Probabilistic Time Petri Nets

Yrvann Emzivat, Benoit Delahaye, Didier Lime, Olivier Henri Roux

Probabilistic Time Petri Nets

Yrvann Emzivat[1,3], Benoît Delahaye[2], Didier Lime[1], and Olivier H. Roux[1]

[1] École Centrale de Nantes, IRCCyN – UMR CNRS 6597 (France)
[2] Université de Nantes, LINA – UMR CNRS 6241 (France)
[3] Renault S.A.S. (France)
{Yrvann.Emzivat, Olivier-H.Roux, Didier.Lime}@irccyn.ec-nantes.fr
Benoit.Delahaye@univ-nantes.fr

Abstract. We introduce a new model for the design of concurrent stochastic real-time systems. Probabilistic time Petri nets (PTPN) are an extension of time Petri nets in which the output of tokens is randomised. Such a design allows us to elegantly solve the hard problem of combining probabilities and concurrency. This model further benefits from the concision and expressive power of Petri nets. Furthermore, the usual tools for the analysis of time Petri nets can easily be adapted to our probabilistic setting. More precisely, we show how a Markov decision process (MDP) can be derived from the classic atomic state class graph construction. We then establish that the schedulers of the PTPN and the adversaries of the MDP induce the same Markov chains. As a result, this construction notably preserves the lower and upper bounds on the probability of reaching a given target marking. We also prove that the simpler original state class graph construction cannot be adapted in a similar manner for this purpose.

Keywords: Time Petri nets, probabilistic systems, state classes, Markov decision processes

1 Introduction

Many highly critical applications, like autonomous vehicles, require the use of modelling tools that integrate concurrency, real-time constraints and probabilities. Designing such models is challenging for they require the development of new algorithms that combine both real-time and probabilistic verification techniques. Continuous-time Markov chains [1], continuous-time Markov decision processes [2], probabilistic timed automata [3], Markov automata [4] and stochastic timed automata [5] are but a few examples of models that were introduced with the intention of formally verifying probabilistic real-time systems. In particular, the product of probabilistic timed automata [6,7] provides the medium for concurrency in a real-time constrained environment. Yet, none of the aforementioned formalisms are adapted to the modelling of systems that exhibit variables whose bounds cannot be inferred. In contrast, the blending of concurrency and of such dynamical bounds is inherent to Petri net models.

Petri nets were enhanced through the use of stochastic temporal parameters and exponential distributions of firing times in [8,9] for the modelling of *concurrent* probabilistic real-time systems. The time Petri net model was extended by adding a probability density-function to the static firing interval of each non-deterministic transition [10]. These *stochastic time Petri nets* generalise time Petri nets [11] and involve the extension of the state class graph of [12] in order to account for stochastic information in each state class.

While stochastic time Petri nets are a powerful formalism in terms of expressivity and conciseness, we argue that the randomisation of transition rates is not necessarily required, while a randomisation of tokens in subsequent places might be needed. For example, a component failure in a gracefully degrading system can be linked to the firing of a transition whose rate is not necessarily subject to some random phenomenon, but whose outcome needs to be specified in terms of token generation. The *extended stochastic Petri nets* introduced in [13] allow firing times to belong to an arbitrary distribution and output places to be randomised, but they still require stringent restrictions, including the randomisation of transition rates.

In this paper, we introduce *probabilistic time Petri nets* (PTPN) as a new modelling formalism. By enhancing the forward incidence mapping of a classic time Petri net in such a way that transitions are mapped to a set of distributions of markings, we are able to extend the class of time Petri nets to a wider class of nets. The output arcs of a transition are effectively replaced with stacks of probabilistic hyperarcs. Each hyperarc contributes to the generation of tokens in output places of the transition. When a transition is fired in a PTPN, one hyperarc is chosen in each stack according to some probability distribution. A resulting marking emerges from this combination of choices. In fact, a time Petri net is a probabilistic time Petri net if the firing of any given transition almost surely leads to a certain marking.

The tools that are used for the analysis of time Petri nets can easily be adapted to our probabilistic setting. Here, we conform the classic atomic state class graph construction [14] to our class of nets in order to isolate the properties of a PTPN into a finite Markov decision process (MDP). We prove that this MDP induces the same Markov chains as the semantics of the PTPN, up to isomorphism. As a result, this construction preserves the lower and upper bounds on the probability of reaching a given marking. This allows us to make use of the extensive set of tools that are used for the study of MDPs in order to thoroughly study the probabilistic real-time reachability problem in the context of PTPNs. The construction we put forward is quite complex, for it is based on the atomic state class graph. Unfortunately, we prove that the simpler original state class graph construction cannot be adapted to our setting as it does not preserve these lower and upper probability bounds.

Outline. We introduce the syntax and the semantics of probabilistic time Petri nets in section 2 and consider the verification of PTPNs against reachability properties in section 3. We conclude the present work in section 4 and suggest directions for future research.

2 Probabilistic Time Petri Nets

2.1 Preliminaries

We denote the set of natural numbers by $\mathbb{N}$, the set of rational numbers by $\mathbb{Q}$ and the set of real numbers by $\mathbb{R}$. We consider 0 to be an element of $\mathbb{N}$ and let $\mathbb{N}^* = \mathbb{N} \setminus \{0\}$. For $n \in \mathbb{N}$, we let $[\![0, n]\!]$ denote the set $\{i \in \mathbb{N} \mid i \leq n\}$. The set of real intervals that have rational or infinite endpoints is denoted $\mathscr{I}(\mathbb{Q}_+)$. A *clock valuation* over a set T is a mapping $v : T \to \mathbb{R}_+$, where $\mathbb{R}_+$ denotes the set of non-negative real numbers. We let 0_T denote the clock valuation that assigns 0 to every clock in T. For $d \in \mathbb{R}_+$, we let $v + d$ be the clock valuation that satisfies $(v + d)(t) = v(t) + d$ for every clock t in the domain of v.

For a given set X, let $\mathscr{P}(X)$ denote the power set of X. The *characteristic (or indicator)* function of $A \in \mathscr{P}(X)$, denoted $\chi_A : X \to \{0, 1\}$, is defined as

$$\chi_A(x) = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{if } x \notin A. \end{cases}$$

Given two arbitrary sets E and F, let F^E denote the set of functions from E to F. When F is an ordered set, we define a partial order $\preceq$ on F^E by $f \preceq g$ if $f(x) \leq g(x)$ for all $x \in E$.

A *discrete probability distribution* over a countable set X is a function $\mu : X \to [0, 1]$ such that $\sum_{x \in X} \mu(x) = 1$. The *support* of a discrete probability distribution μ, denoted $Supp(\mu)$, is the preimage of the interval $]0, 1]$ under μ. For an arbitrary set X, we define $\mathscr{D}ist_X$ to be the set of functions $\mu : X \to [0, 1]$ such that $Supp(\mu)$ is a countable set and μ restricted to $Supp(\mu)$ is a discrete probability distribution. For $x_0 \in X$, let the discrete probability distribution denoted δ_{x_0} be the *Dirac measure* which assigns probability 1 to x_0:

$$\delta_{x_0}(x) = \chi_{\{x_0\}}(x) = \begin{cases} 1 & \text{if } x = x_0, \\ 0 & \text{if } x \neq x_0. \end{cases}$$

2.2 Probabilistic time Petri nets

This section introduces the syntax and the semantics of *probabilistic time Petri nets*. Intuitively, a probabilistic time Petri net is a time Petri net in which every non-deterministic choice involves the resolution of a probabilistic experiment. Such experiments are described explicitly by means of discrete probability distributions. In a typical time Petri net, these probability distributions are Dirac measures. In other words, any given state of a time Petri net has a successor that is uniquely determined by a chosen course of action. This is generally not the case for probabilistic time Petri nets, which extend the class of time Petri nets as a result.

Syntax of a probabilistic time Petri net.

Definition 1 (Probabilistic time Petri net (PTPN)). *A* probabilistic time Petri net *is a quintuple* $\mathcal{N} = (P, T, Pre, Post, I)$ *where*

- P *is a finite, non-empty set of* places,
- T *is a finite set of* transitions *such that* $T \cap P = \emptyset$,
- $Pre : T \to \mathbb{N}^P$ *is the backward incidence mapping,*
- $Post : T \to \mathscr{P}(\mathscr{D}ist_{\mathbb{N}^P})$ *is the forward incidence mapping, and*
- $I : T \to \mathscr{I}(\mathbb{Q}_+)$ *is a function assigning a* firing interval *to each transition.*

An element of $\mathbb{N}^P$ is called a *marking* of the net. A marking denotes a distribution of *tokens* in the places of the net. The forward incidence mapping $Post$ specifies a *finite* set of probability distributions of markings for every transition of the net. For a given transition t, we assume that the probability distributions in $Post(t)$ are associated with *independent* random variables. These random variables each contribute to the production of tokens in subsequent places when that transition is fired. Moreover, we assume that the support of each discrete probability distribution in $Post(t)$ is finite.

A distribution in $Post(t)$ is represented graphically by a stack of *hyperarcs*. A hyperarc is labelled with a probability before it is split into a set of arcs that lead to a set of *output places*. These arcs contain information about the number of tokens that are generated in each one of these places when that hyperarc is selected.

Definition 2 (Marked probabilistic time Petri net). *A* marked probabilistic time Petri net *is a sextuple* $\mathcal{N} = (P, T, Pre, Post, I, \rho_{\mathcal{N}})$ *where*

- $(P, T, Pre, Post, I)$ *is a probabilistic time Petri net, and*
- $\rho_{\mathcal{N}} \in \mathscr{D}ist_{\mathbb{N}^P}$ *is the* distribution of initial markings *of the net.*

The experiment that yields the initial marking of the net is only conducted once. Any marking belonging to the support of $\rho_{\mathcal{N}}$ is *an* initial marking of the net. The value $\rho_{\mathcal{N}}(M)$ specifies the probability that *the* initial marking of the net is indeed M.

