Marco Paolieri

Analysis and verification of regenerative stochastic systems

Ph.D. Thesis

Università di Firenze

UNIVERSITÀ
DEGLI STUDI
FIRENZE

Dottorato di Ricerca in
Ingegneria Informatica, Multimediale
e delle Telecomunicazioni

Ciclo XXVII
Coordinatore Prof. Luigi Chisci

Analysis and Verification of Regenerative Stochastic Systems

Settore scientifico disciplinare ING-INF/05
(Sistemi di elaborazione delle informazioni)

Dottorando
Dott. Marco Paolieri

....

Tutore
Prof. Enrico Vicario

....

Coordinatore
Prof. Luigi Chisci

....

Anni 2012/2014

Contents

Chapter 1
Introduction

Predicting the operational properties of a system is a fundamental task of engineering sciences. System models allow to highlight the critical mechanisms for the satisfaction of performance requirements, enabling an early assessment of design choices. Nonetheless, the modeling of engineering and information systems, such as computer systems, telecommunication networks, critical infrastructures or manufacturing systems, is often complex due to the intertwined effects of concurrency and probability. Distributed processing tasks, for example, are spawned on several nodes in a cluster and proceed concurrently; during their execution, tasks can start new activities and wait for their completion in order to produce the final result of the computation. In addition to concurrency, systems often include randomness. Nodes of a cluster, hardware components or gas distribution pipes can all fail unexpectedly. Similarly, the amount of work required from the system is rarely known in advance: requests to a web server (and response lengths), commits to a database, or phone calls in a telecommunication network are naturally modeled as stochastic phenomena governed by probability laws.

On the one hand, probability allows one to abstract over unnecessary real-world details and summarize them quantitatively. Although apparently random, the failure of a hardware component or that of a network switch are governed by physical laws; analogously, a call on the telephone network is the result of complex interactions among users. In most circumstances, modeling these phenomena in great detail would not help with performance analysis, but lead the modeler astray from a clear understanding of the system.

On the other hand, probability can also be introduced purposefully in the design of the system. Randomization is essential, for example, to break symmetry in distributed systems: when a collision has been detected on a shared communication medium, how should the transmitters react? A simple solution is to let each transmitter wait for a random time: if the waiting times sampled by the transmitters are sufficiently far apart, a new collision will occur with low probability.

To further complicate matters, most real-world systems exhibit not only concurrency and probability, but also "memory". As time advances, the timers associated with concurrent events change their distributions, and the logic state of the system does not carry enough information to predict its future evolution. This phenomenon is intrinsic to aging processes: given a working component, for example, the probability of a failure in the next time unit initially decreases over time due to "infant mortality", and then increases due to progressive degradation. Another example is that of timeouts or watchdog timers that trigger a recovery operation after some deterministic time: the distribution of their remaining time is different after each event, as it reflects the random time spent in previous states. Moreover, the correct operation of real-time systems can depend crucially on periodic task releases and activities with bounded worst-case execution times in order to guarantee deadlines, mutual exclusion, or other concurrency requirements. Stochastic models of these mechanisms need generally distributed timers accumulating memory over time.

This thesis ventures into the analysis of non-Markovian stochastic systems. We focus on *regenerative* systems, which "lose memory" and probabilistically restart after selected discrete events. The corresponding time instants, called *regeneration points*, decompose the execution paths of the process into independent "epochs" with distributions determined only by their initial state (and not by the process evolution in previous epochs).

The work presents a solution for the transient analysis of systems in which multiple generally distributed timers can be started or stopped independently, but regenerations are encountered in a bounded number of discrete events. Regenerations are detected in the state space and epochs are analyzed with the method of *stochastic state classes*, which compute the joint probability density function of timers after each discrete event. Notably, the approach extends the class of models amenable to state-of-the-art analytical or numerical techniques.

Building on these results, we investigate the problem of verifying an *interval until operator* in a regenerative system. The aim is to check whether, with sufficient probability, the system satisfies a goal condition at some time in a given interval $[\alpha, \beta]$ without ever hitting any "forbidden" state. This problem has been widely studied for memoryless systems, in which the evolution before and after α (the beginning of the time window $[\alpha, \beta]$) can be analyzed independently using transient analysis. Such approach cannot be generalized to regenerative systems due to the memory accumulated by the process at time α.

A naïve solution could add a deterministic timer in parallel to the model so as to represent the elapse of α and register the corresponding event in the logic state. Unfortunately, this solution crucially affects regenerative transient analysis, since it is now the deterministic timer that carries memory, forgoing all regenerations points of the model before α.

We present a solution based on a renewal argument specific to the interval until operator, which results in bivariate integral equations (instead of the univariate Markov renewal equations of transient analysis). By establishing the theoretical relationship between stochastic state classes and cylinder sets of sample paths, an algorithm is formulated based on the enumeration of stochastic state classes limited to the first regenerative epoch. The solution fully leverages the repetitive structure of the underlying stochastic process, both before and after α, and results advantageous with respect to other analytical approaches.

1.1 Organization

The thesis is organized as follows.

In Chapter 2, stochastic time Petri nets are presented. We provide a probabilistic semantics of the model and discuss the class of its underlying stochastic process, recalling the main concepts of Markov renewal theory.

Chapter 3 provides an introduction to the method of stochastic state classes, presenting the properties of the computation of successor classes and highlighting the most relevant measures that can be derived about model executions.

In Chapter 4, we introduce the concept of regeneration, provide an algorithm to detect regeneration points over sequences of discrete events, and leverage measures computed from stochastic state classes to evaluate the local and global kernels of a Markov regenerative process. In turn, the local and global kernels are the basis for transient analysis with Markov renewal equations.

Chapter 5 presents the probabilistic model checking problem for Boolean combinations of interval until operators. In order to reason about measures of execution paths of the model, a probability space of paths is defined for stochastic time Petri nets. Through a renewal argument on the time of the next regeneration and on the time remaining before the lower bound α, we formulate a set of bivariate renewal equations specific to the interval until operator. The numerical solution in the time domain is discussed, and the computation of the required parameters is proved possible with stochastic state classes. A case study is also presented, analyzing a probabilistic variant of Fischer's mutual exclusion protocol.

Finally, Chapter 6 draws the conclusions of this work.

Chapter 2
Stochastic Time Petri Nets

Stochastic time Petri nets (STPNs) are a powerful and convenient high-level formalism for the modeling of systems with concurrency, probability and real-time constraints.

Multiple activities with generally distributed duration can be enabled by adding "tokens" to "input places" that represent preconditions; in turn, the fastest activity to complete can start or stop other activities by removing tokens from its input places and adding tokens to its "output places". This mechanism can represent precedence, synchronization, and parallel execution of activities. In addition, deterministic durations or probability density functions with bounded supports can produce an ordering between events that is often essential for the correct operation of real-time systems.

In this chapter, we introduce the formalism and provide a probabilistic semantics in terms of the stochastic process which records, after each discrete event, the new *marking* (number of tokens in each place) and the remaining time of enabled activities. From this process, we construct the marking process of the net and analyze its properties, motivating the discussion with examples of models and relevant measures.

2.1 Definition

Stochastic time Petri nets introduce random durations and probabilistic choices in Petri nets, with particular emphasis on minimum and maximum activity durations. Vertical bars called *transitions* represent activities, while the logic state of the net is represented by *tokens* contained in *places.* A directed arc from a place to a transition represents a *precondition*: the transition is enabled only if the place (called an *input place*) contains at least one token. In contrast, an arc terminating in an empty circle (called *inhibitor arc*) achieves the opposite effect: the transition is enabled only if the connected place (called *inhibitor place*) is empty.

Immediately after becoming enabled, each transition samples a *time to fire* (also called *timer value* or *clock reading*) according to a given probability density function (PDF). When multiple transitions are enabled, the one with minimum time to fire will fire first. The token count of each place, which we call *marking*, is updated after a firing according to the usual "token game" of Petri nets: one token is removed from each input place of the fired transition, and one token is added to each of the *output places* connected with directed arcs starting from the transition.

As a result, new transitions can be enabled (adding tokens to their input places, or removing tokens from their inhibitor places) or disabled (removing tokens from their input places, or adding tokens to their inhibitor places). Transitions that are enabled before the firing, after the tokens removal, and after the tokens addition are called *persistent*: their times to fire are not resampled, but instead decreased by the time to fire of the transition that just fired (which corresponds to the sojourn time in the previous logic state).

Let us formalize the concept and provide some examples.

Definition 2.1 (Stochastic time Petri Net). A stochastic time Petri net is a tuple $\langle P, T, A^-, A^+, A^\circ, EFT, LFT, f, w\rangle$ where:

- P is a finite set of places;
- T is a finite set of transitions, disjoint from P;
- $A^- \subseteq P \times T$ is the precondition relation;
- $A^+ \subseteq T \times P$ is the postcondition relation;
- $A^\circ \subseteq P \times T$ is the inhibitor relation;
- $EFT\colon T \to \mathbb{Q}_{\geqslant 0}$ and $LFT\colon T \to \mathbb{Q}_{\geqslant 0} \cup \{+\infty\}$ associate each transition $t \in T$ with an earliest firing time $EFT(t)$ and a latest firing time $LFT(t) \geq EFT(t)$;
- $f\colon T \to (\mathbb{R}_{\geqslant 0} \to [0,1])$ associates each transition $t \in T$ with a probability density function f_t with support $[EFT(t), LFT(t)]$;
- $w\colon T \to \mathbb{R}_{>0}$ associates each transition with a positive weight.

Given an STPN $\langle P, T, A^-, A^+, A^\circ, EFT, LFT, f, w\rangle$, a marking $m \in \mathbb{N}^P$ assigns a nonnegative number of tokens to each place of the net and identifies a set of *enabled transitions* $E(m)$ according to the usual rules of Petri nets: a transition t is enabled by m if m assigns at least one token to each of its input places and no tokens to its inhibitor places. Formally,

$$E(m) = \big\{t \in T \mid \forall (p,t) \in A^-, m(p) \geq 1 \text{ and } \forall (p,t) \in A^\circ, m(p) = 0\big\}\,.$$

The state of the net includes the marking and the time remaining before the firing of each enabled transition.

Definition 2.2 (State). The state of an STPN is a pair $\langle m, \vec{\tau}\rangle$ where $m \in \mathbb{N}^P$ is a marking and $\vec{\tau} \in \mathbb{R}_{\geqslant 0}^{E(m)}$ is a *times to fire vector* assigning a remaining time to each enabled transition.

Given a state $s = \langle m, \vec{\tau} \rangle$, the next transition is selected from the set

$$E_{\min}(\langle m, \vec{\tau} \rangle) = \arg \min_{t \in E(m)} \vec{\tau}(t)$$

of enabled transitions with minimum time to fire. The selection is performed randomly according to the discrete distribution given by transition weights: in particular, each transition $t \in E_{\min}(s)$ is selected with probability

$$\frac{w(t)}{\sum_{u \in E_{\min}(s)} w(u)} \,.$$

The state $s' = \langle m', \vec{\tau}' \rangle$ reached after the firing of t is computed according to the following rule.

Definition 2.3 (State update rule). Given a state $s = \langle m, \vec{\tau} \rangle$ for the STPN $\langle P, T, A^-, A^+, A^\circ, EFT, LFT, f, w \rangle$ and an enabled transition $t \in E_{\min}(s)$ with minimum time to fire in s, the successor state of s through t is $s' = \langle m', \vec{\tau}' \rangle$ where:

- The new marking m' is derived from m by removing one token from each input place of t, resulting in the marking

$$m_{tmp}(p) = \begin{cases} m(p) - 1 & \text{if } (p, t) \in A^-, \\ m(p) & \text{otherwise,} \end{cases}$$

  and adding one token to each output place of t, resulting in

$$m'(p) = \begin{cases} m_{tmp}(p) + 1 & \text{if } (t, p) \in A^+, \\ m_{tmp}(p) & \text{otherwise.} \end{cases}$$

- The new time to fire $\vec{\tau}'(u)$ of each transition $u \in E(m')$ enabled by m' and also by m and m_{tmp} (which we call *persistent* to the firing) is equal to $\vec{\tau}'(u) = \vec{\tau}(u) - \vec{\tau}(t)$. The time to fire of each persistent transition is thus reduced by the sojourn time of the STPN in the previous marking, which corresponds to the minimum time to fire $\vec{\tau}(t)$.
- Other transitions enabled by m' are called *newly enabled*, and the time to fire $\vec{\tau}'(u)$ of each is sampled independently in $[EFT(u), LFT(u)]$ according to the probability density function f_u associated with u in the STPN definition.

Without loss of generality, we suppose that the initial marking m_0 of the net is given, while the initial times to fire $\vec{\tau}_0$ of the transitions $E(m_0) = \{t_1, t_2, \dots, t_n\}$ enabled by m_0 are sampled according to some initial PDF $f_{\vec{\tau}_0}$ over $\mathbb{R}^n_{\geqslant 0}$. It is common to assume that all transitions are initially newly enabled, and thus

$$f_{\vec{\tau}_0}(x_1,\ldots,x_n) = \prod_{i=1}^{n} f_{t_i}(x_i).$$

More generally, we are interested in the case that each transition $t_i \in E(m_0)$ has been enabled for a deterministic time d_i. Conditioned to this hypothesis, the reduced times to fire are then independently distributed according to the PDF

$$f_{\vec{\tau}_0}(x_1,\ldots,x_n) = \prod_{i=1}^{n} \frac{f_{t_i}(x_i + d_i)}{\int_{\max\{d_i, EFT(t_i)\}}^{LFT(t_i)} f_{t_i}(u)\, du}$$

on the support

$$D_{\vec{\tau}_0} = \prod_{i=1}^{n} \left[\max\{0, EFT(t_i) - d_i\}, LFT(t_i) - d_i\right].$$

Following the usual terminology of stochastic Petri nets, a transition t is called *immediate* (IMM) if $EFT(t) = LFT(t) = 0$ and *timed* otherwise. A timed transition is called *exponential* (EXP) if $EFT(t) = 0$, $LFT(t) = +\infty$ and $f_t(x) = \lambda\, e^{-\lambda x}$ for some rate $\lambda \in \mathbb{R}_{>0}$. Transitions with times to fire distributed according to non-exponential distributions are called *general* (GEN); as a special case, a general transition t is *deterministic* (DET) if $EFT(t) = LFT(t) > 0$. For an immediate or deterministic transition t, the probability mass is concentrated on the value $\overline{x} = EFT(t) = LFT(t)$. With an abuse of notation, we denote its probability density function by $f_t(x) = \delta(x - \overline{x})$ and, for any function g, we write

$$\int g(x)\, \delta(x - \overline{x})\, dx$$

for the Lebesgue-Stieltjes integral

$$\int g(x)\, d\mathbb{1}_{[\overline{x},\infty)}(x)$$

where $\mathbb{1}_A$ represents the indicator function of the set A:

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

Example 2.1 (Preemptive Single-Server Queue). Consider the model of a queue serving a population of two customers. Each customer enters the queue after some activity, gets served, and then starts another activity cycle. The queue has only one server which enforces a *preemptive repeat different* policy among the two customers: if the first customer arrives while the service of the second one is in progress, the server discards the work completed so far

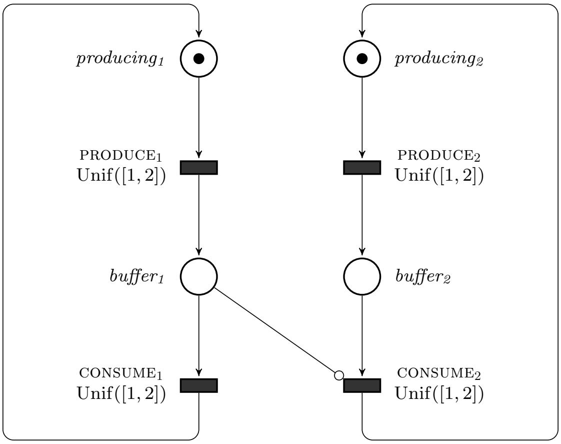


Fig. 2.1: STPN model of a preemptive single-server queue.

for the second customer and switches to serving the first customer. This model represents, for example, a producer–consumer system in which two producers create items for a single consumer with priority policy. The system has limited buffer capacity: after creating an item, the producers must wait for its consumption before starting to create the next one.

