$, and a sequence of $n$ instructions. Each instruction may increment or decrement one of the counters, or jump, conditionally upon one of the counters being zero. After the execution of a non jump instruction, proceeds to the next instruction. A configuration is a triple $(b, n, m)$ where $b \in [0, n - 1]$ is the index of the actual instruction, $n$ is the value of $J$ and $m$ is the value of $K$. Sequences of configurations are defined in on obvious way. The problem to check whether there exists a finite sequence of configurations starting from $(0, 0, 0)$ such that the last configuration is equal to a given configuration $(b, n, m)$ is undecidable.

We consider the Probabilistic Timed Automaton $T$ with set of symbols $\{a\}$, set of clocks $C = \{x, pc, K, J\}$, set of states $\{q_1, q_{pc}, q_K, q_J\}$, labelling $\lambda$, such that $\lambda(q) = q$, for any state $q$, and set of transitions $Tr$ and probabilistic condition function $\gamma$ such that:

- $e = (q_1, a, q) \in Tr$, for any state $q$, and $\gamma(e) = (x = 0, C, \frac{1}{2})$;
- $e = (q_y, a, q_1) \in Tr$, with $y \in C \setminus \{x\}$, and $\gamma(e) = (x = 0 \land pc < n, \emptyset, \frac{1}{2})$;
- $(q_y, a, q_y') \in Tr$, with $y, y' \in C \setminus \{x\}$, and $\gamma(e) = (x = 1, \{x\}, \frac{1}{2})$.

The finite runs $\omega$ of $T$ are such that if $\{i_1, \ldots, i_l\}$ are the indexes of $\omega$ such that $\omega(i_j) = (q_i, \xi)$, then the triple $(\xi(pc), \xi(J), \xi(K))$ represents the configuration of the 2-counter machine at step $j^{th}$. In fact, the clock $x$ permits the clocks to assume only natural values since it is reset in each step and each condition requires that $x$ is either equal to 0 or equal to 1. Hence in the state $q_y$ the clock $y \in \{pc, K, J\}$ is reset. In the state $q_1$ we are able to read the configuration created in states $\{q_{pc}, q_K, q_J\}$.

Now we define a formula $\phi(b, n, m)$ such that $T \not\models \phi(b, n, m)$ if and only if the configuration $(b, n, m)$ is not reachable by the 2-counter machine.
Firstly we model the instructions. We show one modelling the increment of counter $J$. The other instructions can be modelled similarly. If we are on the step $t$, then the formula $\phi(t)$ equal to

$$\exists t'. q_1(t') \land pc_{t'} = pc_t + 1 \land J_{t'} = J_t + 1 \land K_{t'} = K_t \land \forall t'' \in (t, t'). \neg q_1(t'')$$

models the fact that in the next step w.r.t. $t$ (represented by $t'$) the counter $J$ is increased with 1. Hence, the set of sequences of length $\bar{t}$ of a 2-counter machine is modelled by the formula

$$\phi_{prog}(\bar{t}) = \forall t. \left( (t < \bar{t} \land q_1(t)) \Rightarrow \bigvee_{i \in [1,n]} ((pc_t = i) \Rightarrow \phi_i(t)) \right)$$

where $\phi_i(t)$ represents the formula which models the performing of the instruction $i$th of the 2-counter machine at step $t$.

Hence the closed formula modelling that the configuration $(b, n, m)$ is not reachable by the 2-counter machine is the following:

$$\phi(b, n, m) = \forall t. \left( q_1(t) \land \phi_{prog}(\bar{t}) \Rightarrow \neg (pc_t = b \land J_t = n \land K_t = m) \right)$$

Is is obvious that $T \not\models \phi(b, n, m)$ if and only of the 2-counter machine reaches the configuration $(b, n, m)$.

\[ \square \]

### 4.2 Region graph

Let us recall the notion of region graph as given in [AD94]. Since we consider diagonal constraints, we adopt the definition of regions given in [BDFP00].

Let $C$ be a set of clocks and $c_M$ be a natural constant. Let us consider the equivalence relation $\approx$ over clock valuations and constant $c_M$ that contains each pair of clock valuations $\xi$ and $\xi'$ such that:

- for each clock $x$, either $[\xi(x)] = [\xi'(x)]$, or both $\xi(x)$ and $\xi'(x)$ are greater than $c_M$ ([z] indicates the integer part of $z$).
- for each clock $x, y$, either $[\xi(x) - \xi(y)] = [\xi'(x) - \xi'(y)]$, or both $\xi(x) - \xi(y)$ and $\xi'(x) - \xi'(y)$ fall out of the interval $[-c_M, c_M]$.
- for each pair of clocks $x$ and $y$ with $\xi(x) \leq c_M$ and $\xi(y) \leq c_M$, $\text{frac}t(\xi(x)) < \text{frac}t(\xi(y))$ if and only if $\text{frac}(\xi(x)) < \text{frac}(\xi(y))$ ($\text{frac}(z)$ indicates the fractional part of $z$).