Example 1. In order to grasp the intuition behind the proposed model, let us consider the probabilistic time Petri net depicted in Fig. 1. Transition T_2 of the net displays two probability distributions. The first distribution either generates one token in P_1, three tokens in P_3 and one token in P_4 with probability a, or two tokens in P_4, one token in P_5 and one token in P_6 with probability b. No token is generated with probability $c = 1 - a - b$. The second distribution generates one or two tokens in P_6 with probability p and $1-p$ respectively. Since these distributions are associated with independent random variables, it follows that the firing of T_2 leads to the consumption of one token in P_1 and two tokens in P_2, and the generation of two tokens in P_4, one token in P_5 and three tokens in P_6 with probability $b(1-p)$.

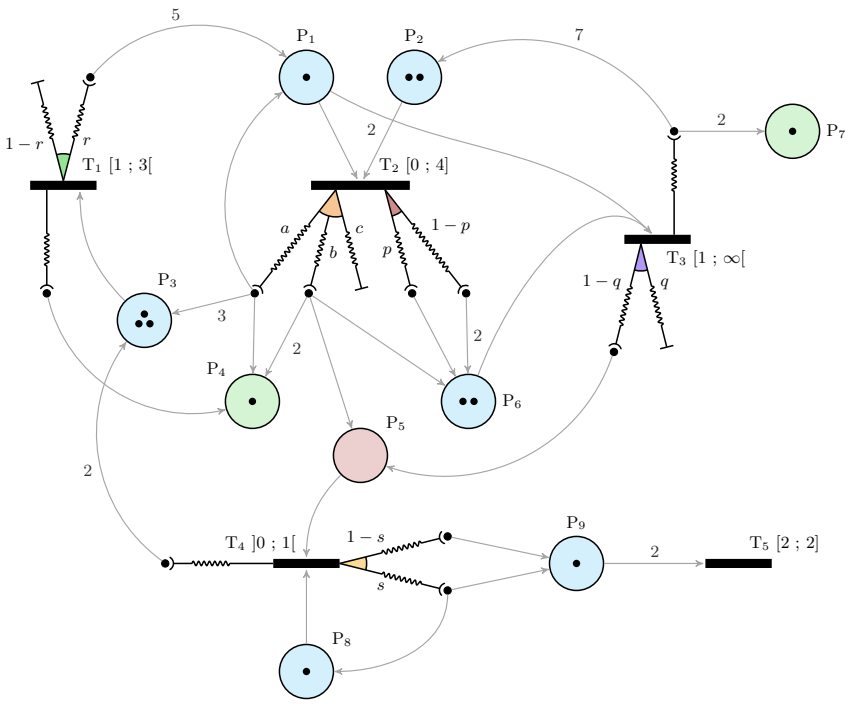

Fig. 1. A marked probabilistic time Petri net in its initial state, given by $\rho_{\mathcal{N}} = \delta_{(1,2,3,1,0,2,1,1,1)}$

The following paragraph introduces the terminology of probabilistic time Petri nets as well as important notations that are used throughout this paper. Let $\mathcal{N} = (P, T, Pre, Post, I)$ be a probabilistic time Petri net. A *state* of the net $\mathcal{N}$ is described by an ordered pair (M, v) in $\mathbb{N}^P \times \mathbb{R}_+^T$, where M is a marking of $\mathcal{N}$ and v is a clock valuation over the set of transitions T. In practice, clock valuations are only defined for transitions that are enabled.

– A transition $t \in T$ is said to be *enabled* by a given marking $M \in \mathbb{N}^P$ if M supplies t with at least as many tokens as required by the backward incidence mapping Pre. We define the set $\mathscr{E}(M)$ of transitions that are enabled by the marking M as

$$\mathscr{E}(M) = \{t \in T \mid M \succeq Pre(t)\}.$$

– A transition $t \in T$ is said to be *firable* from a given state (M, v) if the transition t is enabled by M and if its clock is assigned a value that lies within its firing interval. We define the set $\mathscr{F}(M, v)$ of transitions that are firable from the state (M, v) as

$$\mathscr{F}(M, v) = \{t \in \mathscr{E}(M) \mid v(t) \in I(t)\}.$$

– A time delay $d \in \mathbb{R}_+$ is said to be *compliant* with a given state (M, v) if every transition that is firable from $(M, v+d')$ for some time delay $d' \in [0, d]$ stays firable until $(M, v+d)$. We define the set $\mathscr{C}(M, v)$ of time delays that are compliant with the state (M, v) as

$$\mathscr{C}(M, v) = \{d \in \mathbb{R}_+ \mid \forall t \in T, t \notin \mathscr{F}(M, v+d) \Rightarrow \forall d' \in [0, d], t \notin \mathscr{F}(M, v+d')\}.$$

– An action $(d, t) \in \mathbb{R}_+ \times T$ is said to be *feasible* from a given state (M, v) if the time delay d leads the net to a state from which t is firable. We define the set $\Phi(M, v)$ of actions that are feasible from the state (M, v) as

$$\Phi(M, v) = \{(d, t) \in \mathbb{R}_+ \times T \mid d \in \mathscr{C}(M, v) \text{ and } t \in \mathscr{F}(M, v+d)\}.$$

When adopting a purely semantical standpoint, an element of the set T is best referred to as a *trial*, through the medium of an underlying probability distribution μ_t. Informally, a trial t induces the production of tokens in the net each time it is conducted, by providing alternatives that lead to one marking or another.

– Let $t \in T$ be a transition of $\mathcal{N}$. The discrete probability distributions in $Post(t)$ are associated with random variables that can take one of many values. By definition, these values are endowed with a non-zero probability. An *alternative* is a function f that chooses a value for each one of these random variables. Formally, we define the set $\mathscr{A}(t)$ of *alternatives* provided by the transition t as follows:

$$\mathscr{A}(t) = \left\{ f : Post(t) \to \mathbb{N}^P \mid \forall \mu \in Post(t), f(\mu) \in Supp(\mu) \right\}.$$

– An *outcome* of a given trial t is a marking ω of $\mathcal{N}$ which results from the choices of some alternative in $\mathscr{A}(t)$. This marking ω accounts for the tokens that are to be generated in each output place of t. We define the set $\Omega(t)$ of outcomes of the trial t as

$$\Omega(t) = \left\{ \omega \in \mathbb{N}^P \mid \exists f \in \mathscr{A}(t), \omega = \sum_{\mu \in Post(t)} f(\mu) \right\}.$$

– For a given outcome $\omega \in \Omega(t)$, we define the non-empty set $\mathscr{A}_\omega(t) \subseteq \mathscr{A}(t)$ of alternatives that lead to it as

$$\mathscr{A}_\omega(t) = \left\{ f \in \mathscr{A}(t) \mid \omega = \sum_{\mu \in Post(t)} f(\mu) \right\}.$$

Let us now provide the formal definition of the probability distribution μ_t that governs a trial $t \in T$. Intuitively, the probability of reaching a given outcome $\omega \in \Omega(t)$ is the sum of the probabilities of all the alternatives leading to ω. Since the probability distributions in $Post(t)$ are assumed to be independent, the probability that an alternative is chosen is the product of the probabilities of all the independent choices it makes. Formally, μ_t is defined as follows:

Definition 3. *Let $(P, T, Pre, Post, I)$ be a probabilistic time Petri net. The discrete probability distribution that governs a trial $t \in T$ is a function $\mu_t : \Omega(t) \to [0, 1]$ that assigns probabilities to the outcomes of t as follows:*

$$\mu_t : \omega \to \sum_{f \in \mathcal{A}_\omega(t)} \left(\prod_{\mu \in Post(t)} \mu\big(f(\mu)\big) \right).$$

Lemma 1. *Let $(P, T, Pre, Post, I)$ be a probabilistic time Petri net. For a given trial $t \in T$, the function μ_t is a discrete probability distribution over $\Omega(t)$.*

The probabilistic time Petri nets depicted in Fig. 2 have different structures. Yet they are equivalent from a semantical standpoint, since the discrete probability distribution μ_{T_1} is the same in both nets. In fact, every probabilistic time Petri net can be canonicalised into a probabilistic time Petri net such that $Post(t)$ is a singleton for every transition t.

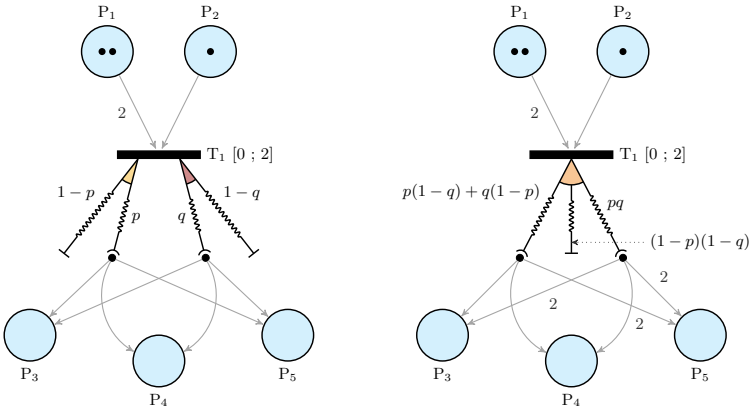

Fig. 2. Two syntactically different probabilistic time Petri nets that are equivalent from a semantical standpoint

It is worth noting that a probabilistic time Petri net is equivalent to a time Petri net if $Supp(\mu_t)$ is a singleton for all trials t. A time Petri net can therefore be interpreted as a probabilistic time Petri net whose transitions yield a single combination of hyperarcs, or similarly, whose trials each lead to a single outcome.