This system can be specified with the stochastic time Petri net of Fig. 2.1. The transitions PRODUCE$_1$ and PRODUCE$_2$ represent the production activities, while CONSUME$_1$ and CONSUME$_2$ correspond to the consumption of items by the server. For simplicity, the duration of all activities is assumed to be uniform on the bounded support $[1, 2]$, although we could associate distinct probability density functions with the production and consumption of items of the two producers.

The inhibitor arc from *buffer₁* to CONSUME$_2$ enforces priority among the two customers. When PRODUCE$_1$ fires, it removes a token from *producing₁* and adds a token to *buffer₁*, enabling CONSUME$_1$: if CONSUME$_2$ was enabled by a token in *buffer₂*, it gets immediately disabled. After the completion of PRODUCE$_1$, a token will be removed from *buffer₁* and added to *producing₁*, thus enabling PRODUCE$_1$. The activity CONSUME$_2$ will then start again by sampling a new time to fire.

The logic state of the net is represented by its marking, which gives a token count for each place. In total, the model can reach four markings of the form $(\mathit{producing}_1, \mathit{buffer}_1, \mathit{producing}_2, \mathit{buffer}_2)$:

1. $(1, 0, 1, 0)$ is the initial state of the net, in which both producers are creating an item;

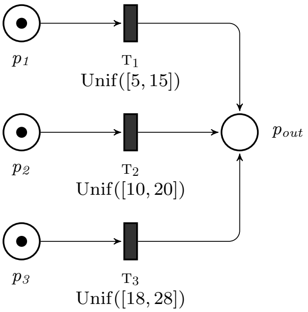


Fig. 2.2: STPN model of three concurrent activities with bounded duration.

2. $(0, 1, 1, 0)$ is the state in which the first producer has created an item and the second producer is still completing its production;
3. $(1, 0, 0, 1)$ represents the inverse situation in which the second producer has created an item and the first one is still completing the production;
4. $(0, 1, 0, 1)$ is the state in which both producers have created an item, and only the first item is being consumed due to the priority policy.

It is common to denote markings by indicating the name of nonempty places preceded by the number of contained tokens. For example, the four markings of the net can be denoted as $1\mathit{producing_1}\,1\mathit{producing_2}$, $1\mathit{buffer_1}\,1\mathit{producing_2}$, $1\mathit{producing_1}\,1\mathit{buffer_2}$, and $1\mathit{buffer_1}\,1\mathit{buffer_2}$, or just as $\mathit{producing_1}\,\mathit{producing_2}$, $\mathit{buffer_1}\,\mathit{producing_2}$, $\mathit{producing_1}\,\mathit{buffer_2}$, and $\mathit{buffer_1}\,\mathit{buffer_2}$ by omitting unitary token counts.

Example 2.2 (PDF supports enforcing precedence). In the definition of stochastic time Petri nets, the support of the probability density function associated with each transition t is made explicit by the earliest firing time $EFT(t)$ and latest firing time $LFT(t)$. By treating time bounds as “first-class citizens”, this approach enables the verification of important real-time properties of STPNs through symbolic state-space analysis in the style of Berthomieu and Diaz (1991) and Vicario (2001).

Fig. 2.2 reports a simple example of three concurrent activities T_1, T_2, and T_3 with durations uniformly distributed on the supports $[5, 15]$, $[10, 20]$ and $[18, 28]$, respectively. Regardless of the probability density functions, the supports impose a strict precedence relation among feasible events: the transition T_1 will *surely* fire before T_3, since its latest firing time $LFT(T_1) = 15$ is strictly lower than the earliest firing time $EFT(T_3) = 18$ of T_3.

It is also important to note that times to fire associated with enabled transitions become *dependent* random variables when they persist to a firing. Let X_1, X_2, X_3 represent the times to fire of enabled transitions: initially, these random variables are independently distributed, with $X_1 \sim \mathrm{Unif}([5,15])$, $X_2 \sim \mathrm{Unif}([10,20])$, and $X_3 \sim \mathrm{Unif}([18,28])$. As a consequence, their joint probability density function

$$f(x_1,x_2,x_3) = \prod_{i=1}^{3} f_{\mathrm{T}_i}(x_i) = \frac{1}{15-5}\cdot\frac{1}{20-10}\cdot\frac{1}{28-18}$$

is given by the product of individual PDFs, and it is nonzero over the support $[5,15]\times[10,20]\times[18,28] \subseteq \mathbb{R}^3_{\geqslant 0}$ that results from the Cartesian product of individual PDF supports. Conditioned to the firing of T_2 and decreased by the elapsed time, the times to fire of T_1 and T_3 are a bivariate random variable

$$(X'_1, X'_3) = (X_1 - X_2, X_3 - X_2 \mid X_2 \leq X_1 \wedge X_2 \leq X_3)$$

with joint probability density function

$$f'(x'_1,x'_3) = \begin{cases} -\frac{8}{1000}x'_1 + \frac{4}{100} & \text{if } (x'_1,x'_3) \in Z_\alpha, \\ -\frac{8}{1000}x'_1 + \frac{8}{1000}x'_3 - \frac{24}{1000} & \text{if } (x'_1,x'_3) \in Z_\beta, \\ -\frac{8}{1000}x'_3 + \frac{144}{1000} & \text{if } (x'_1,x'_3) \in Z_\gamma, \end{cases}$$

over the partitioned support $Z_\alpha \cup Z_\beta \cup Z_\gamma$ with

$$\begin{aligned} Z_\alpha &= \left\{(x'_1,x'_3) \in \mathbb{R}^2_{\geqslant 0} \mid 0 \leq x'_1 \leq 5 \wedge 8 \leq x'_3 \leq 18 \wedge 3 \leq x'_3 - x'_1 \leq 13\right\}, \\ Z_\beta &= \left\{(x'_1,x'_3) \in \mathbb{R}^2_{\geqslant 0} \mid 0 \leq x'_1 \leq 5 \wedge 3 \leq x'_3 \leq 8 \wedge 3 \leq x'_3 - x'_1 \leq 8\right\}, \text{ and} \\ Z_\gamma &= \left\{(x'_1,x'_3) \in \mathbb{R}^2_{\geqslant 0} \mid 0 \leq x'_1 \leq 5 \wedge 13 \leq x'_3 \leq 18 \wedge 13 \leq x'_3 - x'_1 \leq 18\right\}. \end{aligned}$$

Neither the PDF nor its piecewise support are in product-form: as expected, X'_1 and X'_3 are dependent random variables. Stochastic state classes, presented in Chapter 3, allow to compute joint probability density functions of enabled timers after each discrete event. This calculus crucially relies on the manipulation of both analytical PDF expressions and supports.

Example 2.3 (Randomization using weights). When the minimum time to fire in the current state is associated with multiple transitions, one is selected randomly according to the discrete distribution given by their weights. This mechanism allows to introduce randomized choices, or to abstract external phenomena affecting the result of an action.

Consider, for example, a sender transmitting a packet over a lossy channel: with probability q, the packet is dropped, requiring a retransmission. In case of failure, the sender tries to retransmit the packet until the elapse of a timeout.

Fig. 2.3 presents an STPN model of this system. Each transmission activity is modeled by the transition send with duration uniformly distributed over $[1,2]$. After its completion, the token is removed from place *ready* and

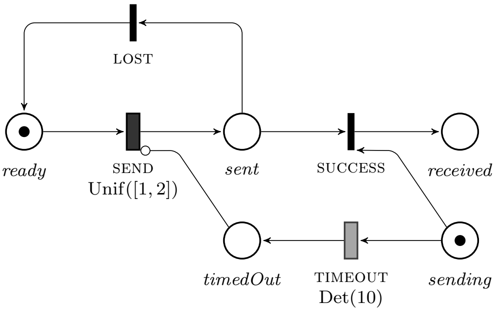


Fig. 2.3: STPN model of a packet transmission over a lossy channel.

added to place *sent*: as a result, the immediate transitions LOST and SUCCESS are enabled. The times to fire of these transitions are deterministic and equal to zero: by assigning $w(\text{LOST}) = q$ and $w(\text{SUCCESS}) = 1 - q$, we obtain that, with probability

$$\frac{w(\text{LOST})}{w(\text{LOST}) + w(\text{SUCCESS})} = \frac{q}{q + (1-q)} = q,$$

LOST is selected, moving the token from *sent* back to *ready*; conversely, with probability

$$\frac{w(\text{SUCCESS})}{w(\text{LOST}) + w(\text{SUCCESS})} = \frac{1-q}{q + (1-q)} = 1 - q,$$

SUCCESS is selected, moving the token from place *sent* to place *received*.

Upon a LOST event, a new transmission is started. After 10 time units, the TIMEOUT transition is fired: by adding a token to the place *timedOut*, transition SEND is inhibited and the transmission procedure halts. In contrast, a SUCCESS event removes the token from place *sending*, thus disabling the timeout.

It is important to stress the fact that weights come into play only for the selection among immediate or deterministic transitions. The probability that two transitions sample an identical time to fire value is in fact nonzero only if they both concentrate some probability mass on the value.

EXAMPLE 2.4 (Enabling and update functions). It is often convenient to extend Petri net models with more general mechanisms for the identification of enabled transitions and for the update of the marking after a firing. In the style of *stochastic reward nets* by Ciardo et al. (1993) and *stochastic activity networks* by Sanders and Meyer (2001), STPNs can be extended with *enabling functions* and *update functions*.

Enabling functions define additional requirements on the token counts of a marking for the enabling of each transition. With enabling functions, a transition t is enabled by a marking m if

1. m assigns at least one token to each input place of t,
2. m assigns no tokens to the inhibitor places of t, and
3. the enabling function E_t associated with t evaluates to TRUE when applied to m, i.e., $E_t(m) = \text{TRUE}$.

Formally, the set of transitions enabled by m becomes

$$E(m) = \big\{t \in T \mid \forall (p,t) \in A^-, m(p) \geq 1 \text{ and } \\ \forall (p,t) \in A^\circ, m(p) = 0 \text{ and } E_t(m) = \text{TRUE}\big\}$$

where each enabling function $E_t \colon \mathbb{N}^P \to \{\text{TRUE}, \text{FALSE}\}$ is an arbitrary test on markings. Graphically, enabling functions are annotated next to transitions after the symbol "?".

Update functions define additional token moves to be performed after the firing of a transition. When the update function $U_t \colon \mathbb{N}^P \to \mathbb{N}^P$ is associated with transition $t \in T$, the new marking m'' after the firing of t is obtained by removing one token from each input place of t, resulting in the marking

$$m_{tmp}(p) = \begin{cases} m(p) - 1 & \text{if } (p,t) \in A^-, \\ m(p) & \text{otherwise,} \end{cases}$$

adding one token to each output place of t, resulting in

$$m'(p) = \begin{cases} m_{tmp}(p) + 1 & \text{if } (t,p) \in A^+, \\ m_{tmp}(p) & \text{otherwise,} \end{cases}$$

and then applying the function U_t to m', which gives $m'' = U_t(m')$. Transitions enabled by m'' are persistent if they are also enabled by m', m_{tmp} and m. Graphically, update functions are annotated next to transitions as token assignments denoted by the symbol "←".

As an example, consider a system in which the items created by two producers are consumed using a shared resource. The access to the resource is mutually exclusive and buffers have unitary capacity: each producer must wait for the consumption of the created item before starting a new production. In addition, the resource requires a "setup operation" every time it switches to the consumption of items created by a different producer.

A model of this system is presented in Fig. 2.4. The place *lastUsed* is empty when the shared resource is in use; when the resource is available, the token count of *lastUsed* records the identifier (1 or 2) of the last process that used the resource. After the production of an item (transitions PRODUCE$_1$ and PRODUCE$_2$), each process tries to acquire the resource for the

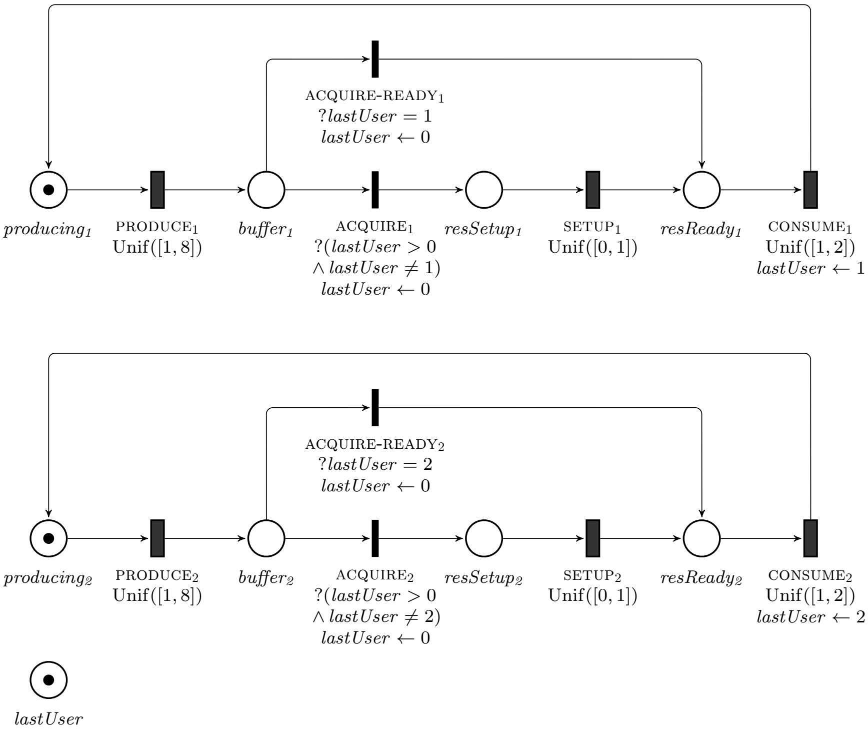


Fig. 2.4: STPN model of two processes sharing a resource with setup times.

item consumption. The resource is acquired by process i for $i = 1, 2$ through transition ACQUIRE-READY$_i$ if the enabling function $?lastUsed = i$ is satisfied: in this case, *lastUsed* is set to zero by the update function $lastUsed \leftarrow 0$ and the activity CONSUME$_i$ can start immediately.

In contrast, if the previous use of the resource was performed by another process, process i acquires the resource through transition ACQUIRE$_i$, which is associated with the enabling function $?(lastUser > 0 \wedge lastUser \neq i)$. Transition ACQUIRE$_i$ adds one token to place $resSetup_i$, enabling the transition SETUP$_i$: after its completion, the resource is ready and the item consumption can begin.

After the item consumption, the resource is released by setting the token count of *lastUser* to the identifier of the process that just used it. This update of the marking is obtained through the update functions $lastUser \leftarrow i$ associated with transitions CONSUME$_i$ for $i = 1, 2$.

Enabling functions and update functions provide flexible constructs for the definition of a model. In the example, additional producers can be added simply by replicating the part of the STPN which represents producer 1 or

2, and by modifying its identifier in enabling and update functions. Nonetheless, enabling and update functions do not extend the modeling power of the STPN formalism, nor affect the solution techniques presented in the rest of this work.

2.2 Probabilistic semantics

In this section, we provide a probabilistic semantics of stochastic Petri nets in terms of the discrete-time process $\{(m_n, \vec{\tau}_n),\ n \in \mathbb{N}\}$ in which the random variable $(m_0, \vec{\tau}_0)$ corresponds to the initial state of the net, and $(m_n, \vec{\tau}_n)$ for $n > 0$ corresponds to the state reached after the nth transition firing. This process is intuitively a Markov process: according to the definition of STPNs given in the previous section, the future evolution of the system depends on the past history only through the current state. However, the state space of the process is uncountably infinite due to the continuous set of values for the times to fire $\vec{\tau}_n$ of transitions enabled by m_n. We thus introduce *general state-space Markov chains* (GSSMCs).

Definition 2.4 (General State-Space Markov Chain). The process $\{\, Z_n,\ n \in \mathbb{N}\,\}$ defined on the probability space $(\Omega, \mathcal{F}, P_\mu)$ and taking values in Γ is a general state-space Markov chain with initial distribution μ and transition kernel P if, for all $A \subseteq \Gamma$, $P_\mu\{Z_0 \in A\} = \mu(A)$ and $P_\mu\{Z_{n+1} \in A \mid Z_0, Z_1, \ldots, Z_n\} = P(Z_n, A)$ almost surely.