Note that for each pair of valuations $\xi$ and $\xi'$, and for each clock constraint $\phi$ with constants enclosed in $[-c_M, c_M]$, it holds that:

$$\text{if } \xi \approx \xi' \text{ then } (\xi \models \phi \iff \xi' \models \phi).$$

A clock region is an equivalence class of clock valuations induced by $\approx$. We denote by $[\xi]$ the equivalence class of $\approx$ containing $\xi$. Note that the set of clock regions is finite.
A region of $T$ is a tuple $(q, [\xi])$ where $q$ is a state of $T$ and $\xi$ is a clock valuation on clocks $C$ of $T$. The idea is that $(q, [\xi])$ represents the set of configurations $(q, \xi')$ such that $\xi' \in [\xi]$.

4.3 Extended region graph with classical semantics

The classical definition of region used in [AD94] and [BDFP00] is of a pair composed by a state and a clock region. Since in our logic we consider predicates in $L_C$, we must extend the classical definition of region graph. In fact the definition of \approx region graph considers states as regions. Since we must distinguish the elapsing of time from the performing of a transition in such a way to value $x_t$ and $x_t'$, we consider a notion of extended region graph. In the definition of the extended region graph the states are also marked with either a mark that represents the elapsing of time (label time) or a mark that represents the instantaneous performing of a transition (label trans).

**Definition 15.** Let $T = (C, (\Sigma, L, Q, q_0, Tr, \lambda), \gamma)$ be a Probabilistic Timed Automaton and $\phi \in \text{PMLO}(L_C)$. If $c_M$ is the natural constant greater than each constant appearing in $T$ and $\phi$, and $G$ is the set of predicates in $L_C \setminus \phi$ appearing in $\phi$, then the extended region graph for $T$ and $\phi$, denoted with $R(T, \phi)$, is the Semi Markov Process $((\Sigma^R, L^R, Q^R, q_0^R, Tr^R, \lambda^R), \rho^R)$ where:

- $\Sigma^R$ is the set of symbols $\Sigma \cup \{\lambda[\xi] \mid [\xi] \text{ is a clock region w.r.t. the constant } c_M\}$.
- $L^R$ is the set $L \cup G \cup \{\text{time}\}$. The atomic proposition time represents the fact that we are in a state marked by “time”.
- $Q^R$ is the set of tuples $(q, [\xi], \text{time})$ and $(q, [\xi], \text{trans})$, where $(q, [\xi])$ is a region of $T$ for $c_M$. The marks time and trans represent the fact that configurations expressed by the region $(q, [\xi])$ are reached with the elapsing of time and with an instantaneous transition, respectively.
- $q_0^R = (q_0, [\xi_0], \text{time})$ where $(q_0, \xi_0)$ is the initial configuration.
- The set of transitions $Tr^R$ is as follows:
  - $((q, [\xi], \text{time}), \lambda[\xi], (q', [\xi'], \text{trans}))$ is in $Tr^R$ if and only if $q = q'$ and there exist a time $\tau$ such that $[\xi'] = [\xi + \tau]$. Hence the label of the transition represents the clock region reachable with the time elapsing from the clock region represented in the source state.
  - $((q, [\xi], \text{trans}), a, (q', [\xi'], \text{time}))$ is in $Tr^R$ if and only if $e = (q, a, q')$ is in $Tr$ where $[\xi'] = [\xi + \text{res}(e)]$ and $\xi \models \text{cond}(e)$. Hence we express the configurations reachable by an instantaneous transition from a configuration expressed by $(q, [\xi])$.
  - $((q, [\xi], \text{trans}), a, (\text{trap}, [\xi'], \text{time}))$ is in $Tr^R$ if and only if $[\xi'] = [\xi]$ and there exists $e = (q, a, q') \in Tr$, for some $q'$, such that $\xi \not\models \text{cond}(e)$.
  - $((\text{trap}, [\xi], \text{trans}), a, (q, [\xi], \text{time}))$ is in $Tr^R$, for any symbol $a$ and clock region $[\xi]$.

- $\lambda^R$ is such that $\lambda^R((q, [\xi], \text{time}))$ is the set $\lambda(q) \cup \{\text{time}\} \cup G'$ such that $G' \subseteq G$ and $B \in G'$ iff either $B = \sim^+_{\xi} y$ and $\xi(x) \sim \xi(y) + c$, or $B = \sim_{\xi} y$ and $\xi(x) \sim c$. Moreover, $\lambda^R((q, [\xi], \text{trans}))$ is the set $\{\sim^+_c \in G \mid \xi(x) \sim c\}$.
The function \( \rho^R \) is such that \( \rho^R((q, [\xi], \text{time}), \lambda_{[\xi]}, (q', [\xi'], \text{trans})) = 1 \), since from each state with mark time it holds that there exists only one transition with label \( \lambda_{[\xi]} \), for each clock region \([\xi]\); and if \( e = ((q, [\xi], \text{trans}), a, (q', [\xi'], \text{time})) \) is a transition in \( Tr^R \), then

\[
\rho^R(e) = \begin{cases} 
\text{prob}((q, a, q')) & \text{if } q, q' \neq \text{trap} \\
\sum_{e' \in T(q, a)} \text{prob}(e') & \text{if } q \neq \text{trap} \text{ and } q' = \text{trap} \\
1 & \text{if } q = q' = \text{trap}
\end{cases}
\]

The extended region graph is a Semi Markov Process since for each \( \xi_1, \xi_2 \in [\xi] \) the set of transitions enabled in \( \xi_1 \) is the same as those enabled in \( \xi_2 \), and hence is unique in the clock region \([\xi]\). This holds since for each condition \( \psi \) that labels a transition it holds that if \( \xi_1 \approx \xi_2 \), then \( \xi_1 = \psi \) if and only if \( \xi_2 = \psi \).