Semantics of a probabilistic time Petri net. A probabilistic time Petri net $\mathcal{N}$ has the following operational behaviour. The distribution $\rho_{\mathcal{N}}$ yields the initial marking M_0 of the net $\mathcal{N}$ and subsequently, the initial state $(M_0, 0_T)$ of $\mathcal{N}$. When in a given state, the net can either fire an enabled transition or let time flow. Doing so changes the state of the net. An enabled transition is firable if and only if its clock value lies within its firing interval. Furthermore, a

time delay must always be compliant with the current state of the net. In other words, time can flow as long as otherwise enabled transitions are not disabled in the process. This behaviour is typical in the context of *strong time semantics* and conveys the notion of urgency. As such, the behaviour of a probabilistic time Petri net is similar to that of a classic time Petri net. Once a choice has been made, however, the next state is selected in a probabilistic manner. The difference therefore lies in the way the subsequent state of the net is computed once the non-determinism has been resolved.

- If the net chooses to let a certain amount of time d to elapse, then the marking remains the same while the clock values of the enabled transitions are increased by that particular amount.
- If the net chooses to fire a certain transition t, tokens are removed from the current marking according to the mapping $Pre(t)$ while the outcome of the trial t generates additional ones. Moreover, the clocks associated with the transition t and with any transition that has been disabled by the removal of the $\sum_{p\in P} Pre(t)(p)$ tokens are reset and disabled. Finally, the clocks associated with newly enabled transitions are activated. This includes those that were previously disabled.

The semantics of a probabilistic time Petri net is defined as a probabilistic timed transition system. Probabilistic timed transition systems can be considered an extension of Markov decision processes that account for the flow of time, leading to a potentially uncountable set of states. Formally:

Definition 4 (Probabilistic timed transition system (PTTS)). *A* probabilistic timed transition system *is a quadruple* (Q, ρ, T, W) *where*

- Q *is a set of* states*,*
- $\rho \in \mathcal{D}ist_Q$ *is the distribution of* initial states*,*
- T *is a set of* trials*, and*
- $W : Q \times (T \cup \mathbb{R}_+) \to \mathcal{D}ist_Q$ *is a (partial) probabilistic transition function.*

We now formally introduce the semantics of marked probabilistic time Petri nets in terms of probabilistic timed transition systems.

Definition 5 (Semantics of a marked probabilistic time Petri net). *The semantics of a marked probabilistic time Petri net* $\mathcal{N} = (P, T, Pre, Post, I, \rho_{\mathcal{N}})$ *is a probabilistic timed transition system* $S_{\mathcal{N}} = (Q, \rho_{S_{\mathcal{N}}}, T, W)$ *where*

- $Q \subseteq \mathbb{N}^P \times \mathbb{R}_+^T$ *is the set of* states *of the net* $\mathcal{N}$*,*
- $\rho_{S_{\mathcal{N}}} : Q \to [0, 1]$ *is the distribution of* initial states*, defined for* $(M, v) \in Q$ *by*
$$\rho_{S_{\mathcal{N}}}(M, v) = \rho_{\mathcal{N}}(M) \times \chi_{\{0_T\}}(v) \text{ , and}$$
- $W : Q \times (T \cup \mathbb{R}_+) \to \mathcal{D}ist_Q$ *is the (partial) piecewise* probabilistic transition function *that defines* continuous time transition relations *over* $Q \times \mathbb{R}_+$ *and* discrete transition relations *over* $Q \times T$*.*

1. *W is defined for $((M,v),d) \in Q \times \mathbb{R}_+$ if and only if the delay d is compliant with the state (M,v). In that case, let $W((M,v),d)$ be the Dirac measure $\delta_{(M,v')}$, where the clock valuation v' is defined for all transitions t' enabled by the marking M by*

$$v'(t') = v(t') + d.$$

2. *W is defined for $((M,v),t) \in Q \times T$ if and only if the transition t is firable from the state (M,v). In that case, let $W((M,v),t) = \tilde{\mu}_t$, where $\tilde{\mu}_t \in \mathscr{D}ist_Q$ is defined as follows:*
   - *Let $(M',v') \in Q$. The state (M',v') lies in $Supp(\tilde{\mu}_t)$ if and only if the two following conditions are met:*
     - *there exists an outcome $\omega_{M'} \in \Omega(t)$ such that*

$$M' = (M - Pre(t)) + \omega_{M'} \tag{1}$$

     - *the clock valuation v' is defined for all transitions t' enabled by the marking M' by*

$$v'(t') = v(t') \times (1 - \chi_t(t')) \times \chi_{\mathscr{E}(M-Pre(t))}(t') \tag{2}$$

   - *Suppose that $(M',v') \in Supp(\tilde{\mu}_t)$. We define the image of (M',v') by the formula*

$$\tilde{\mu}_t(M',v') = \mu_t(\omega_{M'}).$$

Figure 3 depicts a probabilistic time Petri net and a fragment of its semantics. Clock valuations are not represented.

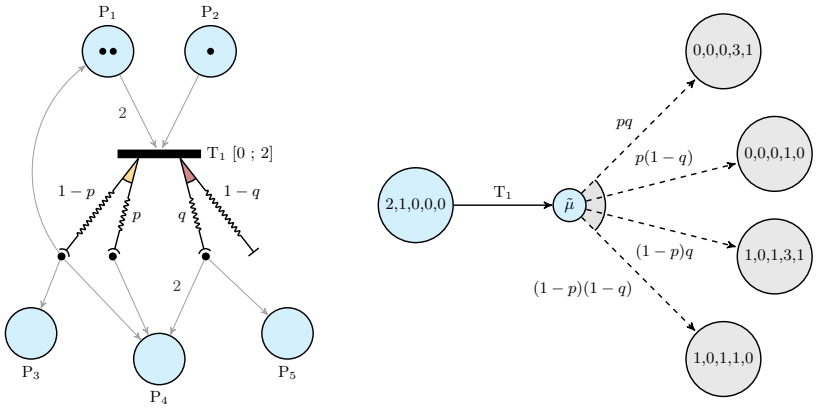

Fig. 3. Correspondence between the transitions of a probabilistic time Petri net and the trials of its semantics

3 The Probabilistic Real-Time Reachability Problem

A state is said to be *reachable* if there exists a sequence of transition relations that leads a probabilistic time Petri net from one of its initial states to that particular one. When considering a given system, one might want to express the fact that certain unwanted events are unlikely to happen when it operates. If that system is modelled as a probabilistic time Petri net, those unwanted events are formally represented by a certain set of states. Proving whether a given set of states can be reached with a certain probability or not is at the core of the *probabilistic real-time reachability problem for probabilistic time Petri nets*. Quantitative reachability properties enable us to assert that the probability of reaching certain unwanted states is sufficiently small and that the probability of achieving a certain desired system behaviour is above a given threshold.

We artificially introduce $(d_{\mathcal{N}}, \mathrm{t}_{\mathcal{N}})$ as an action that the probabilistic time Petri net $\mathcal{N}$ performs when it decides that it will never fire a transition again. We let $d_{\mathcal{N}}$ be a real number that is strictly greater than the greatest real endpoint of any firing interval in $\{I(t) \mid t \in T\}$ and let $\mathrm{t}_{\mathcal{N}}$ be a fictitious trial that does not belong to the set T. Intuitively, we want $(d_{\mathcal{N}}, \mathrm{t}_{\mathcal{N}})$ to be a feasible action whenever the firing intervals of the transitions enabled in the current state of the net have no upper bound.

Subsequently, we define the extended set $\tilde{\Phi}(M, v)$ of actions that are feasible from a given state (M, v) by setting $\tilde{\Phi}(M, v) = \Phi(M, v) \cup \{(d_{\mathcal{N}}, \mathrm{t}_{\mathcal{N}})\}$ if $\mathscr{C}(M, v) = \mathbb{R}_+$, and $\tilde{\Phi}(M, v) = \Phi(M, v)$ otherwise. We let $\tilde{T} = T \cup \{\mathrm{t}_{\mathcal{N}}\}$ denote the *extended set of trials* and extend the domain of the partial piecewise probabilistic transition function W to take $(d_{\mathcal{N}}, \mathrm{t}_{\mathcal{N}})$ into account as follows:

$$W((M, v + d_{\mathcal{N}}), \mathrm{t}_{\mathcal{N}}) = \delta_{(M, v+d_{\mathcal{N}})}.$$

3.1 Paths and schedulers

The possible evolution of a probabilistic time Petri net is described formally by a *path*. Reasoning about probabilities of sets of paths relies on the resolution of non-determinism, which is performed by a *scheduler*. The paths describe the potential computations that are obtained by resolving both the non-deterministic and probabilistic choices in the underlying probabilistic timed transition system. In other words, a path is a sequence of trials that are performed at certain dates. These trials carry the net over a set of states.

Definition 6 (Path in a probabilistic timed transition system). *Let $S_{\mathcal{N}} = (Q, \rho_{S_{\mathcal{N}}}, T, W)$ be the semantics of a marked probabilistic time Petri net $\mathcal{N} = (P, T, Pre, Post, I, \rho_{\mathcal{N}})$.*

- *A finite* path *in the probabilistic timed transition system $S_{\mathcal{N}}$ is a finite sequence*

$$q_0 \xrightarrow{d_1, t_1} q_1 \xrightarrow{d_2, t_2} \dots \xrightarrow{d_n, t_n} q_n$$

where $q_0 \in Supp(\rho_{S_\mathcal{N}})$, $n \in \mathbb{N}$ *and for all* $i \in [\![0, n-1]\!]$,

$$\begin{cases} q_i = (M_i, v_i) \in Q, \\ (d_{i+1}, t_{i+1}) \in \tilde{\Phi}(M_i, v_i), \\ q_{i+1} \in Supp(W((M_i, v_i + d_{i+1}), t_{i+1})). \end{cases}$$

The integer n *is called the* length *of the path. A finite path in* $S_\mathcal{N}$ *is an element of* $Supp(\rho_{S_\mathcal{N}}) \times ((\mathbb{R}_+ \times T) \times Q)^*$. *We denote by* $Path^*_{(S_\mathcal{N})}$ *the set of finite paths in the probabilistic timed transition system* $S_\mathcal{N}$.