Intuitively, the initial distribution μ assigns a probability measure to the initial state Z_0, while the transition kernel $P(Z_n, A)$ gives the probability that, given the current state Z_n, the next state Z_{n+1} belongs to the set A. A general state-space Markov chain is fully defined by its initial distribution and by its transition kernel. In fact, given μ and P, a probability space $(\Omega, \mathcal{F}, P_\mu)$ can be constructed for $\{\, Z_n,\ n \in \mathbb{N}\,\}$ by defining the sample space Ω as the set of sequences $(z_0, z_1, \ldots)$ of states $z_i \in \Gamma$. Then, a probability measure P_μ on the σ-algebra $\mathcal{F}$ of cylinder sets of the form

$$\{\omega \in \Omega \mid z_{i_0} \in A_{i_0}, z_{i_1} \in A_{i_1}, \ldots, z_{i_n} \in A_{i_n}\}$$

for all $n > 0$, indices $i_0, i_1, \ldots, i_n$ and $A_{i_k} \subseteq \Gamma$ for $k = 0, \ldots, n$, can be defined from the finite-dimensional distributions

$$P(\hat{A}_0, \ldots, \hat{A}_m) = \int_{\hat{A}_0} \mu(dz_0) \int_{\hat{A}_1} P(z_0, dz_1) \cdots \int_{\hat{A}_m} P(z_{m-1}, dz_m)$$

by letting $m = \max(i_0, i_1, \ldots, i_n)$ and

$$\hat{A}_j = \begin{cases} A_{i_k} & \text{if } j = i_k \text{ for some } k, \\ \Gamma & \text{otherwise,} \end{cases}$$

and leveraging Kolmogorov's existence theorem, as detailed in Haas (2002). It is thus sufficient to define a general state-space Γ, an initial distribution μ and a transition kernel P in order to provide a probabilistic semantics for stochastic time Petri nets.

The general state-space of the Markov chain $\{(m_n, \vec{\tau}_n),\ n \in \mathbb{N}\}$ can be defined as

$$\Gamma = \bigcup_{m \in \mathcal{M}} \{m\} \times \mathbb{R}_{\geqslant 0}^{|E(m)|}$$

where $\mathcal{M}$ is the set of reachable markings of the net (countable by hypothesis). Each state $\langle m, \vec{\tau} \rangle \in \Gamma$ includes a marking m and an assignment of nonnegative times to fire for the transitions $E(m)$ enabled by m. If we consider sets of states with the simple form

$$A = \{m'\} \times [0, a_1] \times [0, a_2] \times \cdots \times [0, a_n]$$

with $m' \in \mathcal{M}$, $E(m') = \{t_1, t_2, \ldots, t_n\}$ and $a_1, a_2, \ldots, a_n \geq 0$, then the initial distribution μ can be defined as

$$\mu(A) = \begin{cases} \int_{[0,a_1] \times \cdots \times [0,a_n]} f_{\vec{\tau}_0} & \text{if } m' = m_0, \\ 0 & \text{otherwise,} \end{cases}$$

where m_0 is the initial marking and $f_{\vec{\tau}_0}$ is the initial times to fire PDF of the STPN. The transition kernel P can be defined as

$$P(\langle m, \vec{\tau} \rangle, A) = \sum_{t \in E_{\min}(\langle m, \vec{\tau} \rangle)} \frac{w(t)}{\sum_{u \in E_{\min}(\langle m, \vec{\tau} \rangle)} w(u)} P_t(\langle m, \vec{\tau} \rangle, A)$$

where

$$E_{\min}(\langle m, \vec{\tau} \rangle) = \arg \min_{t \in E(m)} \vec{\tau}(t)$$

is the set of enabled transitions with minimum time to fire in state $\langle m, \vec{\tau} \rangle$,

$$\frac{w(t)}{\sum_{u \in E_{\min}(\langle m, \vec{\tau} \rangle)} w(u)}$$

is the probability that transition t is selected from $E_{\min}(\langle m, \vec{\tau} \rangle)$, and

$$P_t(\langle m, \vec{\tau} \rangle, A) = \begin{cases} \prod_{i \in N_{m,t}} \int_{[0,a_i]} f_{t_i} \prod_{i \in O_{m,t}} \mathbb{1}_{[0,a_i]} (\vec{\tau}(t_i) - \vec{\tau}(t)) & \text{if } m \xrightarrow{t} m', \\ 0 & \text{otherwise,} \end{cases}$$

gives the probability that, after the firing of t in $\langle m, \vec{\tau} \rangle$, the next state $\langle m', \vec{\tau}' \rangle$ belongs to A. This probability is nonzero only if m' is the marking resulting from the firing of t in m (which we write as $m \xrightarrow{t} m'$) and the decreased time to fire of each persistent transitions t_i belongs to $[0, a_i]$: in particular, the

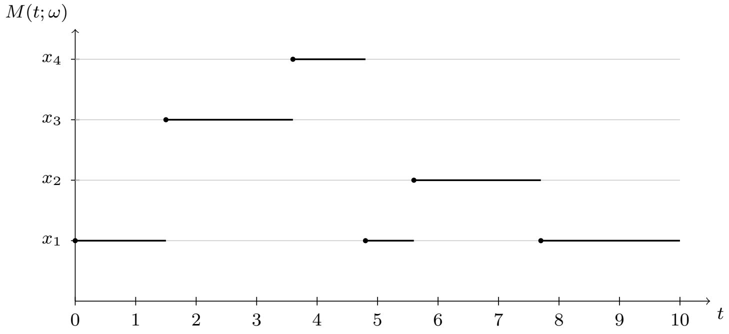


Fig. 2.5: Sample path of the marking process $\{M(t),\ t \geq 0\}$.

set $O_{m,t}$ represents the indices of persistent transitions in $E(m')$ and $\mathbb{1}_{[0,a_i]}$ is the indicator function of the set $[0, a_i]$. Each newly enabled transition t_i is sampled independently, and thus it belongs to $[0, a_i]$ with probability $\int_{[0,a_i]} f_{t_i}$; representing by $N_{m,t}$ the indices of newly enabled transitions in $E(m')$, we obtain the term

$$\prod_{i \in N_{m,t}} \int_{[0,a_i]} f_{t_i} \,.$$

Note that, without loss of generality, we assumed the sampling PDFs f_{t_i} to be zero outside of the supports $[EFT(t_i), LFT(t_i)]$. We also stress the fact that disabled transitions do not influence transition probabilities given the current state $\langle m, \vec{\tau} \rangle$, although they played a role in the past history of the Markov chain by setting an upper bound for the times to fire of transitions that could fire in each state.

2.3 The marking process

The *marking process* $\{M(t),\ t \geq 0\}$ describes the logic state of an STPN at each time instant, and it is thus crucial in the computation of many performance measures. This process is defined over a countable state space (the reachable markings of the net) and evolves over continuous time with piecewise-constant sample paths, as illustrated in Fig. 2.5.

For each $t \geq 0$, the random variable $M(t)$ can be defined as the last marking reached within t by the general state-space Markov chain $\{(m_n, \vec{\tau}_n),\ n \in \mathbb{N}\}$ constructed in the previous section. We denote by Δ_n the duration of the nth sojourn in a state, which is given by the minimum time to fire in the state $(m_{n-1}, \vec{\tau}_{n-1})$ of the GSSMC and thus, for all $n > 0$,

$$\Delta_n = \begin{cases} \min_{1 \leq i \leq k} \vec{\tau}_{n-1}(t_i) & \text{if } E(m_{n-1}) = \{t_1, t_2, \ldots, t_k\}, \\ \infty & \text{if } E(m_{n-1}) = \varnothing. \end{cases}$$

Then, we denote the number of state changes performed within time t by

$$N(t) = \sup\left\{ j \in \mathbb{N} : \sum_{n=1}^{j} \Delta_n \leq t \right\} \tag{2.1}$$

and define the marking process $\{M(t),\ t \geq 0\}$ as $M(t) = m_{N(t)}$ for all $t \geq 0$.

By construction, the marking process is piecewise-constant and it has right-continuous sample paths. This property is evident from Eq. (2.1), in which the jth firing time $\sum_{n=1}^{j} \Delta_n$ is required to be lower or *equal* to the current time t. The current marking $M(t) = m_{N(t)}$ thus refers to the last marking reached within time t that does not enable any immediate transition. We refer to such markings as *tangible* markings, and to markings enabling immediate transitions as *vanishing* markings.

Also note that infinitely many marking changes can occur in a finite time interval (*Zeno behavior*) if the process is absorbed into a set of vanishing markings or if the absolute firing times converge to a finite value. Both phenomena can be excluded w.p.1 in STPNs through the following results, which we report from Haas (2002).

Let $\mathcal{M}'$ denote the set of vanishing markings, and let $\{(m_n, \vec{\tau}_n),\ n \in \mathbb{N}\}$ be the general state-space Markov chain of the states reached by the STPN after each firing, which is defined on the probability space $(\Omega, \mathcal{F}, P_\mu)$ given by the transition kernel P and initial distribution μ.

Theorem 2.1 (Return to tangible markings). *When $\mathcal{M}'$ is finite, $P_\mu\{m_n \notin \mathcal{M}' \text{ i.o.}\} = 1$ for any initial distribution μ if and only if, for each vanishing marking $m \in \mathcal{M}'$, a tangible marking $m' \notin \mathcal{M}'$ can be reached with probability greater than zero through the firing of a sequence of immediate transitions.*

The conditions of Theorem 2.1 are trivially necessary: a tangible marking must be reachable with nonzero probability from each vanishing marking to avoid the absorption into $\mathcal{M}'$ for any initial distribution μ. The proof of sufficiency relies on the fact that sample paths confined to $\mathcal{M}'$ exclude infinitely often sequences of immediate firings that reach tangible markings. Since each of these sequences has nonzero probability, the measure of sample paths that remain in $\mathcal{M}'$ is almost surely zero.

In stochastic time Petri nets, visiting tangible markings infinitely often is a sufficient condition to guarantee that the lifetime

$$\sup_j \sum_{n=1}^{j} \Delta_n$$

of the marking process $M(t)$ is a.s. infinite, thus excluding Zeno behaviors.

Theorem 2.2 (Infinite lifetime of the marking process). *If the general state-space Markov chain satisfies $P_\mu\{m_n \notin \mathcal{M}' \text{ i.o.}\} = 1$, then $P_\mu\{\sup_j \sum_{n=1}^{j} \Delta_n = \infty\} = 1$.*

The proof relies on the fact that STPNs have unitary rates and fixed time to fire PDFs associated with transitions. Since tangible markings are reached infinitely often, at least one timed transition must be enabled and fired infinitely often, each time guaranteeing a time advancement sampled according to its PDF. This infinite sequence of independent and identically distributed times to fire is sufficient to guarantee that $\sup_j \sum_{n=1}^{j} \Delta_n = \infty$ almost surely. Finally, the case of an absorbing state in which no transition is enabled trivially guarantees an infinite lifetime (since $\Delta_n = \infty$ for some $n > 0$).

2.4 Transient analysis of the marking process

The analysis of the general state-space Markov chain underlying an STPN is difficult due to its infinite state-space and to the complex PDFs of remaining times to fire after a partial history of transition firings. Stochastic state classes, presented in the next chapter, provide a means to compute conditional PDFs of persistent timers, but they encounter an exponential complexity in transient analysis. In fact, the maximum number of successive firings that can happen within a limited time bound $t_{\max}$ with probability greater than some value $\epsilon > 0$ can grow linearly with $t_{\max}$. When multiple transitions can fire in each marking, the number of distinct sequences of transition firings before $t_{\max}$ can grow exponentially. Moreover, the worst-case memory and time complexity for the computation of joint times to fire PDFs after a sequence of transition firings grows exponentially with the length of the sequence, as detailed in Carnevali et al. (2009).

It is thus important to analyze the properties of the marking process $\{M(t),\ t \geq 0\}$ in order to devise efficient techniques for its transient analysis. First, we characterize its class in the most general setting by introducing *generalized semi-Markov processes* (GSMPs).

Definition 2.5 (Building blocks of a GSMP). A generalized semi-Markov process is defined by a tuple $\langle X, E, A, p, F, r\rangle$ in which

- X is a countable set of logic states;
- E is a finite set of events;
- $A\colon X \to 2^E$ assigns a subset $A(x)$ of enabled events to each state $x \in X$;
- $p\colon X \times 2^E \times X \to [0,1]$ gives the probability $p(x, \eta, x')$ that the simultaneous firing of the events $\eta \subseteq 2^E$ in $x \in X$ will result in $x' \in X$;

- $F\colon E \times X \times 2^E \times X \to [0,1]^{\mathbb{R}}$ gives the sampling distributions $F(e,x,\eta,x')$ of the timer of event $e \in E$ in state $x' \in X$ reached from $x \in X$ through the simultaneous firing of the events $\eta \subseteq 2^E$;
- $r\colon E \times X \to \mathbb{R}_{\geqslant 0}$ assigns a speed $r(e,x)$ to the timer of each event $e \in E$ in state $x \in X$.

Similarly to STPNs, the state of the system includes the logic state $x \in X$ and the clock readings $\vec{c} \in \mathbb{R}_{\geqslant 0}^{|E|}$ of enabled events. The initial state x_0 is sampled according to a discrete distribution ν_0 and the clock readings $\vec{c}_0$ of enabled events $e \in A(x_0)$ are sampled independently according to given initial distributions $F_0(\,\cdot\,, e, x_0)$. In each state $\langle x, \vec{c}\rangle$, the the next discrete-event happens after a sojourn of duration

$$t^* = \min_{e \in A(x)} \frac{\vec{c}(e)}{r(e,x)}$$

and it is triggered by the simultaneous firing of the events

$$\eta^* = \arg \min_{e \in A(x)} \frac{\vec{c}(e)}{r(e,x)}\,.$$

The next logic state $x' \in X$ is selected according to the discrete distribution $p(x, \eta^*, \,\cdot\,)$. The new clock reading of each persistent event $e \in A(x') \cap (A(x) \setminus \eta^*)$ is equal to

$$\vec{c}\,'(e) = \vec{c}(e) - t^*\, r(e,x),$$

while the clock reading of each new event $e \in A(x') \setminus (A(x) \setminus \eta^*)$ is sampled independently according to the distribution $F(e, x, \eta^*, x')$.

Similarly to STPNs, the general state-space Markov chain $\{(x_n, \vec{c}_n),\ n \in \mathbb{N}\}$ of the logic state and clock readings can be characterized by an initial distribution and a transition kernel defined according to the aforementioned mechanisms for the evolution of the state. We call *generalized semi-Markov process* the continuous-time stochastic process $\{X(t),\ t \geq 0\}$ which records, for each $t \geq 0$, the logic state of the GSSMC defined for some tuple $\langle X, E, A, p, F, r\rangle$.

We highlight the main differences in the definition of GSMPs and STPNs.

- The logic state of a GSMP is an element of an arbitrary countable set, and not a marking associating token counts to places. As a consequence, the definition of the set $A(x)$ of events enabled in each state is arbitrary and must be explicitly specified in the model definition. Nonetheless, a general enabling mechanism can be introduced in STPNs using *enabling functions*, as described in Example 2.4.
- Multiple events can occur simultaneously in GSMPs, while transitions of an STPN always fire in sequence. In particular, when multiple transitions have the same minimum time to fire, one is selected randomly according

to weights. It is important to stress the fact that this situation can arise only with immediate and deterministic transitions.

- After the occurrence of a set of events, the new state of a GSMP is selected randomly according to an arbitrary discrete distribution. In contrast to STPNs, in which the successor marking is obtained through token moves, there are no constraints on the allowed set of successor states. We note that a similar randomization mechanism can be emulated in STPNs with immediate transitions, and general marking updates can be introduced using *update functions*, as described in Example 2.4. Nonetheless, the rules to determine the set of persistent and newly enabled transitions in STPNs are still based on intermediate token counts.
- The clock readings of new events are sampled in GSMPs according to probability distributions $F(e, x, \eta^*, x')$ that can depend on the current state x, set of fired events η^* and next state x'. In contrast, STPNs associate each transition with a fixed PDF for the sampling of its time to fire. Although less general, this simpler mechanism eases the model definition and rules out Zeno behaviors (infinite firings occurring in a finite time) that can be produced by specific successions of PDFs for the sampling of the same transition after each newly enabling.
- The clocks associated with the events of a GSMP can decrease with different speeds $r(e, x)$ specific to the current state x. In contrast, in STPNs all the times to fire are always decreased with unitary rate. This hypothesis is essential for the computation of stochastic state classes, since unitary clock speeds result in PDF supports that can be represented as DBM zones, as described in Carnevali et al. (2009). As a consequence, stochastic time Petri nets are not able to model systems with *preemptive resume* policy, in which the execution of some activities can be suspended (setting their decreasing speeds to 0) and successively resumed, as for example described in Bobbio et al. (2000).