We note also that a sequence of steps of an extended region graph is of the form

\[
\omega' = (q_0, [\xi_0'], \text{time}) \xrightarrow{\lambda_{[\xi_1]}/a} (q_0, [\xi_1'], \text{trans}) \xrightarrow{\lambda_{[\xi_2]}/a} (q_1, [\xi_2'], \text{time}) \xrightarrow{\lambda_{[\xi_3]}/a} \ldots,
\]

namely, an alternating sequence of states with marks \( \text{time} \) and \( \text{trans} \) and hence an alternating sequence of symbols \( \lambda_{[\xi]} \) and symbols in \( \Sigma \). Therefore, the idea is that \( x_1 \) must be evaluated in step 2 \cdot t of \( R(T, \phi) \) (i.e. the \( t \)th state marked by \( \text{time} \)). Moreover, \( x_1^+ \) must be evaluated in step 2 \cdot t + 1 of \( R(T, \phi) \) (i.e. the \( t \)th state marked by \( \text{trans} \)).

### 4.4 Relations between Times Probabilistic Automata and Extended Region Graph

As a consequence of results in [AD94], [BDFP00], [ACD93] and [ACD91] we have the following theorem.

**Theorem 5.** Let \( T \) be a Probabilistic timed Automaton with propositions in \( L \) and clocks in \( C \), and \( \phi \in \text{PMLO}(L_Q) \). The following facts hold.

- Let \( \omega = (q_0, \xi_0) \xrightarrow{(\tau_0, a_0)} (q_1, \xi_1) \ldots \) be in \( \text{Path}_{\text{fut}}(T) \). There exists a unique run
  
  \[
  \omega' = (q_0, [\xi_0'], \text{time}) \xrightarrow{\lambda_{[\xi_1]}/a} (q_0, [\xi_1'], \text{trans}) \xrightarrow{\lambda_{[\xi_2]}/a} (q_1, [\xi_2'], \text{time}) \xrightarrow{\lambda_{[\xi_3]}/a} \ldots
  \]

  of \( R(T, \phi) \) such that \( \xi_i \in [\xi_{2i}] \) and \( \xi_i + \tau_i \in [\xi_{2i+1}] \), for any \( i \geq 0 \). We say that \( \omega' \) is the representant of \( \omega \), and we denote it with \( [\omega] \).

- Let \( \omega = (q_0, [\xi_0], \text{time}) \xrightarrow{\lambda_{[\xi_1]}/a} (q_0, [\xi_1], \text{trans}) \xrightarrow{\lambda_{[\xi_2]}/a} (q_1, [\xi_2], \text{time}) \xrightarrow{\lambda_{[\xi_3]}/a} \ldots \) there exists a run
  
  \[
  \omega' = (q_0, [\xi_0'], \xrightarrow{\tau_0}, a_0) (q_1, \xi_1) \ldots
  \]

  in \( \text{Path}_{\text{fut}}(T) \) such that \( [\omega'] = \omega \). We say that \( \omega' \) is represented by \( \omega \).
It is obvious that the Theorem 5 holds also if one considers finite runs finishing in states marked with time. Moreover, if $S$ is a set of runs of $T$, then with $[S]$ we denote the set $\{ [\omega] \mid \omega \in S \}$. Theorem 5 states that the representant is unique. The following Lemma states that also the represented is unique if one considers runs defined by a certain adversary.

**Lemma 2.** Let $A$ be an adversary of $T$ and $\omega_1, \omega_2$ be two runs in either $\text{Path}_{\text{fin}}^A(T)$ or $\text{Path}_{\text{fin}}^A(T)$. If $[\omega_1] = [\omega_2]$, then $\omega_1 = \omega_2$.

**Proof.** Let $[\omega_1] = [\omega_2]$ and $\omega_1 \neq \omega_2$ where $\omega_i = (q_0, \xi_0^i) \xrightarrow{(r, a)_0} \cdots (q_n, \xi_n^i) \ldots$, for $i = 1, 2$. Let $j$ be the smallest index such that $\omega_1^{(j)} \neq \omega_2^{(j)}$. Obviously, $j > 0$ since $\xi_0^1 = \xi_0^2$. Thus $\omega_1^{(j-1)} \neq \omega_2^{(j-1)}$ and there exists $(r, a) = A(\omega_1^{(j)})$ such that $\omega_1^{(j-1)} \xrightarrow{(r, a)} (q_j, \xi_j^1)$ and $\omega_2^{(j-1)} \xrightarrow{(r, a)} (q_j, \xi_j^2)$. Thus $\xi_j^1 = \xi_j^2$ and it contradicts $\omega_1^{(j)} \neq \omega_2$. Actually, $\xi_j^1 = ((\xi_{j-1}^1 + \tau)(C_1))$ and $\xi_j^2 = ((\xi_{j-1}^2 + \tau)(C_2))$. Now $\xi_j^1 = \xi_j^2$, since $j$ is the minimum. Therefore, let $\xi = \xi_j^1 + \tau$; we have that $\xi_j^1 + \tau \neq \xi_j^2 + \tau$, but $\xi_j^1 + \tau \neq \xi_j^2$ implies $[\xi(C_1)] \neq [\xi(C_2)]$ and hence $[\omega_1] \neq [\omega_2]$. But, this contradicts the fact that $[\omega_1] = [\omega_2]$. □

We now prove an important result concerning the probabilities between runs and represented runs.