– *An infinite* path *in the probabilistic timed transition system* $S_\mathcal{N}$ *is an infinite sequence*

$$q_0 \xrightarrow{d_1,t_1} q_1 \xrightarrow{d_2,t_2} q_2 \xrightarrow{d_3,t_3} \dots$$

such that $q_0 \xrightarrow{d_1,t_1} q_1 \xrightarrow{d_2,t_2} \dots \xrightarrow{d_n,t_n} q_n \in Path^*_{(S_\mathcal{N})}$ *for all* $n \in \mathbb{N}$.
An infinite path in $S_\mathcal{N}$ *is an element of* $(Q \times (\mathbb{R}_+ \times T))^\infty$. *We denote by* $Path^\infty_{(S_\mathcal{N})}$ *the set of infinite paths in the probabilistic timed transition system* $S_\mathcal{N}$.

The resolution of all non-deterministic choices in a probabilistic time Petri net is described formally by a scheduler. A scheduler chooses a feasible action $\tilde{\Phi}(M, v)$ in any state (M, v) of the net, but does not have any influence on the probability that one marking or another will be reached once that action has been chosen.

Definition 7 (Scheduler for a probabilistic timed transition system). *Let* $S_\mathcal{N} = (Q, \rho_{S_\mathcal{N}}, T, W)$ *be the semantics of a marked probabilistic timed transition system* $\mathcal{N} = (P, T, Pre, Post, I, \rho_\mathcal{N})$.

For a given finite path $\pi = q_0 \xrightarrow{d_1,t_1} q_1 \xrightarrow{d_2,t_2} \dots \xrightarrow{d_n,t_n} q_n$ *in* $S_\mathcal{N}$, *let* $last(\pi)$ *denote the state* q_n.

– *A* scheduler *for the probabilistic timed transition system* $S_\mathcal{N}$ *is a (total) function* $\mathfrak{S} : Path^*_{(S_\mathcal{N})} \to (\mathbb{R}_+ \times T) \cup \{(d_\mathcal{N}, t_\mathcal{N})\}$ *such that for all finite paths* π *in* $S_\mathcal{N}$

$$\begin{cases} \mathscr{C}(last(\pi)) \neq \mathbb{R}_+ \Rightarrow \mathfrak{S}(\pi) \in \Phi(last(\pi)), \\ \mathscr{C}(last(\pi)) = \mathbb{R}_+ \Rightarrow \mathfrak{S}(\pi) \in \tilde{\Phi}(last(\pi)). \end{cases}$$

A finite or infinite path $\pi = q_0 \xrightarrow{d_1,t_1} q_1 \xrightarrow{d_2,t_2} \dots$ *of* $S_\mathcal{N}$ *is called a* $\mathfrak{S}$-path *if* $\mathfrak{S}(\pi_{|i}) = (d_{i+1}, t_{i+1})$ *for all prefixes* $\pi_{|i}$ *(the path* $\pi_{|i}$ *denotes the finite prefix of* π *of length* i*). We let* $Path^*_\mathfrak{S}$ *denote the (countable) set of finite* $\mathfrak{S}$*-paths.*

The behaviour of a probabilistic time Petri net that is subject to a scheduler $\mathfrak{S}$ can be formalised by a *Markov chain* [15]. Intuitively, this Markov chain unfolds the net into as many trees as there are elements in $Supp(\rho_\mathcal{N})$.

Definition 8 (Markov chain of a PTPN induced by a scheduler). *Let* $S_\mathcal{N} = (Q, \rho_{S_\mathcal{N}}, T, W)$ *be the semantics of a marked probabilistic time Petri net* $\mathcal{N} = (P, T, Pre, Post, I, \rho_\mathcal{N})$.

A scheduler $\mathfrak{S}$ *of* $S_\mathcal{N}$ *induces a Markov chain* $\mathscr{M}_\mathfrak{S} = (Path^*_\mathfrak{S}, \rho_\mathfrak{S}, \mathbf{P}_\mathfrak{S})$ *where*

- $\rho_{\mathfrak{S}} \in \mathscr{D}ist_{Path^*_{\mathfrak{S}}}$ *is the distribution of initial paths of the chain. Its support is equal to the finite paths in* $S_{\mathcal{N}}$ *of length* 0 *that are also initial states of the probabilistic timed transition system* $S_{\mathcal{N}}$.
  For all $(M_0, 0_T) \in Supp(\rho_{\mathfrak{S}})$, $\rho_{\mathfrak{S}}((M_0, 0_T)) = \rho_{S_{\mathcal{N}}}(M_0, 0_T) = \rho_{\mathcal{N}}(M_0)$.
- $\mathbf{P}_{\mathfrak{S}} : Path^*_{\mathfrak{S}} \to \mathscr{D}ist_{Path^*_{\mathfrak{S}}}$ *is the (total) probabilistic transition function of* $\mathscr{M}_{\mathfrak{S}}$. *For* $\lambda \in Path^*_{\mathfrak{S}}$, *the support of* $\mathbf{P}_{\mathfrak{S}}(\lambda)$ *is the set of* $\mathfrak{S}$*-paths of the form* $\pi \xrightarrow{\mathfrak{S}(d,t)} q$. *For* $(\pi, q) \in Path^*_{\mathfrak{S}} \times Q$,

$$\mathbf{P}_{\mathfrak{S}}(\pi)(\pi \xrightarrow{\mathfrak{S}(d,t)} q) = \tilde{\mu}_t(q).$$

Example 2. Let us consider the marked probabilistic time Petri net depicted in Fig. 4, whose initial marking is given by $\rho_{\mathcal{N}} = \delta_{(1,0,0,1,0,0,0)}$. Since the enabled transition T_4 is not firable before date 2, all paths in $\mathcal{N}_1$ start with the resolution of the trial T_1, which either generates a token in P_2 or in P_3. Every scheduler must first choose when to fire that transition. Depending on the outcome of the trial T_1, a scheduler is not presented with the same opportunities. Let us consider a scheduler $\mathfrak{S}_1$ that chooses to fire T_1 immediately. If a token ends up in P_2, then $\mathfrak{S}_1$ is constrained by the deterministic trial T_3 which ends up being performed at date 1. If a token ends up in P_3, then $\mathfrak{S}_1$ must let time flow before performing either T_2 or T_4.

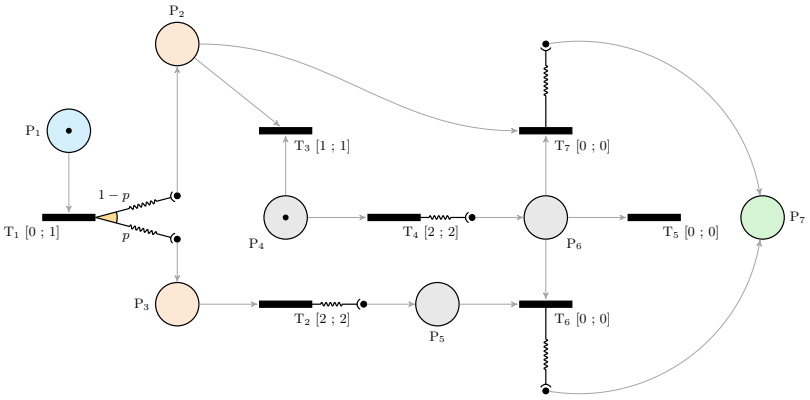

Fig. 4. The probabilistic time Petri net $\mathcal{N}_1$

Suppose that we are interested in reaching the place P_7. Our target set consists of every marking M for which $M(\mathrm{P}_7) > 0$. Figure 5 depicts the choices scheduler $\mathfrak{S}_1$ makes as it resolves all non-determinism before reaching P_7 with probability p. While scheduler $\mathfrak{S}_1$ does exhibit a path leading to a target marking, we would like to thoroughly study the likelihood of reaching those particular markings.

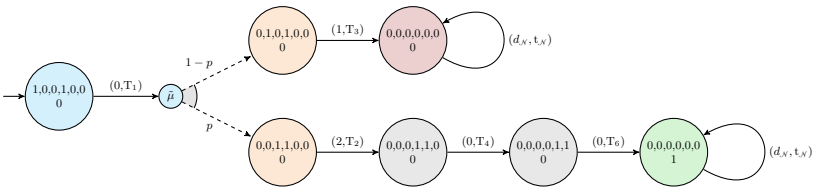

Fig. 5. Abridged representation of the scheduler $\mathfrak{S}_1$

Intuitively, the deterministic trials T_3 and T_5 must be avoided at all costs if P_7 is to be reached. This stems from the fact that these trials eliminate the tokens that are needed to fire T_6 or T_7. To avoid T_3, the trial T_1 must necessarily be resolved at date 1 and no sooner than that. To avoid T_5, the trial T_1 must necessarily be resolved at date 0, without delay. Schedulers that do not fire T_1 at date 0 or at date 1 never reach P_7. Therefore, the minimum probability of reaching P_7 is 0. Since a scheduler has no influence over the outcome of T_1, it has no way of knowing if firing T_1 at date 0 or at date 1 is best. As a result, the probability of reaching P_7 can be no greater than $max(p, 1-p)$.

The probabilistic real-time reachability problem consists in the establishment of these lower and upper probability bounds. The whole set of schedulers of a probabilistic time Petri net is considered in order to compute these bounds, as they cover every possible resolution of non-determinism. This corresponds to a worst-case analysis.