It is evident from this comparison that GSMPs provide mechanisms that are strictly more general than those of STPNs. Although clock setting distributions of GSMPs are required to be positive w.p.1, so that no immediate events are allowed in GSMPs, the following theorem can be derived directly from the results of Haas (2002).

Theorem 2.3. *For each STPN $\langle P, T, A^-, A^+, A^\circ, EFT, LFT, f, w\rangle$ with finite reachable markings $\mathcal{M}$ and marking process $\{M(t),\ t \geq 0\}$, there exists a GSMP $\{X(t),\ t \geq 0\}$, given by some tuple $\langle X, E, A, p, F, r\rangle$, that has the same finite dimensional distributions of $\{M(t),\ t \geq 0\}$ under an appropriate mapping between the state spaces $\mathcal{M}$ and X.*

The marking process $\{M(t),\ t \geq 0\}$ of an STPN thus belongs to a strict subclass of GSMPs. These processes can accumulate memory indefinitely: for all $\tilde{t} > 0$, the future evolution $\{M(t),\ t > \tilde{t}\}$ can depend not only on the current state $M(\tilde{t})$, but also on the previous history $\{M(t),\ t < \tilde{t}\}$.

In this case, simulation is regarded as the only tractable approach to the analysis of the marking process, as detailed in Haas and Shedler (1986), Haas and Shedler (1989), and Haas (2002). In the following subsections, we present two important subclasses of GSMPs that allow a numerical solution for transient probabilities of STPNs.

2.4.1 Markovian marking process

When the time to fire of each transition is sampled according to a fixed exponential distribution, the marking process of an STPN is a *continuous-time Markov chain* (CTMC). This class of stochastic processes does not accumulate memory over time, and its transient probabilities are amenable to an efficient numerical solution.

After recalling the main results on CTMCs, we provide an intuition for the construction of the marking process of an STPN with exponentially distributed times to fire. For a detailed treatment of the subject we refer the reader to Ajmone Marsan et al. (1995) and Stewart (1995).

Definition 2.6 (Continuous-time Markov chain). A stochastic process $\{M(t),\ t \geq 0\}$ taking values in a finite set $\mathcal{M}$ and defined on the probability space $(\Omega, \mathcal{A}, P)$ is a (time-homogeneous) continuous-time Markov chain if

$$P\{M(\tilde{t}+t) = j \mid M(\tilde{t}) = i, M(x) = i_x \ \forall x \in A_{\tilde{t}}\} = P\{M(t) = j \mid M(0) = i\}$$

for all $\tilde{t}, t > 0$, $i, j, i_x \in \mathcal{M}$, $A_{\tilde{t}} \subseteq [0, \tilde{t})$.

For a CTMC $\{M(t),\ t \geq 0\}$, the definition requires that, given the value i of $M(\tilde{t})$, the distributions of $M(t)$ for $t > \tilde{t}$ be conditionally independent of the values of $M(t)$ in any subset of past times $A_{\tilde{t}} \subseteq [0, \tilde{t})$. This property, known as the *Markov property*, implies that the current state summarizes the past history of the process.

The evolution of a CTMC is determined by its *transition probabilities*

$$P_{ij}(t) = P\{M(t) = j \mid M(0) = i\}$$

for all $t > 0$ and $i, j \in \mathcal{M}$, which give the probability that the process will be in state j at time t given the initial state i at time zero. Transition probabilities satisfy the fundamental Chapman-Kolmogorov equations.

Theorem 2.4 (Chapman-Kolmogorov equations). *Let* $\{M(t),\ t \geq 0\}$ *be a CTMC taking values in a finite set* $\mathcal{M}$ *and defined on the probability space* $(\Omega, \mathcal{A}, P)$*. Then, its transition probabilities* $P_{ij}(t)$ *with* $i, j \in \mathcal{M}$ *satisfy*

$$\mathbf{P}(\tilde{t}+t) = \mathbf{P}(\tilde{t})\,\mathbf{P}(t)$$

for all $\tilde{t}, t > 0$*, where* $\mathbf{P}(0)$ *is the* $|\mathcal{M}| \times |\mathcal{M}|$ *identity matrix.*

Given the transition probabilities $\mathbf{P}(t)$ and an initial distribution $\vec{\mu}$ such that $\mu_i = P\{M(0) = i\}$ for all $i \in \mathcal{M}$, all the finite-dimensional distributions of $\{M(t),\ t \geq 0\}$ can be computed as

$$\begin{aligned} & P\{M(t_1) = i_1, M(t_2) = i_2, \ldots, M(t_n) = i_n\} \\ & \qquad\qquad\qquad\qquad = \sum_{i_0 \in \mathcal{M}} \mu_{i_0} \prod_{k=1}^{n} P_{i_{k-1}, i_k}(t_k - t_{k-1}) \end{aligned}$$

for all $n \geq 0$, times $0 = t_0 < t_1 < \cdots < t_n$ and states $i_1, i_2, \ldots, i_n \in \mathcal{M}$.

It is common to define the transition probabilities $\mathbf{P}(t)$ through their right derivatives at zero, given by the infinitesimal generator matrix

$$\mathbf{Q} = \lim_{t \to 0^+} \frac{\mathbf{P}(t) - \mathbf{P}(0)}{t} .$$

Since the transition probabilities $P_{ij}(t)$ sum to 1 over j for all $i \in \mathcal{M}$, the infinitesimal generator matrix $\mathbf{Q}$ always satisfies

$$\begin{aligned} Q_{ii} &= \lim_{t \to 0^+} \frac{P_{ii}(t) - P_{ii}(0)}{t} \\ &= \lim_{t \to 0^+} \frac{\left(1 - \sum_{j \in \mathcal{M} \,|\, j \neq i} P_{ij}(t)\right) - 1}{t} \\ &= - \sum_{j \in \mathcal{M} \,|\, j \neq i} \left(\lim_{t \to 0^+} \frac{P_{ij}(t) - 0}{t} \right) \\ &= - \sum_{j \in \mathcal{M} \,|\, j \neq i} Q_{ij} . \end{aligned} \tag{2.2}$$

Intuitively, each element Q_{ij} measures the infinitesimal "rate of change" of the transition probability from state i to state j. For this reason, the elements of $\mathbf{Q}$ are called *transition rates.*

The transition probabilities $\mathbf{P}(t)$ of a CTMC $\{M(t),\ t \geq 0\}$ can be computed as the solutions of Kolmogorov forward differential equations

$$\frac{d\mathbf{P}(t)}{dt} = \mathbf{P}(t)\mathbf{Q}$$

with initial condition $\mathbf{P}(0) = I$. The unique solution of these ODEs is given by the matrix exponential function

$$\mathbf{P}(t) = e^{\mathbf{Q}t} = \sum_{n=0}^{\infty} \frac{(\mathbf{Q}t)^n}{n!} \tag{2.3}$$

and it can be evaluated numerically using the technique of *uniformization* described in Gross and Miller (1984). Given the transition probabilities $\mathbf{P}(t)$

and an initial state distribution $\vec{\mu}$ of a CTMC $\{M(t),\ t \geq 0\}$ defined on a probability space $(\Omega, \mathcal{A}, P)$, we have

$$\begin{aligned} P\{M(t) = j\} &= \sum_{i \in \mathcal{M}} P\{M(0) = i\}\, P\{M(t) = j \mid M(0) = i\} \\ &= \sum_{i \in \mathcal{M}} \mu_i\, P_{ij}(t) \end{aligned}$$

for all $t \geq 0$ and $j \in \mathcal{M}$.

For stochastic time Petri nets with only exponentially distributed times to fire, the infinitesimal generator matrix $\mathbf{Q}$ of the CTMC can be constructed directly from the set of reachable markings $\mathcal{M}$.

Consider a marking $i \in \mathcal{M}$ in which the transitions $E(i) \subseteq T$ are newly enabled (in particular, this is the case of the initial state of the STPN). The corresponding times to fire $\{\vec{\tau}(t),\, t \in E(i)\}$ are independent random variables defined on a probability space $(\Omega, \mathcal{A}, P)$ and distributed according to the probability density functions $f_t(x) = \lambda_t e^{-\lambda_t x}$ for $t \in E(i)$. Since the sojourn time in marking i is the minimum of the times to fire, its distribution is given by

$$\begin{aligned} H_i(x) &= P\{\exists t \in E(i) \text{ such that } \vec{\tau}(t) \leq x\} \\ &= 1 - P\{\vec{\tau}(t) > x\ \forall t \in E(i)\} \\ &= 1 - \prod_{t \in E(i)} \int_x^{\infty} f_t(u)\, du \\ &= 1 - \prod_{t \in E(i)} e^{-\lambda_t x} = 1 - e^{-\left(\sum_{t \in E(i)} \lambda_t\right) x}. \end{aligned} \tag{2.4}$$

The sojourn time in i is thus an exponential random variable with rate $\sum_{t \in E(i)} \lambda_t$ (it is the minimum of independent exponential random variables).

In addition, the probability that $\vec{\tau}(t)$ is the minimum time to fire in i, triggering the firing of transition t, is

$$\begin{aligned}
p_i(t) &= P\{\,\vec{\tau}(u) > \vec{\tau}(t)\ \forall u \in E(i) \text{ s.t. } u \neq t\,\} \\
&= \int_0^\infty \left(\prod_{u \in E(i) | u \neq t} P\{\vec{\tau}(u) > x\} \right) \lambda_t e^{-\lambda_t x}\, dx \\
&= \int_0^\infty \left(\prod_{u \in E(i) | u \neq t} e^{-\lambda_u x} \right) \lambda_t e^{-\lambda_t x}\, dx \\
&= \lambda_t \int_0^\infty e^{-(\sum_{u \in E(i)} \lambda_u) x}\, dx \\
&= \frac{\lambda_t}{\sum_{u \in E(u)} \lambda_u}
\end{aligned}$$

and the probability p_{ij} that the next state of the CTMC is $j \in \mathcal{M}$, given the current state i, is then

$$p_{ij} = \sum_{t \in E(i)\,|\, i \xrightarrow{t} j} p_i(t) = \frac{\sum_{t \in E(i)\,|\, i \xrightarrow{t} j} \lambda_t}{\sum_{u \in E(u)} \lambda_u}\,. \tag{2.5}$$

Given a sojourn time $\bar{x}$, the distribution of times to fire persistent after the firing $i \xrightarrow{t} j$ is unaffected. This property, known as the *memoryless property* of exponential random variables, is evident if we consider that, for all $t \in E(i)$,

$$P\{\vec{\tau}(t) > x + \bar{x} \mid \vec{\tau}(t) > \bar{x}\} = \frac{e^{-\lambda_t (x+\bar{x})}}{e^{-\lambda_t \bar{x}}} = e^{-\lambda_t x} = P\{\vec{\tau}(t) > x\}. \tag{2.6}$$

Thus, the distributions of times to fire enabled in the next marking j carry no memory of the previous sojourn time or fired transition t. The evolution of the marking process would be the same if these transitions were newly enabled: Eqs. (2.4) and (2.5) can then be repeatedly applied to determine the sojourn time distribution and next state probabilities in each marking. The resulting elements of the infinitesimal generator $\mathbf{Q}$ are thus given by

$$Q_{ij} = \sum_{t \in E(i)\,|\, i \xrightarrow{t} j} \lambda_t \tag{2.7}$$

for each $i, j \in \mathcal{M}$ with $i \neq j$, and by

$$Q_{ii} = - \sum_{j \in \mathcal{M}\,|\, j \neq i} Q_{ij}$$

for $i \in \mathcal{M}$. Transition probabilities can then be computed with Eq. (2.3).

The marking process is still a CTMC when times to fire are sampled according to exponential or *immediate* distributions. In this case, STPNs correspond to generalized stochastic Petri nets (GSPNs) introduced in Ajmone Marsan et al. (1984). When the firing of transition t in i results in a vanishing marking j, the next state of the CTMC $\{M(t),\ t \geq 0\}$ can be any tangible marking k reachable from j through the firing of immediate transitions. The construction of Eq. (2.7) can then be modified by considering state transitions $i \to k$ between tangible markings, with probabilities given by the product of p_{ij} and the absorption probabilities from j into each tangible marking k.

2.4.2 Markov regenerative marking process

When times to fire are sampled according to general distributions, the marking process $\{M(t),\ t \geq 0\}$ can still satisfy the Markov property immediately after selected transition firings. The corresponding time instants are called *regeneration points* and allow to decompose the process evolution in "epochs" or "cycles" that are mutually independent given their initial states.

A simple condition for the identification of regeneration points in STPNs is based on the newly enabling of transitions with generally distributed times to fire. If all the enabled GEN transitions are newly enabled after a firing, the times to fire are independently distributed with joint PDF given by the product of sampling PDFs; immediately after such firing, the marking has thus sufficient information to determine the future evolution of the process, and the absolute firing time is a regeneration point.

When regeneration points are encountered after each transition firing, the marking process is a *semi-Markov process* (SMP). This condition is still quite restrictive as it rules out, for example, a time-out (whose distribution is not memoryless) over a sequence of two actions (with memoryless or general distribution). In order to enlarge the domain of applicability of STPNs, it is thus fundamental to consider processes in which regenerations are still encountered infinitely often, but only after selected transition firings. This is the class of *Markov regenerative processes* (MRPs), also known in the literature as *semi-regenerative processes*. In this work, we consider MRPs that encounter regeneration points w.p.1 after a *bounded* number of firings. This class allows GEN transitions to persist to other GEN transitions, but the maximum number of transitions firings with persistent GEN transitions must be bounded. In contrast, multiple enabled GEN transitions are not allowed in MRPs under *enabling restriction*, which impose that at most one GEN transition be enabled in each marking, as discussed in Choi et al. (1994) and German et al. (1995). The work of Puliafito et al. (1998) provides a solution for the case of multiple GEN transitions that have been *simultaneously* enabled, while the models that we consider in this work allow GEN transitions to be enabled or disabled independently after each firing.

In the rest of this section, we present the main ideas of Markov renewal theory, with a definition of semi-Markov and Markov regenerative processes. For an in-depth introduction, we refer to Çinlar (1975) and Kulkarni (1995).

Definition 2.7 (Markov renewal sequence). Given a probability space $(\Omega, \mathcal{A}, P)$, the sequence of random variables $\{(X_n, T_n),\ n \in \mathbb{N}\}$ such that, for each $n \in \mathbb{N}$, X_n takes values in a finite set R, T_n takes values in $\mathbb{R}_{\geqslant 0}$ and $0 = T_0 \leq T_1 \leq T_2 \leq \cdots \leq T_n$, is a *Markov renewal sequence* with state space R provided that

$$\begin{aligned} P\{X_{n+1} = j,\, T_{n+1} - T_n \leq t \mid X_0, X_1, \ldots, X_n, T_0, T_1, \ldots, T_n,\} \\ = P\{X_{n+1} = j,\, T_{n+1} - T_n \leq t \mid X_n\} \quad (2.8) \end{aligned}$$

for all $n \in \mathbb{N}$, $j \in R$, and $t \in \mathbb{R}_{\geqslant 0}$.

In a Markov renewal sequence, given the current state X_n at time T_n, the time increment $(T_{n+1} - T_n)$ and next state X_{n+1} are thus independent of the previous history. We always assume that the sequence is time-homogeneous and define as *kernel* of the sequence the probabilities

$$\begin{aligned} G_{ij}(t) &= P\{X_{n+1} = j,\, T_{n+1} - T_n \leq t \mid X_n = i\} \\ &= P\{X_1 = j,\, T_1 \leq t \mid X_0 = i\} \end{aligned}$$

which give, for all $i, j \in R$ and $t \in \mathbb{R}_{\geqslant 0}$, the probability (independent of n) that state j will be reached within time t from the initial state i.

From a Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$, we can define a semi-Markov process as the right-continuous and piecewise-constant process taking the value X_n during the interval $[T_n, T_{n+1})$ for all n.

Definition 2.8 (Semi-Markov process). Given a Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$ on the probability space $(\Omega, \mathcal{A}, P)$ and with state space R, we define *semi-Markov process* the process $\{X(t),\ t \geq 0\}$ such that $X(t) = X_n$ for $t \in [T_n, T_{n+1})$ and $n \in \mathbb{N}$.