**Lemma 3.** Let $A$ be an adversary of $T$, and $A'$ be an adversary of $T(\phi)$ such that $\text{Path}_{\text{fin}}^{A'}(R(T, \phi))$ is equal to $\text{Path}_{\text{fin}}^A(T)$. For each measurable set $S \subseteq \text{Path}_{\text{fin}}^A(T)$, it holds that $\mu(S) = \mu([S])$.

**Proof.** By Theorem 5 we have that for each $\omega \in \text{Path}_{\text{fin}}^A(T)$, there exists a unique $\omega' \in \text{Path}_{\text{fin}}^{A'}(R(T, \phi))$ such that $\omega' = [\omega]$. Moreover, by Lemma 2, we have that for each $\omega \in \text{Path}_{\text{fin}}^A(T)$, there exists a unique $\omega' \in \text{Path}_{\text{fin}}^{A'}(R(T, \phi))$ such that $\omega = [\omega']$. Hence there exists a bijective function between $\text{Path}_{\text{fin}}^A(T)$ and $\text{Path}_{\text{fin}}^{A'}(R(T, \phi))$. Therefore the thesis holds if, for each $\omega \in \text{Path}_{\text{fin}}^{A'}(R(T, \phi))$, it holds that $\overline{\mu}(\omega) = \overline{\mu}([\omega])$, where, if $\omega = (q_0, \xi_0) \xrightarrow{(r_0, a_0)} \cdots (q_n, \xi_n)$, then $[\omega] = (q_0, [\xi_0], \text{time}) \xrightarrow{(r_0 + \tau_0)} (q_0, [\xi_0 + \tau_0], \text{trans}) \xrightarrow{a} \cdots (q_n, [\xi_n], \text{time})$.

We prove it by induction on the length of $\omega$. If $\text{length}(\omega) = 0$, then the thesis holds since $\overline{\mu}(\omega) = 1 = \overline{\mu}([\omega])$. If $\text{length}(\omega) = k > 0$, then, by definition of $R(T, \phi)$, we have that $\text{length}([\omega]) = 2 \cdot k$. Let $\omega' = \omega^{(k-1)}$; we have that $\overline{\mu}(\omega') = \overline{\mu}(\omega)$. Therefore the thesis holds since, by definition of $(S^T, \rho^T)$ and $R(T, \phi)$, $\rho^T(e_1) = 1$ and $\rho^T((q, \xi), (r, a), (q, \xi')) = \rho^R(e_2)$, and, by induction, $\overline{\mu}(\omega') = \overline{\mu}(\omega')$. □
If \( t \) is a natural variable, then with \( \overrightarrow{t} \) we consider a new natural variable related to \( t \). Let \( v \) be a valuation; with \( (2v) \) we denote the valuation such that
- for each natural variable \( t \), if \( v(t) \) is defined, then \( (2v)(t) = 2 \cdot v(t) \) and \( (2v)(\overrightarrow{t}) = 2 \cdot v(t) + 1 \); otherwise both \( (2v)(t) \) and \( (2v)(\overrightarrow{t}) \) are undefined.
- for each predicate variable \( X \), if \( v(X) \) is defined, then \( (2v)(X) = \{ 2 \cdot n \mid n \in v(X) \} \); otherwise \( (2v)(X) \) is undefined.

We want to prove that \( T, v, \omega \models \phi \) if and only if \( R(T, \phi), (2v), [\omega] \models Trans(\phi) \), where \( Trans \) is a function that, given a \( \phi \in PMLO(L_C) \), returns a formula in \( PMLO(L^R) \), where \( L^R \) is the set of atomic propositions of \( R(T, \phi) \). Hence in figure 1 we give the table of translations of formulae \( \phi \) in \( PMLO(L_C) \).