3.2 Markov decision processes

Since a probabilistic time Petri net evolves in a dense-time environment, there are usually infinitely many schedulers as soon as a single firing interval is a proper interval. To compute the lower and upper probability bounds by ranging over all schedulers, we proceed to a natural grouping of states that leads to the formation of *state classes*. This enables us to capture the information we need in a finite graph, called a state class graph, which can then be used to apply formal verification techniques. The state space of the net thus takes the form of a Markov decision process. Formally:

Definition 9 (Markov decision process (MDP)). *A* Markov decision process *is a quadruple* $(C, \rho, A, \mathbf{P})$ *where*

- C *is the (countable) set of states of the process,*
- $\rho \in \mathscr{D}ist_C$ *is the distribution of initial states of the process,*
- A *is the set of actions of the process, and*
- $\mathbf{P} : C \times A \to \mathscr{D}ist_C$ *is the (partial) probabilistic transition function.*

For a given state c of a Markov decision process, we define the set $\Sigma(c)$ of actions that are enabled in the state c as

$$\Sigma(c) = \{\alpha \in A \mid \mathbf{P}(c, \alpha) \text{ is defined}\}.$$

The assumption that $\Sigma(c) \neq \emptyset$ for all $c \in C$ is a conventional requirement in the literature that is not specific to our setting [15].

As for probabilistic time Petri nets, the paths in a Markov decision process resolve both probabilistic and non-deterministic choices.

Definition 10 (Path in a Markov decision process). *Let $\mathscr{M} = (C, \rho, A, \mathbf{P})$ be a Markov decision process.*

- *A finite* path *in the Markov decision process $\mathscr{M}$ is a finite sequence*

$$c_0 \xrightarrow{\alpha_1} c_1 \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_n} c_n$$

where for all $i \in [\![0, n-1]\!]$,

$$\begin{cases} c_i \in C, \\ \alpha_{i+1} \in \Sigma(c_i), \\ c_{i+1} \in Supp(\mathbf{P}(c_i, \alpha_{i+1})). \end{cases}$$

The integer n is called the length *of the path. A finite path in $\mathscr{M}$ is an element of $Supp(\rho) \times (A \times C)^*$. We denote by $Path^*_{(\mathscr{M})}$ the set of finite paths in the Markov decision process $\mathscr{M}$.*

- *An infinite* path *in the Markov decision process $\mathscr{M}$ is an infinite sequence*

$$c_0 \xrightarrow{\alpha_1} c_1 \xrightarrow{\alpha_2} c_2 \xrightarrow{\alpha_3} \dots$$

where $c_0 \xrightarrow{\alpha_1} c_1 \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_n} c_n \in Path^*_{(\mathscr{M})}$ for all $n \in \mathbb{N}$.
An infinite path in $\mathscr{M}$ is an element of $(C \times A)^\infty$. We denote by $Path^\infty_{(\mathscr{M})}$ the set of infinite paths in the Markov decision process $\mathscr{M}$.

An adversary of a Markov decision process fulfils the same function a scheduler does for a probabilistic time Petri net.

Definition 11 (Adversary of a Markov decision process). *Let $\mathscr{M} = (C, \rho, A, \mathbf{P})$ be a Markov decision process.*

For a given finite path $\sigma = c_0 \xrightarrow{\alpha_1} c_1 \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_n} c_n$ in $\mathscr{M}$, let $last(\sigma)$ denote the state c_n.

- *An* adversary *of the Markov decision process $\mathscr{M}$ is a (total) function $\Lambda : Path^*_{(\mathscr{M})} \to A$ such that for all finite paths $\sigma = c_0 \xrightarrow{\alpha_1} c_1 \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_n} c_n$ in $Path^*_{(\mathscr{M})}$*

$$\Lambda(\sigma) \in \Sigma(last(\sigma)).$$

*A finite or infinite path $\sigma = c_0 \xrightarrow{\alpha_1} c_1 \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_n} c_n$ of $\mathscr{M}$ is called a Λ-*path *if $\Lambda(\sigma_{|i}) = \alpha_{i+1}$ for all prefixes $\sigma_{|i}$ of σ (the path $\sigma_{|i}$ denotes the finite prefix of σ of length i). We let $Path^*_\Lambda$ denote the (countable) set of finite Λ-paths.*

– *An adversary Λ of the Markov decision process $\mathscr{M}$ induces a Markov chain $\mathscr{M}_\Lambda = (Path^*_\Lambda, \rho_\Lambda, \mathbf{P}_\Lambda)$ where*
  - *ρ_Λ is the distribution of initial paths of the chain. Its support is equal to the finite paths in $\mathscr{M}$ of length 0 that are also initial states of the process. For all $c \in Supp(\rho_\Lambda)$, $\rho_\Lambda(c) = \rho(c)$.*
  - *$\mathbf{P}_\Lambda : Path^*_\Lambda \to \mathscr{D}ist_{Path^*_\Lambda}$ is the (total) probabilistic transition function of $\mathscr{M}_\Lambda$. For $\sigma \in Path^*_\Lambda$, the support of $\mathbf{P}_\Lambda(\sigma)$ is the set of Λ-paths of the form $\sigma \xrightarrow{\Lambda(\sigma)} c$. For $(\sigma, c) \in Path^*_\Lambda \times C$,*

$$\mathbf{P}_\Lambda(\sigma)(\sigma \xrightarrow{\Lambda(\sigma)} c) = \mathbf{P}(last(\sigma), \Lambda(\sigma))(c).$$

3.3 The probabilistic strong state class graph

Time Petri nets typically generate an infinite state space. The *linear state class graph* was introduced in [11] and [12] in order to capture linear time temporal properties of time Petri nets in a finite graph. Intuitively, each class is an element of $\mathbb{N}^P \times \mathscr{P}(\mathbb{R}^T)$ which captures all the states that are reachable from an initial state class by firing schedules of a given support. Since there are generally infinitely many supports, state classes are then considered modulo some equivalence relation. The graph thus becomes *finite* if the net is *bounded*.

The probabilistic strong state class graph extends the construction methods that are proposed in the literature to account for the probabilistic nature of PTPNs. The following definition introduces *strong state classes* for probabilistic time Petri nets and details how the successor of a class is obtained when firing a given transition.

Definition 12 (Strong state classes). *Let $S_\mathcal{N} = (Q, \rho_{S_\mathcal{N}}, T, W)$ be the semantics of a marked probabilistic time Petri net $\mathcal{N} = (P, T, Pre, Post, I, \rho_\mathcal{N})$. The set of* strong state classes *is defined as follows:*

1. *For a given transition t of the net $\mathcal{N}$, we define the set $\Delta(t)$ of* decoupled trials *of t as $\Delta(t) = \left\{t_\omega \in \mathscr{D}ist_{\mathbb{N}^P} \mid \exists \omega \in Supp(\mu_t), t_\omega = \delta_\omega\right\}$ and denote by $T_\Delta = \bigcup_{t \in T} \Delta(t)$ the set of decoupled trials in $S_\mathcal{N}$.*
2. *For a given initial state $q_0 \in Supp(\rho_{S_\mathcal{N}})$, we define a cover $C_{q_0} = \bigcup_{\tau \in T^*_\Delta} c_\tau$ of Q inductively by $c_\varepsilon = \{q_0\}$ and*

$$c_{\tau t_\omega} = \left\{ \begin{array}{l} (M', v') \in \mathbb{N}^P \times \mathbb{R}_+^T \mid \exists (M, v) \in c_\tau, \exists (d, t) \in \Phi(M, v), t_\omega \in \Delta(t) \\ \quad \text{and } \forall t' \in T, v'(t') = (v(t') + d) \times (1 - \chi_t(t')) \times \chi_{\mathcal{E}(M - Pre(t))}(t') \\ \quad \text{and } M' = (M - Pre(t)) + \omega \end{array} \right\}$$

   The classes $c_{\tau t_\omega}$ are the successors *of the state class c_τ.*
3. *The cover C_{q_0} denotes the set of nodes of the tree that is generated by q_0. We must account for all the trees that are generated by an initial state of the net and thus let*

$$C = \bigcup_{q_0 \in Supp(\rho_{S_\mathcal{N}})} C_{q_0}.$$

Let $c \in C$ be a state class in which the shared marking is M. We say that a transition $t \in \mathscr{E}(M)$ is *firable* from the state class c if there exists a state $q \in c$ such that t is firable from q. The *probabilistic strong state class graph* (which remains *finite* if the net is *bounded*) is defined as follows:

Definition 13 (Probabilistic strong state class graph). *Let $S_{\mathcal{N}} = (Q, \rho_{S_{\mathcal{N}}}, T, W)$ be the semantics of a marked probabilistic time Petri net $\mathcal{N} = (P, T, Pre, Post, I, \rho_{\mathcal{N}})$. The* probabilistic strong state class graph *of the net $\mathcal{N}$ is a Markov decision process $\mathscr{M} = (C, \rho, \tilde{T}, \mathbf{P})$ where*

- *C is the set of* strong state classes*,*
- *$\rho : C \to [0,1]$ is the distribution of initial classes of the graph. The support of ρ is equal to the set of singletons $\{q_0\}$, where $q_0 \in Supp(\rho_{S_{\mathcal{N}}})$. For all $(M_0, 0_T) \in Supp(\rho_{S_{\mathcal{N}}})$,*

$$\rho\big(\{(M_0, 0_T)\}\big) = \rho_{S_{\mathcal{N}}}(M_0, 0_T) = \rho_{\mathcal{N}}(M_0).$$

- *$\mathbf{P} : C \times \tilde{T} \to \mathscr{D}ist_C$ is the (partial) transition probability function.*
  1. *$\mathbf{P}$ is defined for $(c,t) \in C \times T$ if and only if the transition t is firable from the state class c. In that case, let $\mathbf{P}(c,t) = \widehat{\mu}_t$, where $\widehat{\mu}_t \in \mathscr{D}ist_C$ is defined as follows:*
     - *Let $c' \in C$. The class c' lies in $Supp(\widehat{\mu}_t)$ if and only if c' is the successor of c for some decoupled transition $t_\omega \in \Delta(t)$.*
     - *Suppose that $c' \in Supp(\widehat{\mu}_t)$. We define the image of c' by the formula*

$$\widehat{\mu}_t(c') = \mu_t(\omega).$$

  2. *$\mathbf{P}$ is defined for $(c,t) \in C \times \{t_{\mathcal{N}}\}$ if and only if $\mathscr{C}(q) = \mathbb{R}_+$ for some $q \in c$. In that case, $\mathscr{C}(q) = \mathbb{R}_+$ for all $q \in c$ since all the states in c have the same marking. We define the image of $(c, t_{\mathcal{N}})$ by the formula*

$$\mathbf{P}(c, t_{\mathcal{N}}) = \delta_c.$$

The probabilistic strong state class graph of the probabilistic time Petri net $\mathcal{N}_1$ is represented in Fig. 6. Each class is represented by a node, which is labelled with the marking that all states share in that particular class. Here, strong state classes are considered modulo an equivalence relation $\equiv$ that asserts that two classes are equivalent if they denote the same set of states. For the sake of clarity, time domains are not represented.