As a result of the satisfaction of Eq. (2.8) by the Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$, the semi-Markov process constructed according to Definition 2.8 satisfies the Markov property immediately after each state change: the evolution after a regeneration point T_n depends only on the state X_n and it is uniquely determined by the kernel $G_{ij}(t)$, which gives, for all $i, j \in R$, the (arbitrary) joint distribution of the sojourn time in i and next state j.

By conditioning on the sojourn time T_1 in X_0, we can express the kernel $G_{ij}(t)$ as

$$G_{ij}(t) = \int_0^t dH_i(x)\, p_{ij}(x) \qquad (2.9)$$

where

$$\begin{aligned} H_i(t) &= P\{T_1 \leq t \mid X_0 = i\} \\ &= \sum_{k \in R} P\{X_1 = k,\, T_1 \leq t \mid X_0 = i\} = \sum_{k \in R} G_{ik}(t) \end{aligned}$$

is the sojourn time distribution in state i (independent of the next state j) and

$$p_{ij}(t) = P\{X_1 = j \mid T_1 = t, X_0 = i\}$$

gives the probability that, after a sojourn in i of t time units, the next state j will be selected, with $\sum_j p_{ij}(t) = 1$ for all $i \in R$ and $t \in \mathbb{R}_{\geqslant 0}$. The interpretation of Eq. (2.9) is that a semi-Markov process allows an arbitrary sojourn time distribution $H_i(t)$ in each state i and next state probabilities $p_{ij}(t)$ that can depend on the sojourn time t.

An alternative interpretation of an SMP can be obtained by conditioning on the next state X_1 and expressing the kernel as

$$G_{ij}(t) = p_{ij}\, H_{ij}(t) \tag{2.10}$$

where

$$p_{ij} = P\{X_1 = j \mid X_0 = i\}$$

is the probability that the next state reached from i is j, with $\sum_{j \in R} p_{ij} = 1$ for all $i \in R$, and

$$H_{ij}(t) = P\{T_1 \leq t \mid X_0 = i, X_1 = j\}$$

is the sojourn time distribution given the initial state i and next state j. An SMP can thus be constructed from

- a discrete-time Markov chain of successive visited states, with transition probabilities p_{ij}, and
- sojourn time distributions $H_{ij}(t)$ that depend on both the initial state i and the next state j.

Example 2.5 (Continuous-time Markov chain). A continuous-time Markov chain $\{X(t),\ t \geq 0\}$ with infinitesimal generator $\mathbf{Q}$ and state space R is a semi-Markov process with

$$p_{ij} = -\frac{Q_{ij}}{Q_{ii}}$$

for $i, j \in R$ such that $i \neq j$, $p_{ii} = 0$ for all $i \in R$, and

$$H_{ij}(t) = 1 - e^{Q_{ii} t}$$

independently of j. Note that by setting $p_{ii} = 0$ for all $i \in R$ we modeled transitions from each state i back to itself as "prolonged sojourns" in i.

In fact, when $p_{ii} > 0$, the probability of n sojourns in i is $p_{ii}^{n-1}(1 - p_{ii})$. Since the sum of n independent exponential random variables with rate λ (corresponding to the sojourn times in i) has the Erlang PDF

$$f_n(x) = \frac{(\lambda x)^{n-1}}{(n-1)!} \lambda e^{-\lambda x},$$

by the theorem of total probability, we see that

$$\begin{aligned}
\frac{dH_{ii}(x)}{dx} &= \sum_{n=1}^{\infty} P\{n \text{ sojourns in } i\}\, f_n(x) \\
&= \sum_{n=1}^{\infty} p_{ii}^{n-1}(1 - p_{ii}) \frac{(\lambda x)^{n-1}}{(n-1)!} \lambda e^{-\lambda x} \\
&= (1 - p_{ii})\lambda e^{-\lambda x} \sum_{n=1}^{\infty} \frac{(p_{ii}\lambda x)^{n-1}}{(n-1)!} = (1 - p_{ii})\lambda e^{-(1-p_{ii})\lambda x}
\end{aligned}$$

and the sojourn in i can be modeled, without loss of generality, as a single exponential random variable with rate $(1 - p_{ii})\lambda$.

Semi-Markov processes thus generalize CTMCs by allowing general sojourn time distributions that can depend on the current state i and next state j. As a consequence of this modeling freedom, the Markov property is not satisfied for all t, but only at jump times T_n.

Markov regenerative processes relax this constraint even further: the process is not required to satisfy the Markov property after each state change, but only at selected random times (called *regeneration points*) governed by a Markov renewal sequence. Unlike SMPs, Markov regenerative processes are not constrained to be constant between regeneration points: they evolve over “regenerative epochs” delimited by regeneration points and “probabilistically restart” at the beginning of each epoch with a probability law that depends on the corresponding regeneration condition. In this sense, Markov regenerative processes also generalize regenerative processes, which restart after regeneration points with identical distributions. We remark that Markov regenerative processes are also known as *semi-regenerative processes*, as for example in Çinlar (1975), and can in general take values in any topological space. In this work, we consider Markov regenerative processes taking values in a finite set (the reachable markings), and adopt the terminology of Kulkarni (1995), which is more common in the area of stochastic Petri nets.

Definition 2.9 (Stopping time). Let $\{M(t),\ t \geq 0\}$ be a continuous-time stochastic process on the probability space $(\Omega, \mathcal{A}, P)$ with sample paths that are right-continuous and have limits from the left. Then, a real-valued random variable $T\colon \Omega \to \mathbb{R}_{\geqslant 0}$ is said to be a *stopping time* with respect to $\{M(t),\ t \geq 0\}$ if the occurrence or nonoccurrence of the event $\{T \leq t\}$ is completely determined by $\{M(u),\, 0 \leq u \leq t\}$ for all $t \geq 0$.

Definition 2.10 (Markov regenerative process). A stochastic process $\{M(t),\ t \geq 0\}$ defined on the probability space $(\Omega, \mathcal{A}, P)$ and taking values in $\mathcal{M}$ is said to be a *Markov regenerative process* if there exists a Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$ with finite state space R such that

- for each $n \in \mathbb{N}$, T_n is a stopping time for $\{M(t),\ t \geq 0\}$;
- for each $n \in \mathbb{N}$, X_n is completely determined by $\{M(u),\, 0 \leq u \leq T_n\}$;
- for each $n \in \mathbb{N}$, $0 \leq t_1 \leq \cdots \leq t_m$ with $m > 0$, and bounded function f on $\mathcal{M}^m$,

$$\begin{aligned} E\{f(M(T_n + t_1), \ldots, M(T_n + t_m)) \mid X_0 = i, M(u) \text{ for } u \leq T_n, X_n = j\} \\ = E\{f(M(t_1), \ldots, M(t_m)) \mid X_0 = j\}. \end{aligned}$$

The first condition of Definition 2.10 states that an observer who has watched $\{M(u),\, 0 \leq u \leq t\}$ can tell whether the next regeneration point T_n is less than or equal to t, or not. The second condition states that the observer can determine X_n from the evolution $\{M(u),\, 0 \leq u \leq T_n\}$ of M until T_n. Intuitively, these clauses imply a causal relation between $\{M(t),\ t \geq 0\}$ and the Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$.

The third condition states that the random variable

$$f(M(T_n + t_1), \ldots, M(T_n + t_m)),$$

which is a function of the values of M at times $T_n + t_1, \ldots, T_n + t_m$ after T_n, is independent of the past evolution $\{M(u),\, 0 \leq u \leq T_n\}$ given the value of X_n. The expectation of $f(M(T_n + t_1), \ldots, M(T_n + t_m))$ given $\{X_n = j\}$ is then the same as that of $f(M(t_1), \ldots, M(t_m))$ given $\{X_0 = j\}$: after T_n, we observe the same evolution of $M(T_n + t)$ that we would see for $M(t)$ by starting the Markov renewal sequence in $X_0 = j$.

In a sense, $M(t)$ restarts after T_n with a behavior that depends only on X_n. In turn, the evolution of $M(t)$ causes the next regeneration point T_{n+1} and regeneration condition $X_{n+1} \in R$.

Note that the state spaces $\mathcal{M}$ and R are in general distinct: in our case, $\mathcal{M}$ is the finite set of markings reachable in the STPN and R is the set of regeneration conditions X_n of the Markov renewal sequence. In Chapter 4 we will show that the marking process encounters a regeneration point when, immediately after a firing, all GEN transitions are either disabled, or they have been enabled for a *deterministic* time. Thus, the set R contains pairs $(m, \vec{d})$ in which $m \in \mathcal{M}$ is the marking after the firing and $\vec{d}$ is the *enabling vector* with deterministic enabling times $d_i \in \mathbb{R}_{\geqslant 0}$ for the GEN transitions enabled by m. Each pair $(m, \vec{d})$ has sufficient information to determine the joint PDF of the times to fire of all enabled transitions, and to characterize the evolution of the STPN after the regeneration point.

This idea extends state-of-the-art analysis of regeneration points in stochastic Petri nets. In fact, the usual notion of regeneration corresponds to time instants of firings after which all GEN transitions are either disabled

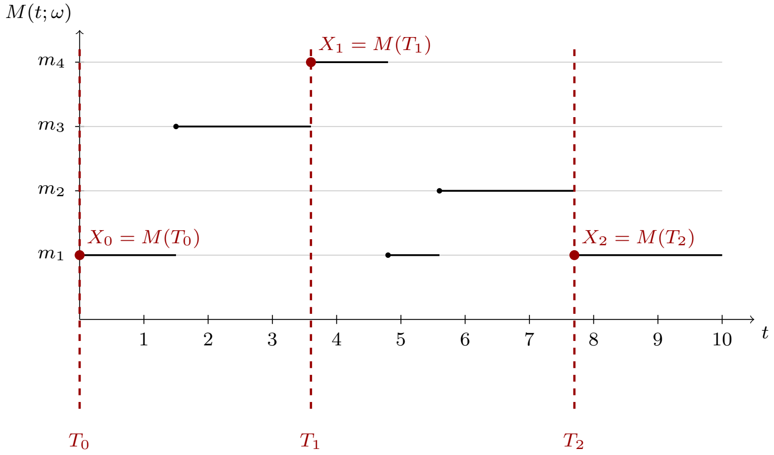


Fig. 2.6: Sample path of a Markov regenerative process $\{M(t),\ t \geq 0\}$ with Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$ on the same state space.

or *newly enabled*. This condition always results in regenerations of the form $(m, \vec{0})$: the marking at the regeneration point has sufficient information to characterize the future evolution of the STPN. In this case, the set R of regeneration conditions is a set of markings, $X_n = M(T_n)$ for all n, and the third condition of Definition 2.10 becomes the Markov property at T_n. In Fig. 2.6 we report an example of the sample paths of a Markov regenerative process with Markov renewal sequence defined on the same state space.

The repetitive structure of Markov regenerative processes can be leveraged to compute transient probabilities. Let $\{M(t),\ t \geq 0\}$ be a Markov regenerative process defined on the probability space $(\Omega, \mathcal{A}, P)$ and taking values in $\mathcal{M}$, and let $\{(X_n, T_n),\ n \in \mathbb{N}\}$ be its Markov renewal sequence with state space R. We define the transition probabilities of $\{M(t),\ t \geq 0\}$ as

$$P_{ij}(t) = P\{M(t) = j \mid X_0 = i\}$$

for all regeneration conditions $i \in R$ and states $j \in \mathcal{M}$. We can then apply a "renewal argument" to $P_{ij}(t)$ so as to distinguish whether the first regeneration T_1 is reached after t or within t, obtaining

$$\begin{aligned} P_{ij}(t) = P\{M(t) = j \mid X_0 = i\} &= P\{M(t) = j,\, T_1 > t \mid X_0 = i\} \\ &\quad + P\{M(t) = j,\, T_1 \leq t \mid X_0 = i\}\,. \end{aligned} \tag{2.11}$$

We define the first term

$$L_{ij}(t) = P\{M(t) = j, T_1 > t \mid X_0 = i\} \tag{2.12}$$

as *local kernel* of $\{M(t),\ t \geq 0\}$ and, by conditioning on all possible values for the first regeneration condition X_1 and regeneration time T_1, we obtain for the second term

$$\begin{aligned} &P\{M(t) = j, T_1 \leq t \mid X_0 = i\} \\ &\qquad = \sum_{k \in R} \int_0^t P\{M(t) = j \mid X_0 = i, X_1 = k, T_1 = u\}\, dG_{ik}(u) \\ &\qquad = \sum_{k \in R} \int_0^t P\{M(t-u) = j \mid X_0 = k\}\, dG_{ik}(u) \\ &\qquad = \sum_{k \in R} \int_0^t P_{ik}(t-u)\, dG_{ik}(u) \qquad (2.13) \end{aligned}$$

where

$$G_{ij}(t) = P\{X_1 = j,\, T_1 \leq t \mid X_0 = i\}$$

is the kernel of the Markov renewal sequence (also called *global kernel* of the Markov regenerative process) and

$$P\{M(t) = j \mid X_0 = i, X_1 = k, T_1 = u\} = P\{M(t-u) = j \mid X_0 = k\}$$

is a consequence of the third condition of Definition 2.10. By substituting the identities of Eq. (2.12) and Eq. (2.13) into Eq. (2.11), we obtain

$$P_{ij}(t) = L_{ij}(t) + \sum_{k \in R} \int_0^t P_{ik}(t-u)\, dG_{ik}(u)$$

for all $i \in R$ and $j \in \mathcal{M}$, or, equivalently, the matrix form

$$\mathbf{P}(t) = \mathbf{L}(t) + \int_0^t d\mathbf{G}(u)\, \mathbf{P}(t-u)\,. \qquad (2.14)$$

Eq. (2.14) comprises a set of Volterra integral equations of the second kind, known as *generalized Markov renewal equations.* The local kernel $\mathbf{L}$ characterizes the evolution of $\{M(t),\ t \geq 0\}$ within a regenerative epoch, while the global kernel $\mathbf{G}$ characterizes the successive regenerations and the duration of the epochs.

Once the kernels are known, the solution of Eq. (2.14) can be performed in the time domain through a discretization approach, or in the frequency domain through Laplace transform. For a detailed discussion we refer to Brunner and van der Houwen (1986) and Kulkarni (1995).

In Chapter 4 we will present an algorithm for the detection of regeneration points and the computation of the kernels of the marking process of STPNs in which GEN transitions can persist to the firing or to the newly enabling of other GEN transitions, but regenerations are encountered w.p.1 in a bounded number of transition firings.

Chapter 3
Stochastic State Classes

Stochastic state classes compute the logic state and joint probability density function (PDF) of the active timers of a discrete-event system *given a sequence of events.* Each event is triggered by the elapse of a timer: therefore, its occurrence conditions the PDF of other persistent timers, which must be lower than the elapsed timer and are decreased, after the event, by its random value. The closed-form PDF of times to fire persistent after a sequence of events is computed from the initial sampling PDFs through repeated integration and conditioning over the set of timer values "compatible" with the sequence.

The method of stochastic state classes was introduced for the analysis of stochastic time Petri nets in Vicario et al. (2009) and Carnevali et al. (2009), but it has been applied also to other stochastic discrete-event systems, such as the stochastic extensions of timed automata presented in Ballarini et al. (2013). The calculus requires sampling PDFs with closed-form antiderivatives and timers decreasing with the same rate in each state. The latter condition allows to represent the supports of joint PDFs as *zones* encoded by *difference bound matrices* (DBMs).

In this perspective, stochastic state classes enrich *nondeterministic state classes* with probability. Reachability analysis based on state classes, as for example in Berthomieu and Diaz (1991) and Vicario (2001), computes the continuous set of values for the active timers given a sequence of firings; stochastic state classes enrich this set with a closed-form PDF which allows to compute quantitative measures on the transient behavior of the system.

3.1 Definition

In this chapter, we present stochastic state classes for the marking and times to fire PDF of a stochastic time Petri net, conditioned to a sequence of transition firings. Given an STPN $\langle P, T, A^-, A^+, A^\circ, EFT, LFT, f, w\rangle$, stochastic state classes are defined as follows.