<table>
<thead>
<tr>
<th>( \phi \in PMLO(L_C) )</th>
<th>( Trans(\phi) \in PMLO(L^R) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( B(t) ) with ( B \in L \cup { \sim^x y, \sim^c x \mid x, y \in C \land c \in \mathbb{N} } )</td>
<td>( B(t) )</td>
</tr>
<tr>
<td>( B(t) ) with ( B \in { \sim^x y, \sim^c x \mid x \in C \land c \in \mathbb{N} } )</td>
<td>( B(\overrightarrow{t}) )</td>
</tr>
<tr>
<td>( t &lt; t' )</td>
<td>( t &lt; t' )</td>
</tr>
<tr>
<td>( t \in X )</td>
<td>( t \in X )</td>
</tr>
<tr>
<td>( \exists t. \phi_1 )</td>
<td>( \exists t, \overrightarrow{t}(time(t) \land \overrightarrow{t} = t + 1) \land Trans(\phi_1) )</td>
</tr>
<tr>
<td>( \exists X. \phi_1 )</td>
<td>( \exists X. [\forall t.(X(t) \Rightarrow time(t))] \land Trans(\phi_1) )</td>
</tr>
<tr>
<td>( \exists P_{\sim p}(\phi_1</td>
<td>\phi_2) )</td>
</tr>
<tr>
<td>( \sim \phi_1 )</td>
<td>( \neg Trans(\phi_1) )</td>
</tr>
<tr>
<td>( \psi_1 \lor \phi_2 )</td>
<td>( Trans(\phi_1) \lor Trans(\phi_2) )</td>
</tr>
</tbody>
</table>

**Fig. 1.** The function \( Trans \).

We explain the main idea. The predicates \( B \in \{ \sim^x y, \sim^c x \mid x, y \in C \land c \in \mathbb{N} \} \) must be evaluated in the even steps \( t \), when conditions \( B \in \{ \sim^x y, \sim^c x \mid x \in C \land c \in \mathbb{N} \} \) must be evaluated in step \( t + 1 = \overrightarrow{t} \). When we have a formula \( \exists t. \phi_1 \) we must ensure to consider in \( Trans(\exists t. \phi_1) \) the valuations that give an even value to \( t \) (as required by definition of \( (2v) \)). Hence, since the proposition \( time \) holds in \( R(T, \phi) \) only in the even steps, with \( time(t) \) we ensure that \( t \) is even, and with \( \overrightarrow{t} = t + 1 \) we ensure that \( \overrightarrow{t} \) is the successor of \( t \) (as the definition of \( (2v) \) requires). In a similar way the condition \( \exists t. (X(t) \Rightarrow time(t)) \) of \( Trans(\exists X. \phi_1) \) ensures that in \( X \) we have only even naturals. Now we prove the following result.

**Theorem 6.** It holds that \( T, v, \omega \models \phi \iff R(T, \phi) \models Trans(\phi) \).

**Proof.** We prove by induction on the structure of \( \phi \) of probabilistic operators that for each \( \omega \) and valuation \( v \), it holds that \( T, v, \omega \models \phi \) if and only if \( R(T, \phi), (2v), [\omega] \models Trans(\phi) \).

The case \( \phi = B(t) \) with \( B \in L \) is obvious since \( Trans(B(t)) = B(t) \) and by construction of \( R(T, \phi) \), Theorem 5 and by definition of \( (2v) \) we have that \( B \in \lambda(\omega(v(t))) \) if and only if \( B \in \lambda(q) \) where \( (\omega)[(2 \cdot v(t))] = (q, [\xi], time) \) if and
only if \( B \in \lambda^R(q) \) where (\([\omega](2v) \langle t \rangle\)) = (q, [\xi], \text{time})

The case \( \phi = B(t) \) with either \( B \in \{\sim^x_c, \sim^+_c, \sim^c \} \quad | \ x, y \in C \land c \in \mathbb{N} \) can be proved similarly. In fact the formulae in \( \{\sim^x_c, \sim^c \} \quad | \ x, y \in C \land c \in \mathbb{N} \) must be evaluated in the even steps, and hence, by definition of \((2v)\), in the step \( t \). Moreover, the formulae in \( \{\sim^+_c \} \quad | \ x, y \in C \land c \in \mathbb{N} \) must be evaluated in the step \( t + 1 \) since the value of \( x_t^+ \) is that expressed in the states with mark \( \text{trans} \). Therefore, by definition of \((2v)\), in the step \( t \).

For the case \( \phi = t < t' \), we have that \( v(t) < v(t') \) iff \( 2 \cdot v(t) < 2 \cdot v(t') \), but by definition of \((2v)\) we have \((2v)(t) < (2v)(t')\).

The case \( \phi = t \in X \) is obvious since \( \text{Trans}(t \in X) = t \in X \) and by definition of \((2v)\) we have that \( v(t) \in v(X) \) if and only if \( 2 \cdot v(t) \in \{2 \cdot n \mid n \in v(X)\} \) if and only if \((2v)(t) \in (2v)(X)\).

The cases \( \neg \phi_1 \) and \( \phi_1 \lor \phi_2 \) hold by induction.

We prove now the case \( \exists \lambda. \exists \xi. \exists \xi_0. \exists \xi_1. \phi_1 \phi_2 \).

First of all we note that \( \text{Trans}(\exists \xi. \exists \xi_0. \exists \xi_1. \phi_1 \phi_2) = \exists \xi. \exists \xi_0. \exists \xi_1. \text{Trans}(\phi_1) \text{Trans}(\phi_2) \).

Therefore, by induction, we have that for each \( \omega \) and valuation \( v \), it holds that \( T, v, \omega \models \phi_1 \) if and only if \( R(T, \phi_1) \), \((2v), [\omega] \models \text{Trans}(\phi_1) \), for \( i = 1, 2 \).