Let us now consider the adversary Λ_1 of the probabilistic state class graph of $\mathcal{N}_1$, depicted in Fig. 7. Depending on the outcome of the trial T_1, it either performs the untimed sequence of actions $\mathrm{T}_1 \to \mathrm{T}_4 \to \mathrm{T}_7$ or the untimed sequence $\mathrm{T}_1 \to \mathrm{T}_2 \to \mathrm{T}_4 \to \mathrm{T}_6$ before reaching a target state. However, there is no scheduler for $\mathcal{N}_1$ that can perform both of these paths since the path $\mathrm{T}_1 \to \mathrm{T}_4 \to \mathrm{T}_7$ can only be performed when T_1 is fired at date 1 while the path $\mathrm{T}_1 \to \mathrm{T}_2 \to \mathrm{T}_4 \to \mathrm{T}_6$ can only be performed when T_1 is fired at date 0. As a result, the probabilistic strong state class graph potentially generates duplicitous adversaries, which display a probability of reaching target states greater than that of any scheduler.

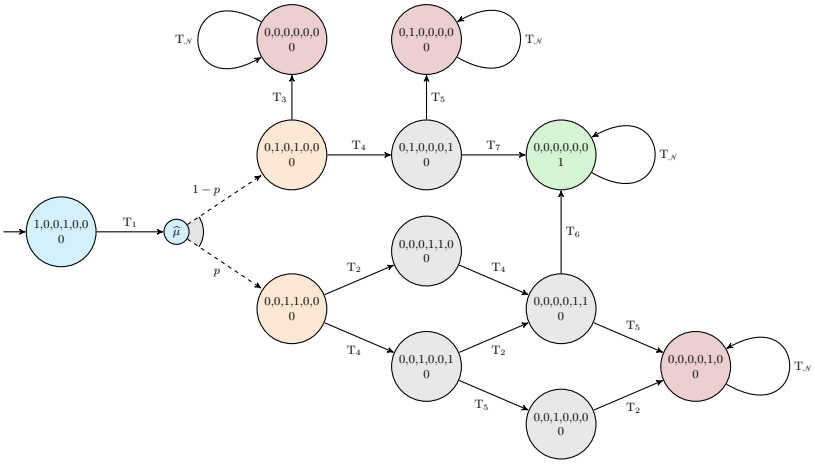

Fig. 6. The probabilistic strong state class graph of the probabilistic time Petri net $\mathcal{N}_1$

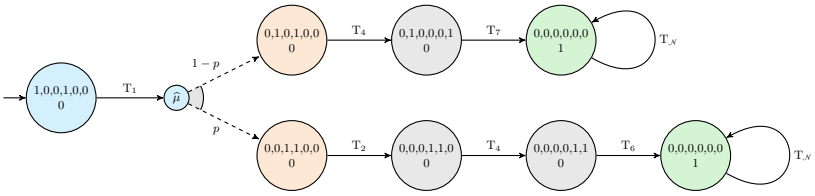

Fig. 7. Abridged representation of the duplicitous adversary Λ_1

3.4 The probabilistic atomic state class graph

The reason why the probabilistic strong state class graph fails to produce proper adversaries lies in the way time and probabilities are intertwined in probabilistic time Petri nets. A graph that better captures the effect the firing date of probabilistic trials has on future actions is needed in order to solve the probabilistic real-time reachability problem.

Berthomieu and Vernadat introduced the atomic state class graph for time Petri nets in order to preserve their branching time temporal properties in a finite graph [14]. The construction of this graph can be adapted for probabilistic time Petri nets to preserve the adversaries we need. Let us consider the following properties of interest for state class graphs:

– (EE) For all classes $c, c' \in \mathbb{N}^P \times \mathscr{P}(\mathbb{R}^T)$ and for all $t \in \Sigma(c)$,

$$c \xrightarrow{t} c' \in Path^*_{(\mathscr{M})} \iff \exists q \in c, \exists q' \in c', \exists d \in \mathbb{R}_+, \begin{cases} (d,t) \in \Phi(q) \\ q \xrightarrow{(d,t)} q' \in Path^*_{(S_{\mathcal{N}})} \end{cases}$$

– (AE) For all classes $c, c' \in \mathbb{N}^P \times \mathscr{P}(\mathbb{R}^T)$ and for all $t \in \Sigma(c)$,

$$c \xrightarrow{t} c' \in Path^*_{(\mathscr{M})} \Longrightarrow \forall q \in c, \exists q' \in c', \exists d \in \mathbb{R}_+, \begin{cases} (d,t) \in \Phi(q) \\ q \xrightarrow{(d,t)} q' \in Path^*_{(S_{\mathcal{N}})} \end{cases}$$

State class graphs typically satisfy property (EE) and so does the probabilistic strong state class graph. The *probabilistic atomic state class graph* we introduce in this section is built from the probabilistic strong state class graph, by refining its classes into *atomic* ones. An atomic class is a state class in which each state has a successor in each of the successors of that class. Intuitively, each atomic class captures all the states that are reachable from an initial state by firing schedules of a given support *during certain time windows*.

The algorithm that details how to split strong state classes into atomic ones can be found in [14]. Splitting a class c replaces it with a pair of classes which both inherit the predecessors of c and the successors of c that they can still reach. This technically causes multiple hyperarcs leaving the predecessors of c to have the same label. However, each one of these hyperarcs is implicitly augmented with a time interval. This time window corresponds to the set of delays that enforce property (EE) in each one of the states it leads to. Since no time delay is shared among those hyperarcs, any ambiguity is lifted.

This stable refinement enforces property (AE) in the probabilistic atomic state class graph, which once again takes the form of a Markov decision process $\mathscr{M}_A = (C_A, \rho_A, \tilde{T}, \mathbf{P}_A)$. However, this graph is usually significantly bigger than the probabilistic state class graph from which is it built. The probabilistic atomic state class graph of the probabilistic time Petri net $\mathcal{N}_1$ is represented in Fig. 8.

The proof of the following theorem is omitted due to lack of space.

Theorem 1. *Let $\mathcal{N} = (P, T, Pre, Post, I, \rho_{\mathcal{N}})$ be a bounded marked probabilistic time Petri net, let $S_{\mathcal{N}} = (Q, \rho_{S_{\mathcal{N}}}, T, W)$ be its semantics and let $\mathscr{M}_A = (C_A, \rho_A, \tilde{T}, \mathbf{P}_A)$ be the probabilistic atomic state class graph of $\mathcal{N}$.*

1. *Let Λ be an adversary of the Markov decision process $\mathscr{M}_A$. There exists a scheduler for the probabilistic timed transition system $S_{\mathcal{N}}$ that induces the same Markov chain as Λ up to isomorphism.*
2. *Conversely, let $\mathfrak{S}$ be a scheduler for the probabilistic timed transition system $S_{\mathcal{N}}$. There exists an adversary of the Markov decision process $\mathscr{M}_A$ that induces the same Markov chain as $\mathfrak{S}$ up to isomorphism.*

As a result of theorem 1, the probabilistic real-time reachability problem can be solved by computing the probability of reaching a target state for every adversary of the probabilistic atomic state class graph (with the tools commonly used for Markov decision processes). For example, it can easily be shown that the sought probability bounds for reaching P7 in the net $\mathcal{N}_1$ (Fig. 4) are indeed 0 and $max(p, 1-p)$, by considering all the adversaries of its probabilistic atomic state class graph (Fig. 8). In fact, an array of algorithms can now be used to prove that the net verifies the following properties:

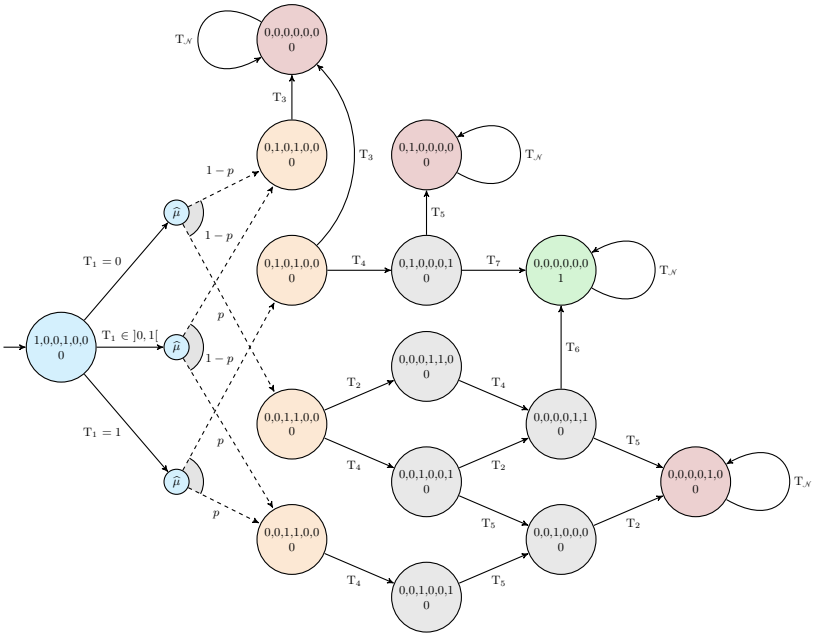

Fig. 8. The probabilistic atomic state class graph of the PTPN $\mathcal{N}_1$

- **reachability**: the net $\mathcal{N}_1$ can reach P_7 with probability (at least) 0.5,
- **inevitability**: the net $\mathcal{N}_1$ inevitably leaves P_1 with probability 1,
- **time bounded reachability**: the net $\mathcal{N}_1$ can reach P_7 within two time units with probability 0.5,
- **bounded response**: the net $\mathcal{N}_1$ inevitably reaches P_5 or P_7 within two time units with probability 1 after reaching the marking $(0, 0, 1, 1, 0, 0, 0)$.