Definition 3.1 (Stochastic state class). A (transient) *stochastic state class* is a tuple

$$\Sigma = \langle m, D_{\langle \tau_{age}, \vec{\tau}\rangle}, f_{\langle \tau_{age}, \vec{\tau}\rangle}\rangle$$

where

- $m \in \mathbb{N}^P$ is a marking;
- $f_{\langle \tau_{age}, \vec{\tau}\rangle}$ is the PDF (immediately after a firing) of the random vector $\langle \tau_{age}, \vec{\tau}\rangle$ including the age timer τ_{age} and the times to fire $\vec{\tau} = (\tau_1, \ldots, \tau_n)$ of transitions $E(m) = \{t_1, \ldots, t_n\}$ enabled by m;
- $D_{\langle \tau_{age}, \vec{\tau}\rangle} \subseteq \mathbb{R}^{n+1}$ is the support of $f_{\langle \tau_{age}, \vec{\tau}\rangle}$.

Note that Definition 3.1 includes the additional τ_{age} timer in stochastic state classes. This timer is not associated with any transition: it is initially set to the deterministic value 0 and decreased with unitary rate after each firing, similarly to all of the other times to fire. Its value is thus equal to the *opposite* of the absolute time of the last firing, and the joint PDF $f_{\langle \tau_{age}, \vec{\tau}\rangle}$ allows to compute measures for a specific subset of times to fire and firing time values. The reason for encoding the opposite of the absolute time is technical: when all the timers in the model decrease with the same rate, the support of their joint PDF can be represented as zones encoded by difference bound matrices. In summary, a stochastic state class includes a marking, a set of values for τ_{age} and for the times to fire of enabled transitions, and a joint PDF on these values.

In the initial stochastic state class $\Sigma_0 = \langle m_0, D_{\langle \tau_{age}, \vec{\tau}_0\rangle}, f_{\langle \tau_{age}, \vec{\tau}_0\rangle}\rangle$, the marking m_0 is the initial marking of the STPN, τ_{age} has deterministic value 0, and the times to fire of the enabled transitions $E(m_0) = \{t_1, \ldots, t_n\}$ are distributed according to a given joint PDF $f_{\vec{\tau}_0}$ with support $D_{\vec{\tau}_0} \subseteq \mathbb{R}^n_{\geqslant 0}$. Therefore, we have

$$D_{\langle \tau_{age}, \vec{\tau}_0\rangle} = [0, 0] \times D_{\vec{\tau}_0}$$

and

$$f_{\langle \tau_{age}, \vec{\tau}_0\rangle}(x_{age}, x_1, \ldots, x_n) = \delta(x_{age}) \cdot f_{\vec{\tau}_0}(x_1, \ldots, x_n)\,.$$

Without loss of generality, we assume that, in the initial state $s_0 = \langle m_0, \vec{\tau}_0\rangle$ of the STPN, each transition $t_i \in E(m_0)$ has been enabled for a deterministic time $d_i \leq LFT(t_i)$. As a consequence, the times to fire $\vec{\tau}_0(t_1), \vec{\tau}_0(t_2), \ldots, \vec{\tau}_0(t_n)$ are independent random variables with product-form joint PDF

$$f_{\vec{\tau}_0}(x_1, \ldots, x_n) = \prod_{i=1}^{n} \frac{f_{t_i}(x_i + d_i)}{\int_{\max\{d_i, EFT(t_i)\}}^{LFT(t_i)} f_{t_i}(u)\, du} \tag{3.1}$$

on the support

$$D_{\vec{\tau}_0} = \prod_{i=1}^{n} \left[\max\{0, EFT(t_i) - d_i\}, LFT(t_i) - d_i \right]. \tag{3.2}$$

In particular, $d_i = 0$ for all $i = 1, 2, \ldots, n$ when each enabled transition is newly enabled in the initial state.

Given a class Σ, the state PDF conditioned to the firing of a transition γ at an absolute time in the interval I is given by the *successor class* of Σ through γ and I.

Definition 3.2 (Succession relation). We say that, with probability μ, the stochastic state class $\Sigma' = \langle m', D'_{\langle \tau_{age}, \vec{\tau} \rangle}, f'_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ is the *successor* of $\Sigma = \langle m, D_{\langle \tau_{age}, \vec{\tau} \rangle}, f_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ through $\gamma \in E(m)$ at some time in I, and we write $\Sigma \xRightarrow{\gamma, I, \mu} \Sigma'$, if, given that the marking of the STPN is m and $\langle \tau_{age}, \vec{\tau} \rangle$ is a random vector distributed over $D_{\langle \tau_{age}, \vec{\tau} \rangle}$ according to $f_{\langle \tau_{age}, \vec{\tau} \rangle}$, then:

- the transition γ has non-null probability μ to fire in Σ at some time in I;
- if γ fires in Σ at some time in I, its firing yields the marking m' and, conditioned to this event, the new times to fire vector is distributed over $D'_{\langle \tau_{age}, \vec{\tau} \rangle}$ according to $f'_{\langle \tau_{age}, \vec{\tau} \rangle}$.

The relation $\xRightarrow{\gamma, I, \mu}$ can be enumerated through a calculus for the computation of the probability of outgoing events, and for the symbolic derivation of the support and closed-form PDF of $\langle \tau_{age}, \vec{\tau} \rangle$ in successor classes.

3.2 Calculus of successor classes

Successors of stochastic state classes according to Definition 3.2 can be computed through symbolic integration and conditioning over the set of times to fire values compatible with the firing. This calculus satisfies two fundamental properties, discussed in detail in Carnevali et al. (2009):

1. The support of joint PDFs can be encoded as difference bound matrices.
2. When each transition t is distributed according to an *expolynomial* PDF

$$f_t(x) = \sum_{j=1}^{m} c_j \, x^{a_j} e^{-\lambda_j x}$$

for some $c_j \in \mathbb{R}$, $a_j \in \mathbb{N}$, and $\lambda_j \in \mathbb{R}_{\geqslant 0}$, the joint PDF of $\langle \tau_{age}, \vec{\tau} \rangle$ after each firing is a piecewise function defined on a partition of the DBM support into sub-zones that can also be represented as DBMs. The function is continuous and each piece is a multivariate expolynomial of the form

$$f_{\langle \tau_{age},\vec{\tau}\rangle}(x_{age}, x_1, \ldots, x_n) = \sum_{j=1}^{k} c_j \, x_{age}^{a_{0j}} e^{-\lambda_{0j} x_{age}} \left(\prod_{i=1}^{n} x_i^{a_{ij}} e^{-\lambda_{ij} x_i} \right)$$

with $c_j \in \mathbb{R}$, $a_{ij} \in \mathbb{N}$, and $\lambda_{ij} \in \mathbb{R}_{\geqslant 0}$ for $j = 1, 2, \ldots, k$ and $i = 0, 1, \ldots, n$.

The first property allows an efficient computation of the supports of successor classes, which are amenable to the calculus of successor state classes of Vicario (2001). The second property provides an algorithmic approach to the computation of indefinite integrals (antiderivatives), which is possible for each variable $x_{age}, x_1, \ldots, x_n$ of a multivariate expolynomial through integration by parts. In the following, we define difference bound matrices and recall the main steps of the calculus.

Difference bound matrices, introduced in Dill (1990) and leveraged in Berthomieu and Diaz (1991) and Vicario (2001) for the analysis of time Petri nets, represent upper and lower bounds of pairwise differences $x_i - x_j$ between variables x_k for $k = 1, \ldots, n$. By introducing a fictitious variable $x_* = 0$, a difference bound matrix **B** represents the terms B_{ij} of all the constraints of the form $x_i - x_j \leq B_{ij}$ and $x_i = x_i - x_* \leq B_{i*}$; lower bounds of the form $x_i - x_j \geq c$ or $x_i \geq c$ are represented by imposing $x_j - x_i \leq B_{ji}$ with $B_{ji} = -c$, or $x_* - x_i \leq B_{*i}$ with $B_{*i} = -c$, respectively. Let $\mathbb{Q}_\infty$ denote the set of rational numbers $\mathbb{Q}$ extended with a "positive infinity" element ∞ and its opposite $-\infty = -(\infty)$, such that, for all $c \in \mathbb{Q}$, $c + \infty = \infty$, $c - \infty = -\infty$, $c < \infty$, and $c > -\infty$.

Definition 3.3 (Difference bound matrix). Given a set of variables $\{x_1, x_2, \ldots, x_n\}$, a difference bound matrix is a matrix **B** of upper bounds $B_{ij} \in \mathbb{Q}_\infty$ for the pairwise differences $x_i - x_j$ for all $i, j \in \{*, 1, 2, \ldots, n\}$ such that $i \neq j$, with $x_* := 0$.

A difference bound matrix on the variables $\{x_1, x_2, \ldots, x_n\}$ identifies the convex subset of $\mathbb{R}^n$

$$\left\{ \vec{x} \in \mathbb{R}^n \mid x_i - x_j \leq B_{ij} \text{ for all } i, j \in \{*, 1, 2, \ldots, n\} \text{ such that } i \neq j \right\}$$

which is said to be a *zone*. Distinct DBMs can identify the same zone when some constraint $x_i - x_j \leq B_{ij}$ is the logical implication of other constraints: in this case, in fact, if the constraint is relaxed by increasing B_{ij}, the same zone is identified. In order to remove this ambiguity, we always consider DBMs in *normal form*.

Definition 3.4 (DBM in normal form). A difference bound matrix **B** on the variables $\{x_1, x_2, \ldots, x_n\}$ is in *normal form* if $B_{ik} + B_{kj} \geq B_{ij}$ for all $i, j, k \in \{*, 1, 2, \ldots, n\}$ such that $i \neq k$, $k \neq j$ and $i \neq j$.

As a consequence of the definition, when a DBM is in normal form, the coefficient B_{ij} always represents the tightest upper bound on the difference $x_i - x_j$ for all $i, j \in \{*, 1, 2, \ldots, n\}$ with $i \neq j$, even when the constraint

$x_i - x_j \leq B_{ij}$ is made irrelevant by other constraints. Given a DBM zone on n variables, its normal form can be computed with time complexity $O(n^3)$ by the Floyd-Warshall algorithm for the all-pairs shortest path problem.

The zone identified by a normal-form DBM $\mathbf{B}$ on the variables $\{x_1, x_2, \ldots, x_n\}$ can be projected over a subset of variables $I \subset \{x_1, x_2, \ldots, x_n\}$ by removing from $\mathbf{B}$ the rows and columns that correspond to variables not in I; we indicate the resulting DBM as $\mathbf{B} \downarrow I$.

The Cartesian product between the zone over $\{x_1, x_2, \ldots, x_n\}$ defined by DBM $\mathbf{B}$ and a real interval $[a, b]$ for $a, b \in \mathbb{Q}$ is identified by the DBM $\mathbf{B}'$ over the variables $\{x_1, x_2, \ldots, x_n, x_{n+1}\}$ and such that, for all $i \neq j$,

$$B'_{ij} = \begin{cases} B_{ij} & \text{if } i \neq n+1 \text{ and } j \neq n+1, \\ b & \text{if } i = n+1 \text{ and } j = *, \\ b + B_{*j} & \text{if } i = n+1 \text{ and } j \neq *, \\ -a & \text{if } i = * \text{ and } j = n+1, \\ B_{i*} - a & \text{if } i \neq * \text{ and } j = n+1. \end{cases}$$

Difference bound matrices can encode the piecewise support $D_{\langle \tau_{age}, \vec{\tau} \rangle}$ of PDFs $f_{\langle \tau_{age}, \vec{\tau} \rangle}$ for vector $\langle \tau_{age}, \vec{\tau} \rangle$: the zone in $\mathbb{R}^{n+1}$ over the variables $x_{age}, x_1, x_2, \ldots, x_n$ gives the possible values for τ_{age} and for the times to fire $\vec{\tau}(t_1), \vec{\tau}(t_2), \ldots, \vec{\tau}(t_n)$ of enabled transitions.

Given a stochastic state class $\Sigma = \langle m, D_{\langle \tau_{age}, \vec{\tau} \rangle}, f_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ with $E(m) = \{t_1, t_2, \ldots, t_n\}$, the probability μ that t_k with $1 \leq k \leq n$ will fire first at some time in I, as well as the stochastic state class $\Sigma' = \langle m', D'_{\langle \tau_{age}, \vec{\tau} \rangle}, f'_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ after the firing, can be computed through the following steps.

Succession probability. Given the stochastic state class $\Sigma = \langle m, D_{\langle \tau_{age}, \vec{\tau} \rangle}, f_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ after the previous firing, the probability that t_k has minimum time to fire and that the next firing will be at an absolute time in I is nonzero only if the zone

$$D^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle} = \{ (x_{age}, \vec{x}) \in D_{\langle \tau_{age}, \vec{\tau} \rangle} \mid x_k \leq x_j \text{ for all } j \neq k,\ x_k - x_{age} \in I \}$$

is not empty. The DBM $\mathbf{B}^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}$ of this zone can be constructed from the DBM $\mathbf{B}$ of $D_{\langle \tau_{age}, \vec{\tau} \rangle}$ as the normal form of

$$B^{k,I}_{ij} = \begin{cases} \min(B_{ij}, \sup I) & \text{if } i = k \text{ and } j = age, \\ \min(B_{ij}, -\inf I) & \text{if } i = age \text{ and } j = k, \\ \min(B_{ij}, 0) & \text{if } i = k \text{ and } j = 1, \ldots, k-1, k+1, \ldots, n, \\ B_{ij} & \text{otherwise.} \end{cases}$$

Then, $D^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}$ is nonempty if and only if $B^{k,I}_{ij} + B^{k,I}_{ji} \geq 0$ for all $i \neq j$. Under this assumption, when $\vec{\tau}(t_k)$ is not deterministic, the probability value μ can be computed as

$$\mu = \int_{D^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}} f_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, x_1, x_2, \ldots, x_n) \, dx_{age} \, d\vec{x}$$

while, when $\vec{\tau}(t_k)$ is deterministic, the probability value μ is given by

$$\frac{w(t_k)}{\sum_{u \in E_{\min}} w(u)} \int_{D^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}} f_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, x_1, x_2, \ldots, x_n) \, dx_{age} \, d\vec{x}$$

where $E_{\min} = \{t_i \in E(m) \mid \tau(t_i) \text{ is deterministic and minimum}\}$ and $w \colon T \to \mathbb{R}_{>0}$ is the weight assignment function of the STPN.

Conditioning. The PDF $f_{\langle \tau_{age}, \vec{\tau} \rangle}$ is divided by μ to condition the random vector $\langle \tau_{age}, \vec{\tau} \rangle$ to the firing of t_k at some time in I, resulting in the probability density function

$$f^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, x_1, x_2, \ldots, x_n) = \frac{1}{\mu} \, f_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, x_1, x_2, \ldots, x_n)$$

over the support $D^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}$.

Time advancement and projection. According to the semantics of STPNs, the time to fire $\vec{\tau}(t_k)$ must be subtracted from that of each persistent transition and discarded from $\vec{\tau}$ (since t_k is always newly enabled after its firing). The PDF of the vector $\langle \sigma_{age}, \vec{\sigma} \rangle$ where $\sigma_{age} = \tau_{age} - \vec{\tau}(t_k)$ and

$$\begin{aligned} \vec{\sigma} = \big(& \vec{\tau}(t_1) - \vec{\tau}(t_k), \ldots, \vec{\tau}(t_{k-1}) - \vec{\tau}(t_k), \\ & \qquad \vec{\tau}(t_{k+1}) - \vec{\tau}(t_k), \ldots, \vec{\tau}(t_n) - \vec{\tau}(t_k) \big), \end{aligned}$$

conditioned to the firing of t_k in I, can be computed from $f^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}$ as

$$\begin{aligned} & f_{\langle \sigma_{age}, \vec{\sigma} \rangle}(x_{age}, x_1, \ldots, x_{k-1}, x_{k+1}, \ldots, x_n) = \\ & \quad \int_{L_k(x_{age}, \vec{x})}^{U_k(x_{age}, \vec{x})} f^{k,I}_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age} + x_k, x_1 + x_k, \ldots, x_k, \ldots, x_n + x_k) \, dx_k \end{aligned} \tag{3.3}$$

where

$$\begin{aligned} U_k(x_{age}, \vec{x}) = \min \Big(& \{B^{k,I}_{k*}\} \cup \{B^{k,I}_{j*} - x_j \text{ for all} \\ & \qquad j = age, 1, \ldots, k-1, k+1, \ldots, n\} \Big) \end{aligned}$$

and

$$L_k(x_{age}, \vec{x}) = \max\Big(\{-B^{k,I}_{*k}\} \cup \{-B^{k,I}_{*j} - x_j \text{ for all } j = age, 1, \ldots, k-1, k+1, \ldots, n\}\Big)$$

give the upper and lower bounds for the integral of Eq. (3.3) as a function of the variables $x_{age}, x_1, \ldots, x_{k-1}, x_{k+1}, \ldots, x_n$, which can result in a piecewise function with at most $(n+1)(n+1)$ sub-zones.