We prove the two implications:

\( T, v, \omega \models \exists \xi. \exists \xi_0. \exists \xi_1. \phi_1 \phi_2 \) implies \( R(T, \phi), (2v), [\omega] \models \text{Trans}(\exists \xi. \exists \xi_0. \exists \xi_1. \phi_1 \phi_2) \)

Since \( T, v, \omega \models \exists \xi. \exists \xi_0. \exists \xi_1. \phi_1 \phi_2 \), we have that there exists an adversary \( A \) of \( T \) such that \( \mu(S_1^A) \sim_p \mu(S_2^A) \) where \( S_1^A \) and \( S_2^A \) denote the sets of runs defined by adversary \( A \) and satisfying \( \phi_1 \) and \( \phi_2 \), respectively. More precisely, for \( i = 1, 2, \)

\[
S_i^A = \{ \omega' \mid \omega' \in \text{Path}^{A_i}_{\text{fin}}(T) \land T, v, \omega' \models \phi_i \}.
\]

We can construct an adversary \( A' \) for \( R(T, \phi) \) such that for each \( \omega = (q_0, \xi_0) \overset{a_0, \tau_0}{\longrightarrow} \cdots (q_n, \xi_n) \) in the set \( \text{Path}^{A_i}_{\text{fin}}(T) \), such that \( A(\omega) = (a, \tau) \) it holds that \( A'(\omega) = A(\omega) \) and \( A'(\omega') = a, \)

\[
\omega' = [\omega] \overset{\lambda(\xi_n + \tau, \xi_n + \tau)}{\longrightarrow} (q_n, [\xi_n + \tau], \text{trans}).
\]

\( A' \) is defined in any way for \( \omega \in \text{Path}^{A_i}_{\text{fin}}(R(T, \phi)) \) such that \( \omega \) represents no run in \( \text{Path}^{A_i}_{\text{fin}}(T) \).

This construction is possible thanks to Lemma 2 that states that there exists at most one run for each representant.

Therefore, by Theorem 5 it holds that \( \omega \in \text{Path}^{A_i}_{\text{fin}}(T) \) if and only if \( [\omega] \in \text{Path}^{A_i}_{\text{fin}}(R(T, \phi)) \), and hence \( \text{Path}^{A_i}_{\text{fin}}(R(T, \phi)) = [\text{Path}^{A_i}_{\text{fin}}(T)] \).
Now, by induction, we have that $S^A_i$ is equal to $[S^A_i]$ and $S^A_i$ is equal to $[S^A_i]$, where

$$S^A_i = \{ \omega \mid \omega \in \text{Path}^A_i(R(T, \phi)) \land R(T, \phi), (2v), \omega \models \text{Trans}(\phi_i) \},$$

for $i = 1, 2$.

Hence to have the thesis it is sufficient to prove that $\mu(S^A_1) = \mu(S^A_1)$ and $\mu(S^A_2) = \mu(S^A_2)$, but this holds by Lemma 3. Therefore we have proved that if there exists an adversary $A$ of $T$ such that $\mu(S^A_1) \sim p \cdot \mu(S^A_2)$, then there exists an adversary $A'$ of $T(\phi)$ such that $\mu(S^A_1) \sim p \cdot \mu(S^A_2)$.

- $R(T, \phi), (2v), [\omega] \models \text{Trans}(\exists P_{\exists p}(\phi_1|\phi_2))$ implies $T, v, \omega \models \exists P_{\exists p}(\phi_1|\phi_2)$

Since $R(T, \phi), (2v), [\omega] \models \exists P_{\exists p}(\text{Trans}(\phi_1)|\text{Trans}(\phi_2))$, we have that there exists an adversary $A$ of $R(\phi)$ such that $S^A_1 \sim p \cdot S^A_2$ with $S^A_1$ and $S^A_2$ denoting the sets of runs defined by adversary $A$ and satisfying $\text{Trans}(\phi_1)$ and $\text{Trans}(\phi_2)$, respectively. More precisely, for $i = 1, 2$,

$$S^A_i = \{ \omega' \mid \omega' \in \text{Path}^A_i(R(T, \phi)) \land R(T, \phi), (2v), \omega' \models \text{Trans}(\phi_i) \}$$

We construct the adversary $A'$ for $T$ such that for each run $\omega = (q_0, \xi_0) \xrightarrow{a_0, \tau_0} \ldots (q_n, \xi_n)$ in $\text{Path}_{\text{fin}}(T)$ it holds that $A'(\omega) = (a, \tau)$ if $A([\omega]) = \lambda_{[\xi_n+\tau]}$ and $A(\omega') = a$ where

$$\omega' = [\omega] \xrightarrow{\lambda_{[\xi_n+\tau]}} (q_n, [\xi_n + \tau], \text{trans}).$$

Therefore, by Theorem 5 it holds that $\omega \in \text{Path}^A_i(T)$ if and only if $[\omega] \in \text{Path}^A_i(R(T, \phi))$, and hence $\text{Path}^A_i(R(T, \phi)) = [\text{Path}^A_i(T)]$.