4 Conclusion

We have introduced a new formalism for the modelling of concurrent probabilistic real-time systems. This new model extends time Petri nets by enhancing the forward incidence mapping with sets of probability distributions. Probabilistic time Petri nets natively integrate time, concurrency and probabilities. In the spirit of probabilistic timed automata [16], we have restricted all random phenomena to the discrete behaviour of a time Petri net. Time and concurrency are still resolved in a non-deterministic manner. We have shown how the atomic state class graph construction of TPNs can be adapted to our model and how this enables us to recover a Markov decision process that induces the same Markov chains as the semantics of the PTPN. Therefore, the use of a wide range of tools for the analysis of PTPN is made available to us. We have also proved that the

simpler non-atomic state class graph construction cannot be adapted in a similar manner.

Future work includes the addition of timing and probability parameters in probabilistic time Petri nets, the implementation of the proposed method in our tool Roméo and the application of this model to the automotive industry.

References

1. W. J. Stewart, *Introduction to the numerical solutions of Markov chains*. Princeton Univ. Press, 1994.
2. M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, Inc., 1994.
3. M. Kwiatkowska, G. Norman, R. Segala, and J. Sproston, "Automatic verification of real-time systems with discrete probability distributions," *Theoretical Computer Science*, vol. 282, no. 1, pp. 101–150, 2002.
4. C. Eisentraut, H. Hermanns, and L. Zhang, "On probabilistic automata in continuous time," in *Logic in Computer Science (LICS), 2010 25th Annual IEEE Symposium on*, pp. 342–351, IEEE, 2010.
5. N. Bertrand, P. Bouyer, Th. Brihaye, Q. Menet, C. Baier, M. Größer, and M. Jurdziński, "Stochastic timed automata," *Logical Methods in Computer Science*, vol. 10, no. 4:6, 2014.
6. 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*, vol. 14, no. 3, pp. 295–318, 2003.
7. M. Kwiatkowska, G. Norman, and D. Parker, "Prism 4.0: Verification of probabilistic real-time systems," in *Computer aided verification*, vol. 6806 of *LNCS*, pp. 585–591, 2011.
8. M. A. Marsan, G. Conte, and G. Balbo, "A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems," *ACM Trans. Comput. Syst.*, vol. 2, no. 2, pp. 93–122, 1984.
9. M. K. Molloy, "Discrete time stochastic Petri nets," *IEEE Trans. Software Eng.*, vol. 11, no. 4, pp. 417–423, 1985.
10. E. Vicario, L. Sassoli, and L. Carnevali, "Using stochastic state classes in quantitative evaluation of dense-time reactive systems," *IEEE Trans. Software Eng.*, vol. 35, no. 5, pp. 703–719, 2009.
11. B. Berthomieu and M. Diaz, "Modeling and verification of time dependent systems using time Petri nets," *IEEE trans. on Soft. Eng.*, no. 3, pp. 259–273, 1991.
12. B. Berthomieu and M. Menasche, "An enumerative approach for analyzing time Petri nets," in *Information Processing: proceedings of the IFIP congress 1983* (R. E. A. Mason, ed.), vol. 9 of *IFIP congress series*, pp. 41–46, 1983.
13. J. B. Dugan, K. S. Trivedi, R. M. Geist, and V. F. Nicola, "Extended stochastic Petri nets: Applications and analysis.," tech. rep., DTIC Document, 1984.
14. B. Berthomieu and F. Vernadat, "State class constructions for branching analysis of time Petri nets," in *TACAS'2003*, vol. 2619 of *LNCS*, pp. 442–457, Springer, 2003.
15. C. Baier and J. Katoen, *Principles of model checking*. MIT Press, 2008.
16. G. Norman, D. Parker, and J. Sproston, "Model checking for probabilistic timed automata," *Formal Methods in System Design*, vol. 43, no. 2, pp. 164–190, 2013.

A Appendix

Lemma 1. *Let $\mathcal{N} = (P, T, Pre, Post, I)$ be a probabilistic time Petri net. For a given trial $t \in T$, the function μ_t is a discrete probability distribution over $\Omega(t)$.*

Proof. Let us consider a trial t of $\mathcal{N}$.

- For each $f \in \mathscr{A}(t)$ and each $\mu \in Post(t)$, the marking $f(\mu)$ lies in $Supp(\mu)$, hence $\mu(f(\mu)) \in [0,1]$ and $\mu_t(\omega) > 0$ for all $\omega \in \Omega(t)$.
- Let us now show that
$$\sum_{\omega \in \Omega(t)} \mu_t(\omega) = 1.$$
  - The family of subsets $\{\mathscr{A}_\omega(t)\}_{\omega \in \Omega(t)}$ of $\mathscr{A}(t)$ defines a partition of the set $\mathscr{A}(t)$. Therefore,
$$\sum_{\omega \in \Omega(t)} \left[\sum_{f \in \mathscr{A}_\omega(t)} \left(\prod_{\mu \in Post(t)} \mu\big(f(\mu)\big) \right) \right] = \sum_{f \in \mathscr{A}(t)} \left(\prod_{\mu \in Post(t)} \mu\big(f(\mu)\big) \right).$$
  - An element of $\mathscr{A}(t)$ is characterised by its graph, which consists of $|Post(t)|$ *independent* choices. Each one of these choices corresponds to a probabilistic experiment by means of the discrete probability distributions that make up $Post(t)$. Consequently, the function
$$\begin{aligned} \mu_\times : \mathscr{A}(t) &\longrightarrow [0,1] \\ f &\longmapsto \prod_{\mu \in Post(t)} \mu\big(f(\mu)\big) \end{aligned}$$
  is a discrete probability distribution over $\mathscr{A}(t)$ and
$$\sum_{\omega \in \Omega(t)} \mu_t(\omega) = \sum_{f \in \mathscr{A}(t)} \mu_\times(f) = 1$$

□

Theorem 1. *Let $\mathcal{N} = (P, T, Pre, Post, I, \rho_\mathcal{N})$ be a bounded marked probabilistic time Petri net, let $S_\mathcal{N} = (Q, \rho_{S_\mathcal{N}}, T, W)$ be its semantics and let $\mathscr{M}_A = (C_A, \rho_A, \tilde{T}, \mathbf{P}_A)$ be the probabilistic atomic state class graph of $\mathcal{N}$.*

1. *Let Λ be an adversary of the Markov decision process $\mathscr{M}_A$. There exists a scheduler for the probabilistic timed transition system $S_\mathcal{N}$ that induces the same Markov chain as Λ up to isomorphism.*
2. *Conversely, let $\mathfrak{S}$ be a scheduler for the probabilistic timed transition system $S_\mathcal{N}$. There exists an adversary of the Markov decision process $\mathscr{M}_A$ that induces the same Markov chain as $\mathfrak{S}$ up to isomorphism.*

Proof. 1. For a given adversary Λ of the Markov decision process $\mathscr{M}_A$, let us define a scheduler $\mathfrak{S} : Path^*_{(S_\mathcal{N})} \to (\mathbb{R}_+ \times T) \cup \{(d_\mathcal{N}, \mathrm{t}_\mathcal{N})\}$ for the probabilistic timed transition system $S_\mathcal{N}$ as follows:

– Let $\pi = q_0 \xrightarrow{d_1,t_1} \dots \xrightarrow{d_n,t_n} q_n \in Path^*_{(S_{\mathcal{N}})}$ be a finite path in the probabilistic timed transition system $S_{\mathcal{N}}$. According to property (EE), there exists a path $\sigma = c_0 \xrightarrow{t_1} \dots \xrightarrow{t_n} c_n \in Path^*_{(\mathcal{M}_A)}$ in the probabilistic atomic state class graph $\mathcal{M}_A$ such that $q_i \in c_i$ for all $i \in [\![0, n]\!]$. Let $q_n = (M_n, v_n)$.
  - If $\Lambda(\sigma) = \mathrm{t}_{\mathcal{N}}$, then $\mathrm{t}_{\mathcal{N}} \in \Sigma(c_n)$. It follows that $\mathscr{C}(q) = \mathbb{R}_+$ for all $q \in c_n$. In particular, $\mathscr{C}(q_n) = \mathbb{R}_+$. We can thus set
  $$\mathfrak{S}(\pi) = (d_{\mathcal{N}}, \mathrm{t}_{\mathcal{N}}).$$
  - If $\Lambda(\sigma) \neq \mathrm{t}_{\mathcal{N}}$, then property (AE) guarantees the existence of a delay $d_{n+1} \in \mathscr{C}(M_n, v_n)$ such that $\Lambda(\sigma) \in \mathscr{F}(M_n, v_n + d_{n+1})$. In that case, $(d_{n+1}, \Lambda(\sigma)) \in \Phi(q_n)$. We can thus set
  $$\mathfrak{S}(\pi) = (d_{n+1}, \Lambda(\sigma)).$$