Disabling. The times to fire of disabled transitions are discarded by integrating $f_{\langle\sigma_{age},\vec{\sigma}\rangle}$ over the domain of the corresponding variables, resulting in the marginal PDF of times to fire of persistent transitions. The support of this PDF is obtained through the DBM projection over the times to fire variables of persistent transitions.

Newly enabling. The times to fire of newly enabled transitions are introduced multiplying the PDF of persistent transitions by the PDF f_t associated with each newly enabled transition t. The support $D'_{\langle\tau_{age},\vec{\tau}\rangle}$ of the resulting PDF $f'_{\langle\tau_{age},\vec{\tau}\rangle}$ is computed as the Cartesian product of the support of the PDF of persistent transitions with the supports $[EFT(t), LFT(t)]$ of the times to fire of newly enabled transitions.

A detailed description of the computation of successor stochastic state classes over partitioned DBM supports can be found in Carnevali et al. (2009). In Vicario et al. (2009), an efficient technique is presented to encompass deterministic time to fire variables in the calculus. The closed-form computation of expolynomial PDFs with piecewise representation over DBM zone supports is implemented in the ORIS Tool, described in Bucci et al. (2010).

3.3 Transient measures

Given an initial stochastic state class Σ_0, a sequence of transition firings $\gamma_1, \gamma_2, \ldots, \gamma_n$, and a sequence of real intervals $I_1, I_2, \ldots, I_n$ for the absolute firing times, the sequence of stochastic state classes

$$\Sigma_0 \xrightarrow{\gamma_1, I_1, \mu_1} \Sigma_1 \xrightarrow{\gamma_2, I_2, \mu_2} \cdots \xrightarrow{\gamma_n, I_n, \mu_n} \Sigma_n \tag{3.4}$$

with $\Sigma_i = \langle m_i, D_i, f_i\rangle$ for $i = 1, \ldots, n$ computes the probability μ_i of each firing event, and the resulting marking m_i and joint PDF f_i (over the support D_i) for τ_{age} and for the times to fire $\vec{\tau}$ after the firing. The probability density functions f_i, computed from the initial sampling PDFs of enabled transitions, allow to derive important measures on the transient behavior of the STPN along the sequence of transition firings.

From the definition of the successor relation (Definition 3.2), it follows directly that, given an initial state distributed according to Σ_0, the probability that the STPN will perform the sequence of firings $\gamma_1, \gamma_2, \ldots, \gamma_n$ at times in the intervals $I_1, I_2, \ldots, I_n$ is given by

$$p_{seq}(\Sigma_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n) := \prod_{i=1}^{n} \mu_i \tag{3.5}$$

if $\mu_i > 0$ for $i = 1, 2, \ldots, n$ (and thus the sequence of Eq. (3.4) is defined), or it is equal to zero otherwise.

To require the completion of the sequence of transition firings $\gamma_1, \gamma_2, \ldots, \gamma_n$ within a maximum time t, thus reaching the stochastic state class Σ_n within time t, we can set $I_i = [0, \infty)$ for $i < n$ and $I_n = [0, t]$ in Eq. (3.5). We indicate the corresponding probability as

$$p_{reach}(\Sigma_n, t) := p_{seq}(\Sigma_0, \gamma_1, [0, \infty), \gamma_2, [0, \infty), \ldots, \gamma_n, [0, t]) . \tag{3.6}$$

Finally, we are interested in the probability that, given an initial state class Σ_0, the STPN has completed the sequence of firings $\gamma_1, \gamma_2, \ldots, \gamma_n$ within time t without performing any subsequent firing. We first compute the sequence of stochastic state classes as in Eq. (3.6) with $I_i = [0, \infty)$ for $i < n$ and $I_n = [0, t]$; then, given the last stochastic state class $\Sigma_n = \langle m, D_{\langle \tau_{age}, \vec{\tau} \rangle}, f_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$, we restrict its support $D_{\langle \tau_{age}, \vec{\tau} \rangle}$ to

$$D^{in,t}_{\langle \tau_{age}, \vec{\tau} \rangle} = \{(x_{age}, \vec{x}) \in D_{\langle \tau_{age}, \vec{\tau} \rangle} \mid x_i - x_{age} > t \text{ for all } i \neq age\},$$

where the constraints $x_i - x_{age} > t$ for all $i \neq age$ impose that the absolute time of next firing be higher than t. Then, the desired measure can be computed as

$$p_{in}(\Sigma_n, t) = p_{reach}(\Sigma_n, t) \int_{D^{in,t}_{\langle \tau_{age}, \vec{\tau} \rangle}} f_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, x_1, x_2, \ldots, x_n) \, dx_{age} \, d\vec{x} . \tag{3.7}$$

In Section 4.4, we will consider the *transient tree* of stochastic state classes resulting from a finite set

$$S \subset \{\gamma_1, \gamma_2, \ldots, \gamma_k \mid k \in \mathbb{N} \text{ and } \gamma_i \in T \text{ for } i = 1, 2, \ldots, k\}$$

of transition sequences: each edge of the tree is labeled with a fired transition and each node is associated with the stochastic state class reached after firing the sequence of transitions that label the edges from the root to the node (without constraints on the absolute firing times). Given a time instant t, we are interested in computing:

- the measure $p_{in}(\Sigma_{inner}, t)$ for each stochastic state class Σ_{inner} associated with an inner node of the tree;
- the measure $p_{reach}(\Sigma_{leaf}, t)$ for each stochastic state class Σ_{leaf} associated with a leaf node of the tree.

On the one hand, the measures $p_{in}(\Sigma_{inner}, t)$ give the probabilities that the system has performed all and only the firings in distinct, strict prefixes of firing sequences in S; on the other hand, the measures $p_{reach}(\Sigma_{leaf}, t)$ give the probabilities that the system has completed maximal sequences in S within time t (and possibly other subsequent transition firings). These measures correspond to mutually exclusive events and give a full picture of the system behavior at time t with respect to the firing sequences in S.

The measure $p_{reach}(\Sigma_{leaf}, t)$ can be computed with Eq. (3.6) by considering the sequence of stochastic state classes from the root of the tree to Σ_{leaf}. The measures $p_{in}(\Sigma_{inner}, t)$ for the inner nodes of the same sequence can be computed by restricting the PDF support $D_{\langle \tau_{age}, \vec{\tau} \rangle}$ of stochastic state classes $\Sigma_{inner} = \langle m, D_{\langle \tau_{age}, \vec{\tau} \rangle}, f_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ on the path to Σ_{leaf} as

$$\begin{aligned} D^{inner,t}_{\langle \tau_{age}, \vec{\tau} \rangle} = \{ & (x_{age}, \vec{x}) \in D_{\langle \tau_{age}, \vec{\tau} \rangle} \mid \\ & x_i - x_{age} > t \text{ for all } i \neq age \text{ and } -x_{age} \leq t \} \end{aligned}$$

and then computing $p_{in}(\Sigma_{inner}, t)$ as

$$p_{in}(\Sigma_{inner}, t) = \int_{D^{inner,t}_{\langle \tau_{age}, \vec{\tau} \rangle}} f_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, x_1, x_2, \ldots, x_n)\, dx_{age}\, d\vec{x}\,. \qquad (3.8)$$

With respect to Eq. (3.7), this approach avoids the repeated enumeration of inner nodes of the transient tree. The stochastic state classes of the transient tree are computed only once, and the integrals of Eq. (3.8) are then evaluated on inner nodes for each value of t.

Example 3.1. Fig. 3.1 introduces a small sized running example inspired by Martinez and Haverkort (2006) that represents a G/D/1/2/2 queue with server breakdowns. Tokens in places *free* and *buffer* represent customers in the idle state or inside the queue, respectively. Idle customers arrive in series after times uniformly distributed over $[1, 2]$ (transition *arrival*), while service has a deterministic duration 1.5 (transition *service*) and requires the server to be *operational*; times to failure are exponentially distributed with rate 0.1 (transition *fail*), and repairs are completed in a time uniformly distributed over $[1, 2]$ (transition *restart*). Fig. 3.2 presents the stochastic state classes for the sequence of transition firings fail, restart, arrival, service, without constraints on the absolute firing times, and from an initial stochastic state class with marking *2free operational* and zero enabling times (all enabled transitions are newly enabled). The probability that the STPN performs the sequence of transitions is given by the product of succession probabilities (reported up to the fourth significant figure).

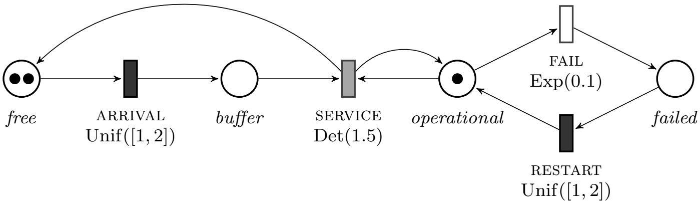


Fig. 3.1: STPN model of a G/D/1/2/2 queue with server breakdowns.

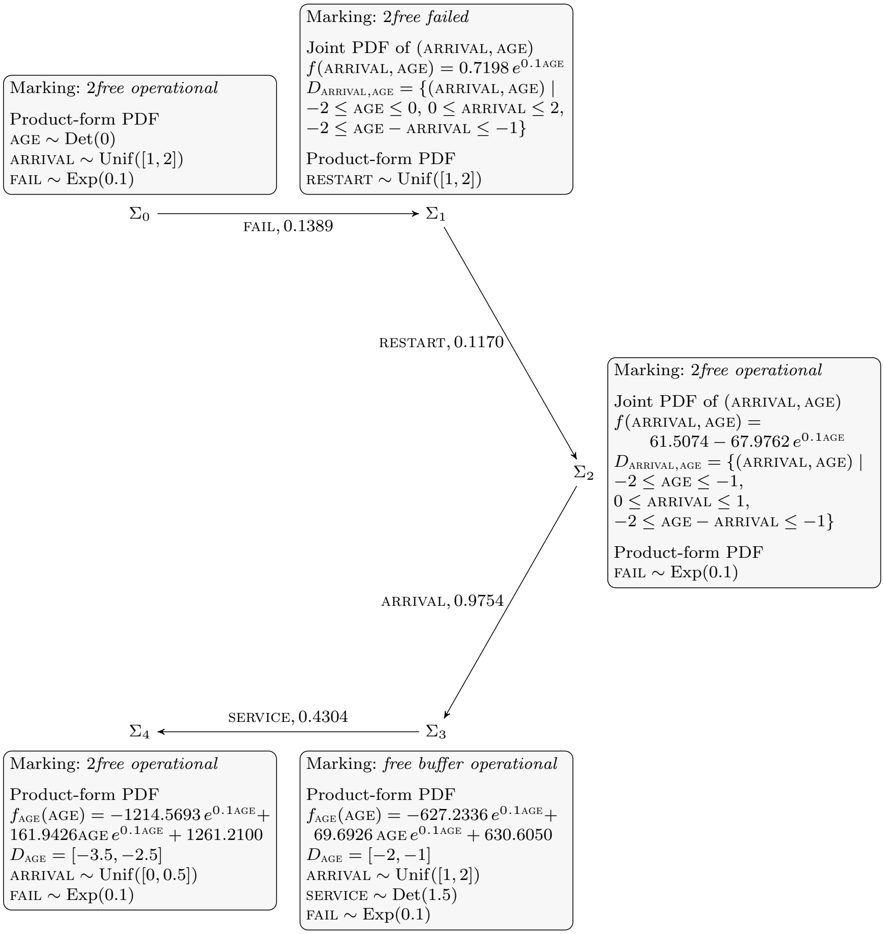

Marking: 2*free operational*
Product-form PDF
AGE $\sim$ Det(0)
ARRIVAL $\sim$ Unif([1, 2])
FAIL $\sim$ Exp(0.1)
Marking: 2*free failed*
Joint PDF of (ARRIVAL, AGE)
$f(\text{ARRIVAL}, \text{AGE}) = 0.7198\, e^{0.1\text{AGE}}$
$D_{\text{ARRIVAL},\text{AGE}} = \{(\text{ARRIVAL}, \text{AGE}) \mid$
$-2 \leq \text{AGE} \leq 0,\ 0 \leq \text{ARRIVAL} \leq 2,$
$-2 \leq \text{AGE} - \text{ARRIVAL} \leq -1\}$
Product-form PDF
RESTART $\sim$ Unif([1, 2])
Σ_0
FAIL, 0.1389
Σ_1
RESTART, 0.1170
Σ_2
Marking: 2*free operational*
Joint PDF of (ARRIVAL, AGE)
$f(\text{ARRIVAL}, \text{AGE}) =$
$61.5074 - 67.9762\, e^{0.1\text{AGE}}$
$D_{\text{ARRIVAL},\text{AGE}} = \{(\text{ARRIVAL}, \text{AGE}) \mid$
$-2 \leq \text{AGE} \leq -1,$
$0 \leq \text{ARRIVAL} \leq 1,$
$-2 \leq \text{AGE} - \text{ARRIVAL} \leq -1\}$
Product-form PDF
FAIL $\sim$ Exp(0.1)
ARRIVAL, 0.9754
Σ_4
SERVICE, 0.4304
Σ_3
Marking: 2*free operational*
Product-form PDF
$f_{\text{AGE}}(\text{AGE}) = -1214.5693\, e^{0.1\text{AGE}} +$
$161.9426\text{AGE}\, e^{0.1\text{AGE}} + 1261.2100$
$D_{\text{AGE}} = [-3.5, -2.5]$
ARRIVAL $\sim$ Unif([0, 0.5])
FAIL $\sim$ Exp(0.1)
Marking: *free buffer operational*
Product-form PDF
$f_{\text{AGE}}(\text{AGE}) = -627.2336\, e^{0.1\text{AGE}} +$
$69.6926\, \text{AGE}\, e^{0.1\text{AGE}} + 630.6050$
$D_{\text{AGE}} = [-2, -1]$
ARRIVAL $\sim$ Unif([1, 2])
SERVICE $\sim$ Det(1.5)
FAIL $\sim$ Exp(0.1)

Fig. 3.2: Stochastic state classes for the sequence of events FAIL, RESTART, ARRIVAL, SERVICE in the queue of Fig. 3.1.

Chapter 4
Regenerative Transient Analysis

The evolution of a Markov regenerative process can be decomposed into independent "epochs" delimited by an infinite sequence of random times called *regeneration points*. The *regeneration condition* at each regeneration point provides sufficient information to characterize the future evolution of the system, which probabilistically "restarts", oblivious of its previous history. As discussed in Section 2.4.2, this repetitive structure of the stochastic process can be leveraged to compute transient probabilities as the solution of a system of integral equations governed by a *global* and a *local* kernel. The global kernel characterizes the duration of regenerative epochs and the sequence of regeneration conditions; the local kernel provides the transient probabilities of the process within a regenerative epoch, given the initial regeneration condition.

Despite the large domain of applicability of Markov regenerative processes and the availability of established techniques for their transient solution, an automated approach to the analysis of systems modeled by this class of stochastic processes presents considerable challenges. One must in fact come up with an appropriate definition of regeneration condition, provide an algorithm for the identification of regeneration points and corresponding regeneration conditions, and devise a way to compute the local and global kernels, at least numerically.

In this chapter, we provide a concept of regeneration condition which extends the usual notion from the literature on stochastic Petri nets. The time instants that we consider as regeneration points correspond to firings after which all GEN transitions are either disabled or they have been enabled for a *deterministic* time. This requirement is less strict than that of "newly enabled" GEN transitions (for which the enabling time is zero), and it allows to detect more regeneration points in STPNs including deterministic timers.

We discuss the properties of stochastic state classes reached after a regeneration point, and prove important results on the evolution of an STPN after a regeneration point. We provide an algorithm for the detection of regeneration points in the enumeration of stochastic state classes, and a solution for the numerical evaluation of the local and global kernels from the stochastic state classes enumerated within a regenerative epoch. When regeneration points are reached w.p.1 in a bounded number of transition firings, this solution allows to combine the algorithmic approach of stochastic state classes, computing closed-form PDFs of timers from the initial sampling PDFs, with integral equations leveraging the results of Markov renewal theory for the efficient evaluation of transient probabilities.