Now, by induction, we have that $S^A_1$ is equal to $[S^A_1]$ and $S^A_2$ is equal to $[S^A_2]$, where

$$S^A_i = \{ \omega \mid \omega \in \text{Path}^A_i(T) \land T, v, \omega \models \phi_i \},$$

for $i = 1, 2$.

Hence to have the thesis it is sufficient to prove that $\mu(S^A_1) = \mu(S^A_1)$ and $\mu(S^A_2) = \mu(S^A_2)$, but this holds by Lemma 3.

Therefore we have proved that if there exists an adversary $A$ of $R(\phi)$ such that $\mu(S^A_1) \sim p \cdot \mu(S^A_2)$, then there exists an adversary $A'$ of $T$ such that $\mu(S^A_1) \sim p \cdot \mu(S^A_2)$.

□

Hence by Theorems 6 and 3 we have the following corollary.

**Corollary 1.** Let $T$ be a Timed Probabilistic Automaton with propositions in $L$ and clocks in $C$. If $\phi \in PMLO(L_C)$ is a qualitative formula, then one can compute a formula $\phi' \in WMLO(L)$ such that $T \models \phi$ if and only if $R(T, \phi) \models \phi'$. Hence the model checking problem for Time Probabilistic Automata of qualitative formulae in PMLO($L_C$) is decidable.
4.5 Urgent semantics

We can consider also an urgent semantics (see [BT01] and [ABL98]), where transitions must be taken as soon as possible.

**Definition 16 (urgent semantics).** The Probabilistic Timed Automaton $T = (C, (\Sigma, L, Q, q_0, Tr, \lambda), trap, \gamma)$ defines with urgent semantics the Probabilistic Structure $(S^U, \rho^U)$ such that the labelled transition system $S^U$ is equal to the tuple $((\mathbb{R}^{\geq 0} \times \Sigma), L, C(T), (q_0, v_0), Tr^U, \lambda^U)$ where $((q, \xi), (\tau, a), (q', \xi'))$ is in $Tr^U$ if and only if one of the following requirements holds:

1. $e = (q, a, q')$ in $Tr$, $\xi + \tau \models \text{cond}(e)$ and $\xi' = (\xi + \tau)[\text{res}(e)]$. Moreover, if $e' \in T(q, a')$, then $\xi + \tau' \not\models \text{cond}(e')$, for each $\tau' < \tau$ and $a' \in \Sigma$. We call these kinds of steps real urgent transitions.
2. $\xi' = \xi + \tau$, $q' = \text{trap}$, and there exists a transition $e = (q, a, q'') \in T(q, a)$ such that $((q, \xi), (\tau, a), (q''')) \in Tr^U$, for some $(q', \xi')$.
3. $\tau = 0$, $\xi' = \xi$, $q' = \text{trap}$ and, for each time $\tau'$ and symbol $a'$ and configuration $(q'', \xi'')$, there exists no real urgent transition $((q, \xi), (\tau', a'), (q'', \xi''))$.

Moreover, $\lambda^U$ and $\rho^U$ are defined as $\lambda^T$ and $\rho^T$ of Definition 15, respectively.

The definition of step with urgent semantics requires that a step can be performed by using a transition with the minimum possible delay. Hence we have called these kinds of steps real urgent transitions since they are performed by using a real transition in $Tr$. Moreover, a step can lead into a trap state. We have two cases. In the former case, there exists a real urgent step in the meantime, hence a trap state is reached because in the meantime some transition is not enabled. In the latter case there is no real urgent step that is performable, and hence, the step is performed with time 0.

We will write $T \models_u \phi$ to denote $(S^U, \rho^U) \models \phi$.

As a consequence of Theorem 4 we have the following corollary.

**Corollary 2.** It is undecidable to check whether $T \models_u \phi$ for a given Probabilistic Timed Automaton $T$ and a given formula $\phi \in WML0(L \cup L_{\text{diff}})$.

**Proof.** Since the Probabilistic Timed Automaton defined in the proof of Theorem 4 has the same set of runs with urgent semantics, then the thesis holds. \qed

The extended region graph $R^u(T, \phi)$ corresponding to a urgent semantics can be constructed from $R(T, \phi)$ by deleting the transitions that do not satisfy the urgent semantics. These transitions can be easily computable. First of all the set of valuations in $[\xi]$ can be written as the convex space represented by a linear formula $\pi_\xi$ on real variables $\{x_{old} \mid x \in C\}$, where $x_{old}$ represents the value of clocks $x \in C$ before the elapsing of time (see [Bou02]). This holds also for condition $\text{cond}(e)$ labelling transition $e$. Actually, $\text{cond}(e)$ can be written as a linear formula $\pi_e$ on real variables $\{x_{new} \mid x \in C\}$, where $x_{new}$ represents the
value of clocks $x \in C$ after the time elapsing. Now, for each region $(q, [\xi])$, it is sufficient to construct the linear formula
\[
\exists \{x_{\text{next}}, x_{\text{old}} \mid x \in C\}, \bigwedge_{a \in \Sigma} \bigwedge_{e \in T(a,q)} x_{\text{next}} = x_{\text{old}} + \tau \land \pi_{[\xi]} \land \pi_{e},
\]
where $\tau$ represents the time in which the transition can be taken. By using quantifier elimination algorithm in [FR75], we have an equivalent formula of the form $\tau \in I$, where $I$ a finite union of intervals. If $I$ has minimum $\tau_M$, the only reachable clock region from $[\xi]$ is $[\xi + \tau_M]$. If $I$ has not a minimum, then the only reachable state is trap with a time $\tau = 0$.