– We will now demonstrate that the Markov chain $\mathcal{M}_{\mathfrak{S}} = (Path^*_{\mathfrak{S}}, \rho_{\mathfrak{S}}, \mathbf{P}_{\mathfrak{S}})$ induced by $\mathfrak{S}$ and the Markov chain $\mathcal{M}_\Lambda = (Path^*_\Lambda, \rho_\Lambda, \mathbf{P}_\Lambda)$ induced by Λ are the same, up to isomorphism. To do so, we will introduce a bijection ζ that maps the nodes of the chain $\mathcal{M}_{\mathfrak{S}}$ to the nodes of the chain $\mathcal{M}_\Lambda$. We will then show that this mapping is a graph isomorphism by proving that it preserves the structure of $\mathcal{M}_{\mathfrak{S}}$, as defined by the distribution of initial paths $\rho_{\mathfrak{S}}$ and the probabilistic transition function $\mathbf{P}_{\mathfrak{S}}$, in the chain $\mathcal{M}_\Lambda$.
  (a) Let us define the canonical bijection ζ between the $\mathfrak{S}$-paths of $Path^*_{\mathfrak{S}}$ and the Λ-paths of $Path^*_\Lambda$ as follows:
  Let $\pi = q_0 \xrightarrow{d_1,t_1} \dots \xrightarrow{d_n,t_n} q_n$ be a $\mathfrak{S}$-path. According to property EE, there exists a path $\sigma = c_0 \xrightarrow{t_1} \dots \xrightarrow{t_n} c_n \in Path^*_{(\mathcal{M}_A)}$ in the probabilistic atomic state class graph $\mathcal{M}_A$ that verifies $q_i \in c_i$ for all $i \in [\![0, n]\!]$. Since every prefix $\pi_{|i}$ of π is a $\mathfrak{S}$-path, it follows that $\Lambda(\sigma_{|i}) = t_{i+1}$ for all prefixes $\sigma_{|i}$ of σ, by definition of the scheduler $\mathfrak{S}$. This implies that σ is a Λ-path and that such a path is unique for any given π. Let
  $$\zeta : Path^*_{\mathfrak{S}} \to Path^*_\Lambda$$
  be the function that maps each finite $\mathfrak{S}$-path π to its correponding Λ-path σ.
    - Let us prove that ζ is injective. Let $\pi = q_0 \xrightarrow{d_1,t_1} \dots \xrightarrow{d_n,t_n} q_n$ and $\pi' = q'_0 \xrightarrow{d'_1,t'_1} \dots \xrightarrow{d'_n,t'_n} q'_n$ be two Λ-paths, such that $\zeta(\pi) = \zeta(\pi')$. Let $\zeta(\pi) = c_0 \xrightarrow{t_1} \dots \xrightarrow{t_n} c_n$. Since π' is a $\mathfrak{S}$-path and $q'_i \in c_i$ for all $i \in [\![0, n]\!]$, it follows that $(d'_{i+1}, t'_{i+1}) = \mathfrak{S}(\pi_{|i}) = (d_{i+1}, t_{i+1})$ for all $i \in [\![0, n-1]\!]$ and $q_i = q'_i$ for all $i \in [\![0, n]\!]$. Consequently, the equality $\pi = \pi'$ holds. This proves that the function ζ is injective.

- Let us prove that ζ is surjective. Let $\sigma = c_0 \xrightarrow{t_1} \dots \xrightarrow{t_n} c_n$ be a Λ-path. The only $\mathfrak{S}$-path $\pi = q_0 \xrightarrow{d_1,t_1} \dots \xrightarrow{d_n,t_n} q_n$ that verifies $q_i \in c_i$ for all $i \in [\![0, n]\!]$ also verifies $\zeta(\pi) = \sigma$. Consequently, the function ζ is surjective.

(b) For every $q_0 = (M_0, 0_T) \in Supp(\rho_{\mathfrak{S}})$, $\zeta(q_0) = \{q_0\} \in Supp(\rho_\Lambda)$.

- The distribution of initial paths $\rho_{\mathfrak{S}}$ of the Markov chain induced by the scheduler $\mathfrak{S}$ is defined (Def. 8) from the distribution of initial markings $\rho_{S_\mathcal{N}}$ of the probabilistic timed transition system $S_\mathcal{N}$ (Def. 5) as follows:

$$\rho_{\mathfrak{S}}(q_0) = \rho_{S_\mathcal{N}}(q_0) = \rho_\mathcal{N}(M_0).$$

- The distribution of initial markings ρ_Λ of the Markov chain induced by the adversary Λ is defined (Def. 11) from the distribution of initial classes ρ_A of the atomic state class graph, which is the same as the distribution of initial classes of the strong state class graph (Def. 13):

$$\rho_\Lambda(\{q_0\}) = \rho_A(\{q_0\}) = \rho_\mathcal{N}(M_0).$$

Therefore,

$$\rho_{\mathfrak{S}}(q_0) = \rho_\Lambda(\zeta(q_0)).$$

(c) Let π be a Λ-path and $\sigma = \zeta(\pi)$ be the $\mathfrak{S}$-path that is canonically associated with π. Let $q = (M, v) \in \mathbb{N}^P \times \mathbb{R}_+^T$ be a state of $\mathcal{N}$ such that $\pi \xrightarrow{\mathfrak{S}(\pi)} q$ is a $\mathfrak{S}$-path and let $\sigma \xrightarrow{\Lambda(\sigma)} c = \zeta(\pi \xrightarrow{\mathfrak{S}(\pi)} q)$.

- By definition of the partial probabilistic transition function $\mathbf{P}_{\mathfrak{S}}$ (Def. 8) of the scheduler $\mathfrak{S}$ and by definition of the probability distribution of states $\tilde{\mu}_t$ (Def. 5),

$$\begin{aligned}\mathbf{P}_{\mathfrak{S}}(\pi)(\pi \xrightarrow{\mathfrak{S}(\pi)} q) &= \tilde{\mu}_t(q) \\ &= \mu_t(\omega_M).\end{aligned}$$

- By definition of the partial probabilistic transition functions $\mathbf{P}_\Lambda$ of the adversary Λ and $\mathbf{P}_A$ of the atomic state class graph (Def. 11) and by definition of the probability distribution of classes $\widehat{\mu}_t$ (Def. 13),

$$\begin{aligned}\mathbf{P}_\Lambda(\sigma)(\sigma \xrightarrow{\Lambda(\sigma)} c) &= \mathbf{P}_A(last(\sigma), \Lambda(\sigma))(c) \\ &= \widehat{\mu}_t(c) \\ &= \mu_t(\omega_M).\end{aligned}$$

Therefore,

$$\mathbf{P}_{\mathfrak{S}}(\pi)(\pi \xrightarrow{\mathfrak{S}(\pi)} q) = \mathbf{P}_\Lambda(\zeta(\pi))(\zeta(\pi \xrightarrow{\mathfrak{S}(\pi)} q)).$$

This proves that ζ is edge-preserving from a probabilistic standpoint. It follows that ζ is a Markov chain isomorphism. The forest of trees generated by the Markov chains $\mathscr{M}_{\mathfrak{S}}$ and $\mathscr{M}_{\Lambda}$ are therefore identical up to isomorphism.

2. Conversely, let $\mathfrak{S}$ be a scheduler for $S_{\mathcal{N}}$. Let us define an adversary $\Lambda : Path^*_{(\mathscr{M}_A)} \to \tilde{T}$ of the probabilistic atomic state class graph as follows:
   - Let $\sigma = c_0 \xrightarrow{t_1} c_1 \xrightarrow{t_2} \dots \xrightarrow{t_n} c_n \in Path^*_{(\mathscr{M}_A)}$ be a finite path in the probabilistic atomic state class graph $\mathscr{M}_A$. According to property (EE), there exists a path $\pi = q_0 \xrightarrow{d_1,t_1} q_1 \xrightarrow{d_2,t_2} \dots \xrightarrow{d_n,t_n} q_n \in Path^*_{(S_{\mathcal{N}})}$ in the probabilistic timed transition system $S_{\mathcal{N}}$ such that $q_i \in c_i$ for all $i \in [\![0, n]\!]$.
     - If $\mathfrak{S}(\pi) = (d_{\mathcal{N}}, \mathrm{t}_{\mathcal{N}})$, then $\mathscr{C}(q_n) = \mathbb{R}_+$. Therefore $\mathbf{P}_A$ is defined for $(c_n, \mathrm{t}_{\mathcal{N}})$ and $\mathrm{t}_{\mathcal{N}} \in \Sigma(c_n)$ as a result. We can thus set
     $$\Lambda(\sigma) = \mathrm{t}_{\mathcal{N}}.$$
     - If $\mathfrak{S}(\pi) = (d_{n+1}, t_{n+1}) \in \Phi(q_n)$, then the sequence $\sigma.t_{n+1}$ is the support of a path of $Path^*_{(S_{\mathcal{N}})}$. The transition $t_{n+1} \in T$ is therefore firable from the state class c_n and $\mathbf{P}_A$ is defined for (c_n, t_{n+1}). In that case, $t_{n+1} \in \Sigma(c_n)$. Let
     $$\Lambda(\sigma) = t_{n+1}.$$
     Choosing t_{n+1} in $\Sigma(c_n)$ can be ambiguous since there is potentially more than one output hyperarc of c_n that is labelled with t_{n+1} as a result of the splitting process. However, since each one of these hyperarcs is implicitly augmented with a time window that correponds to the set of delays that are allowed before the firing of t_{n+1}, the chosen arc can only be the one that allows a time delay of d_{n+1}.
   - The graph isomorphism introduced in 1. can be used to prove that the Markov chain $\mathscr{M}_{\mathfrak{S}} = (Path^*_{\mathfrak{S}}, \rho_{\mathfrak{S}}, \mathbf{P}_{\mathfrak{S}})$ induced by $\mathfrak{S}$ and the Markov chain $\mathscr{M}_{\Lambda} = (Path^*_{\Lambda}, \rho_{\Lambda}, \mathbf{P}_{\Lambda})$ induced by Λ are the same, up to isomorphism.

□