Following the proof of Theorem 6, we can prove also the following theorem.

**Theorem 7.** Let $\phi \in \text{PMLO}(L_C)$. It holds that $T \models_{u} \phi$ if and only if $R^u(T, \phi) = \text{Trans}(\phi)$.

Hence by Theorems 7 and 3 we have the following corollary.

**Corollary 3.** Let $T$ be a Timed Probabilistic Automaton with propositions in $L$ and clocks in $C$. If $\phi \in \text{PMLO}(L_C)$ is a qualitative formula, then one can compute a formula $\phi' \in \text{WMLO}(L)$ such that $T \models_{u} \phi$ if and only if $R^u(T, \phi) = \phi'$. Hence the model checking problem for Time Probabilistic Automata with urgent semantics of qualitative formulae in $\text{PMLO}(L_C)$ is decidable.

Now, for urgent semantics we can consider cases in which the region graph $R^u(T, \phi)$ is a Markov Process.

**Lemma 4.** If $T$ is such that for each state $q$ there exists at most one symbol $a$ such that the set $T(q,a) \neq \emptyset$, then the extended region graph $R^u(T, \phi)$ is a Markov Process.

**Proof.** Let $q$ be a state of $T$ and $[\xi]$ be a clock region. For states with mark time we have that the definition of urgency requires that if $[\xi']$ is the clock region reachable form $[\xi]$ with the minimum time possible and such that for $[\xi']$ there exists at least one real urgent step, then there exists only one transition $e$ with source state $(q, [\xi], \text{time})$ and $e$ is labelled with $\lambda_{[\xi]}$. For states with mark trans, by hypothesis, we have that for each state $q$ there exists at most one symbol $a$ such that $T(q,a) \neq \emptyset$, but this implies that for each $r = (q, [\xi], \text{trans})$ there exists at most one symbol $a$ such that $R^u(T, \phi)(q,a) \neq \emptyset$.

Hence by Theorems 2 and 7, and by Lemma 4, we have the following result.

**Corollary 4.** Let $T$ be a Probabilistic Timed Automaton, $\epsilon > 0$ be a rational, and $\phi \in \text{PMLO}(L_C)$ be a parametrized formula where each probabilistic operator is of the form $\exists P_{\alpha_i}(\phi')$ where in $\phi'$ free variables are natural variables and no probabilistic operators appear. One can compute, for each parameter $\alpha_i$ in $\phi$ ($i = 1, \ldots, n$), a finite set of intervals $H_i$ not containing zero and with length less than $\epsilon$ such that, if $\overline{\alpha} \notin H_1 \times \cdots \times H_n$, then one can compute a formula $\phi' \in \text{WMLO}(L)$ such that $T \models_{u} \phi_{\overline{\alpha}}$ if $R^u(T, \phi_{\alpha}) \models \phi'$.
5 Discussion

In this paper we have considered the model checking problem of a logic with probabilities for Semi Markov Process and for Probabilistic Timed Automata. The logic considered extends the Weak Monadic Second Order Logic with probabilities operators and formulae on values of clocks in a certain step. We have proved decidability results for the class considering qualitative properties.

In this paper we have not considered two important features: repeated states and progress. We discuss now how treat these. Now, if one is interested to consider only runs that crosses infinitely many times states in a certain set $F$, then it is sufficient to consider the formula $\forall t.\exists t'.t' > t \land \text{rep}(t')$, where the atomic proposition $\text{rep}$ labels the states in $F$. Progress runs of a Probabilistic Timed Automaton are runs such that the sum of times diverges. Also this case can be tested. In [AD94] it is proved that progressive runs are runs that satisfy $x = 0 \lor x > c_M$ infinitely many times. But this is expressed by the formula $\forall t.\exists t'.t' > t \land x_t = 0 \land x_{t'} > c_M$.

Decidability results shown for Probabilistic Timed Automata, have consequences also in the non probabilistic case. In fact Theorem 6 implies that the model checking problem of a formula in $WMLO(L_C)$ for a non probabilistic Timed Automaton is decidable.

Finally, we compare the decidable classes defined in this paper with those known in the literature. Different works deal with logics with probabilities for Markov Processes (see [ASB+95], [HJ94] and [LS82], and [CY95] for a survey). These logics are extension of Branching Temporal logics. In [BRSS02] it is proved that in $pCTL$ (Probabilistic Branching Temporal Logic) there is no formula equivalent to $\phi = \exists t.P_{\leq 1}(B(t))$. The formula $\phi$ means that there exists a certain step $n$ such that each run at step $n$ satisfies $B$. As a consequence of this fact we have that there are qualitative formulae in $PMLO(L_C)$ non expressible in the logic $pTCTL$ (Probabilistic Branching Time Temporal Logic) defined in [KNSS02]. On the other hand, in [KNSS02] they consider also non qualitative properties. Hence the two classes are incomparable.

References