4.1 Regeneration conditions in STPNs

The definition of Markov regenerative process (Definition 2.10) allows for considerable freedom in the construction of a Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$ for the marking process $\{M(t),\ t \geq 0\}$. On the one hand, there must be a causal relation between the marking process and the regeneration points T_n and regeneration conditions X_n:

- The occurrence of the nth regeneration point T_n within time t, corresponding to the event $\{T_n \leq t\}$, must be completely determined by $\{M(u),\, 0 \leq u \leq t\}$ for all $t \geq 0$ and $n \in \mathbb{N}$.
- The regeneration condition X_n must be completely determined by $\{M(u),\, 0 \leq u \leq T_n\}$, for all $n \in \mathbb{N}$.

On the other hand, the choice of the state space R for the regeneration conditions X_n of the Markov renewal sequence $\{(X_n, T_n),\ n \in \mathbb{N}\}$ is arbitrary: richer state-space representations can add more information into regeneration conditions, allowing to summarize the past evolution in a larger class of situations after a firing, and thus to detect more regeneration points.

In STPNs, memory after a firing is due to persistent GEN times to fire, whose distribution depends on the time elapsed in previous states; in contrast, persistent EXP transitions always have independent times to fire with exponential distributions, thanks to the memoryless property of exponential random variables of Eq. (2.6). Therefore, a simple class of regeneration points is that of transition firings after which all GEN transitions are either disabled or newly enabled. In this case, the marking reached after the firing has sufficient information to determine the times to fire PDF: each enabled transition t has a time to fire independently distributed according to its sampling PDF f_t.

For this class of regeneration points, the set of regeneration conditions R is thus a subset of the reachable markings $\mathcal{M}$. In particular, for each regeneration point T_n, the regeneration condition is $X_n = M(T_n)$, and the independence of the future evolution of $M(t)$ from its past history given X_n corresponds to the Markov property. This class of regeneration points has been investigated, for example, in the seminal works of Ciardo et al. (1994), Choi et al. (1994) and Puliafito et al. (1998).

A more general class of regeneration points can be defined observing that, when a GEN transition t has been enabled for a deterministic amount of time d, its time to fire is always an independent random variable with PDF

$$f(x) = f_t(x+d)\left(\int_{\max\{d,EFT(t)\}}^{LFT(t)} f_t(u)\,du\right)^{-1} \tag{4.1}$$

on the support $[\max\{0, EFT(t)-d\}, LFT(t)-d]$. This condition can occur immediately after the firing of a deterministic transition enabled together with t, or enabled after (or before) a deterministic delay with respect to the enabling of t. When each GEN transition has been enabled for a deterministic time, the PDF of the times to fire $\vec{\tau}$ after the firing is uniquely identified by a vector $\vec{d}$ of enabling times, where d_i is the deterministic enabling time of the ith enabled GEN transition. Therefore, we consider, for the space of regeneration conditions R, pairs of the form $(m, \vec{d})$ where:

- The marking m identifies the set $E(m)$ of enabled transitions, and the probability density functions of EXP and IMM transitions (which are always equal to their sampling PDFs).
- The enabling times $\vec{d}$ identify the supports and probability density functions for the times to fire of enabled GEN transitions.

Regeneration points in which all the GEN transitions are newly enabled correspond to regeneration conditions of the form $(m, \vec{0})$, in which each GEN transition has been enabled for a deterministic time equal to zero.

Note that, in order to have a causal relation between the marking process $\{M(t),\ t \geq 0\}$ and the regeneration points T_n and regeneration conditions X_n (as required by Definition 2.10), the marking process must be extended to represent not only the current marking, but also the information required to detect regenerations. In the following, we avoid to do so explicitly by detecting regeneration points through the analysis of sequences of state transitions in the underlying general state-space Markov chain (which causes both the marking process and the Markov renewal sequence of the MRP).

4.2 Regenerative stochastic state classes

A stochastic state class reached through the transition firing associated with a regeneration point is said to be *regenerative.*

Definition 4.1 (Regenerative stochastic state class). The stochastic state class Σ_n in a sequence

$$\Sigma_0 \xrightarrow{\gamma_1, I_1, \mu_1} \Sigma_1 \xrightarrow{\gamma_2, I_2, \mu_2} \cdots \xrightarrow{\gamma_n, I_n, \mu_n} \Sigma_n$$

is *regenerative* if, for each GEN transition t enabled in Σ_n, the time elapsed from its enabling until the firing of γ_n is equal to some deterministic value $d \in \mathbb{R}_{\geqslant 0}$, which we call *enabling time* of t in Σ_n.

Regenerative stochastic state classes have the characteristic property that the random variables of the vector $\langle \tau_{age}, \vec{\tau} \rangle$ are all independent. Their joint PDF is in product form, and it can be constructed from the sampling PDFs of the STPN, given the PDF of τ_{age} and the deterministic enabling times of GEN transitions.

Lemma 4.1 (PDF and support of regenerative classes). *Let* $\Sigma = \langle m, D_{\langle \tau_{age}, \vec{\tau} \rangle}, f_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ *be a regenerative stochastic state class, and let* $\{t_1, \ldots, t_n\}$*,* $\{t_{n+1}, \ldots, t_m\}$ *and* $\{t_{m+1}, \ldots, t_l\}$ *be the sets of enabled GEN, EXP, and IMM transitions, respectively. Then, if* $\vec{d} = (d_1, \ldots, d_n) \in \mathbb{R}^n_{\geqslant 0}$ *gives the enabling time* d_i *of each enabled GEN transition* t_i*, the support* $D_{\langle \tau_{age}, \vec{\tau} \rangle}$ *and the probability density function* $f_{\langle \tau_{age}, \vec{\tau} \rangle}$ *of* $\langle \tau_{age}, \vec{\tau} \rangle$ *in* Σ *are given by*

$$
\begin{aligned}
D_{\langle \tau_{age}, \vec{\tau} \rangle} = D_{age} \times \prod_{i=1}^{n} & [\max\{0, EFT(t_i) - d_i\}, LFT(t_i) - d_i] \\
& \times \prod_{i=n+1}^{m} [0, +\infty) \times \prod_{i=m+1}^{l} [0, 0]
\end{aligned}
$$

and

$$
\begin{aligned}
f_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, \vec{x}) = f_{age}(x_{age}) \prod_{i=1}^{n} & \frac{f_{t_i}(x_i + d_i)}{\int_{\max\{d_i, EFT(t_i)\}}^{LFT(t_i)} f_{t_i}(u)\, du} \\
& \prod_{i=n+1}^{m} \lambda_{t_i} e^{-\lambda_{t_i} x_i} \prod_{i=m+1}^{l} \delta(x_i)
\end{aligned}
$$

respectively, for some PDF f_{age} *of* τ_{age} *with support* D_{age}*.*

The lemma allows to uniquely identify a regenerative class by

1. its marking m,
2. the vector $\vec{d}$ of enabling times for enabled GEN transitions, and
3. the PDF f_{age} and support D_{age} of τ_{age} immediately after the last firing, which corresponds to a regeneration point.

We represent regeneration conditions of the marking process by pairs $(m, \vec{d})$. This information is in fact sufficient to determine the stochastic evolution of the STPN, and thus of the marking process $\{M(t),\ t \geq 0\}$, after a regeneration point. To prove this result we consider an infinite *transient tree* encoding the succession relations

$$\Sigma_0 \xRightarrow{\gamma_1,[0,+\infty),\mu_1} \Sigma_1 \xRightarrow{\gamma_2,[0,+\infty),\mu_2} \cdots \xRightarrow{\gamma_n,[0,+\infty),\mu_n} \Sigma_n$$

among stochastic state classes for all the sequences $\gamma_1, \gamma_2, \ldots, \gamma_n$ of fired transitions and for all $n \in \mathbb{N}$. The stochastic state classes in the transient tree correspond to the state PDF of the STPN after each feasible sequence of transition firings, and thus characterize its transient evolution.

In the following, we omit the firing time interval I in the notation $\Sigma \xRightarrow{\gamma,I,\mu} \Sigma'$ whenever $I = [0, +\infty)$, which corresponds to the case that no constraint is imposed on the absolute firing time of γ.

Definition 4.2 (Transient tree). The *transient tree* from an initial class Σ_0 is a tuple $\langle N, E, n_0, \Sigma\rangle$ where:

- the set N is a countable set of nodes;
- $n_0 \in N$ is the root node;
- the function Σ associates each node $n \in N$ with a stochastic state class $\Sigma(n)$, with $\Sigma(n_0) := \Sigma_0$;
- the labeled edges $E \subseteq N \times T \times (0, 1] \times N$ represent the (unconstrained) successions of stochastic state classes associated with transition firings, so that $(n, t, \mu, n') \in E$ if and only if $\Sigma(n) \xRightarrow{t,\mu} \Sigma(n')$.

A node n associated with a regenerative class $\Sigma(n)$ is said to be *regenerative.* The following lemma guarantees that two regenerative nodes reached at different times, but associated with the same regeneration condition $(m, \vec{d})$, enable, in the subtrees subsequent to the regeneration points, the same firing sequences with the same probabilities.

Lemma 4.2. *Let n_k and n_h be two regenerative nodes associated with regenerative stochastic state classes $\Sigma(n_k)$ and $\Sigma(n_h)$ that have the same regeneration condition $(m, \vec{d})$ in a transient tree with initial stochastic state class Σ_0. Then, the succession sequences feasible from n_k and from n_h are the same, have the same probability, and end up in nodes associated with the same marking and PDF for times to fire $\vec{\tau}$ of enabled transitions.*

Proof. The proof runs by induction on the length of the succession sequences originating from the nodes n_k and n_h, and leverages the fact that the two classes have the same marginal distribution of times to fire $\vec{\tau}$, so that they will allow the same set of feasible behaviors with the same probabilities. According to Lemma 4.1, $\Sigma(n_k)$ and $\Sigma(n_h)$ have the same support and distribution for $\vec{\tau}$: the former condition implies that they accept the same set of feasible behaviors (sequences of firable transitions), and that equal succession sequences result in the same final markings and times to fire supports (due to the underlying non-deterministic model); the latter condition implies that the probabilities of these firing sequences are also the same, and that they end up in classes with the same times to fire PDF.

The next lemma completes the picture by focusing on the advancement of τ_{age} after a regenerative node, and it fully exploits the properties of the product-form PDF of regenerative stochastic state classes to show that the amounts of time elapsed before and after a regeneration point are independent random variables.

Lemma 4.3. *Let n_k be a regenerative node in a transient tree $\langle N, E, n_0, \Sigma \rangle$ from an initial class Σ_0, n_j be the node reached from n_k through the firing of transitions $\gamma_0, \ldots, \gamma_n$, and f^k_{age} and f^j_{age} be the marginal PDFs of the τ_{age} variable in $\Sigma(n_k)$ and $\Sigma(n_j)$, respectively. Then,*

$$f^j_{age}(x_{age}) = \int_{-\infty}^{\infty} f^k_{age}(u)\, \hat{f}^j_{age}(x_{age} - u)\, du \tag{4.2}$$

where $\hat{f}^j_{age}$ is the marginal PDF of τ_{age} for the node $\hat{n}_j$ reached through the same sequence of firings $\gamma_0, \ldots, \gamma_n$ in the transient tree $\langle N', E', n'_0, \Sigma' \rangle$ from an initial class $\Sigma'_0 = \langle m, D_{\langle \tau_{age}, \vec{\tau} \rangle}, f_{\langle \tau_{age}, \vec{\tau} \rangle} \rangle$ such that m is the same marking of $\Sigma(n_k)$,

$$f_{\langle \tau_{age}, \vec{\tau} \rangle}(x_{age}, \vec{x}) = \delta(x_{age}) f^k_{\vec{\tau}}(x_{age})$$

and

$$D_{\langle \tau_{age}, \vec{\tau} \rangle} = [0, 0] \times D^k_{\vec{\tau}},$$

so that $\vec{\tau}$ has the same marginal PDF and support, but $\tau_{age} = 0$.

Proof. According to Lemma 4.2, the time spent in the execution of the sequence $\gamma_0, \ldots, \gamma_n$ from $\Sigma(n_k)$ or from Σ'_0 is the same as it only depends on the marginal PDF of the times to fire $\vec{\tau}$, which is the same in the two stochastic state classes. Moreover, the PDF of this time is given by $\hat{f}^j_{age}$, since τ_{age} is equal to zero in $\Sigma(n'_0)$. Since n_k is regenerative, f^k_{age} is in product form with respect to the marginal PDF of times to fire (Lemma 4.1), and thus the evolution from n_k is independent of the time at which the node is reached; the age in n_j is then the sum of the independent random variables associated with the age in n_k and with the duration of the sequence $\gamma_0, \ldots, \gamma_n$, and it is thus distributed as the convolution of Eq. (4.2).

Note that in Lemma 4.2 the assumption that n_k and n_h are regenerative nodes is used only to guarantee that they have the same marginal PDF for the vector of times to fire $\vec{\tau}$. In Lemma 4.3, the assumption of regeneration is also used to guarantee that the PDFs of τ_{age} and $\vec{\tau}$ are in product form.

4.3 Detection of regeneration points

Regeneration points as defined in Section 4.1 can be detected on-the-fly during the computation of sequences of stochastic state classes. Our goal is to verify, after each firing, whether the enabled GEN transitions $\{t_1, t_2, \ldots, t_n\}$ have been enabled for a deterministic amount of time; in this case, the deterministic enabling times $\{d_1, d_2, \ldots, d_n\}$ form, together with the marking m reached after the firing, the regeneration condition $(m, \vec{d})$ associated with the regeneration point.

We observe that, immediately after the firing of a transition t, a newly enabled GEN transition t_i has been trivially enabled for a deterministic time $d_i = 0$. Whereas, if t_i is persistent, the time from its enabling until the firing of t is deterministic if and only if

- t is IMM or DET, and
- t_i was enabled together with t, or with a deterministic delay (or advance) with respect to the enabling of t.

In this case, the time from the enabling of t_i to the firing of t is equal to the deterministic time to fire associated with t in the STPN definition, reduced by the deterministic delay of the enabling of t_i with respect to that of t.

Therefore, to detect regeneration points we keep track of synchronizations between the enabling time of GEN transitions and DET or IMM transitions. To this end, each enabled DET or IMM transition t_d is associated with a set of GEN transitions $\textsc{Sync}(t_d)$ and with a function $\textsc{Enabling-Delay}(\,\cdot\,, t_d)\colon \textsc{Sync}(t_d) \to \mathbb{R}_{\geqslant 0}$ that are constructed so as to guarantee the following invariant: at the firing of t_d, the time elapsed since the newly enabling of t_i is deterministic if and only if $t_i \in \textsc{Sync}(t_d)$; in this case, the deterministic enabling time of t_i is given by $\textsc{Enabling-Delay}(t_i, t_d)$, and t_i is said to be *renewed*.

To assert and maintain the invariant, at the firing of each transition t:

- If the new stochastic state class includes some newly enabled DET or IMM transition t_d with deterministic time to fire x_d, every newly enabled or renewed GEN transition t_i is added to $\textsc{Sync}(t_d)$ with
$$\textsc{Enabling-Delay}(t_i, t_d) = x_i + x_d,$$
where x_i is the deterministic enabling time of t_i, equal to zero if t_i is newly enabled, or equal to $\textsc{Enabling-Delay}(t_i, t)$ if t_i was renewed at the firing of t.
- For every DET or IMM transition t_d that persisted after the firing of t, disabled GEN transitions are removed from $\textsc{Sync}(t_d)$, while persistent transitions t_i in $\textsc{Sync}(t_d)$ keep the same value of $\textsc{Enabling-Delay}(t_i, t_d)$.
- For every DET or IMM transition t_d disabled by the firing of t, $\textsc{Sync}(t_d)$ is emptied and $\textsc{Enabling-Delay}(\,\cdot\,, t_d)$ is discarded.

Example 4.1. In order to illustrate the concept of regeneration points, we highlight in Fig. 4.1 the regenerative classes and regeneration conditions for the sequence of transition firings fail, restart, arrival, service in the STPN of Fig. 3.1 from the initial marking 2*free operational*. In regenerative stochastic state classes Σ_0, Σ_3, Σ_4, the absolute time of the last event and all the times to fire of enabled transitions are independent random variables with product-form PDF; notably, the GEN variable ARRIVAL in Σ_4 is not newly enabled, but it does not carry memory given its deterministic enabling time 1.5.

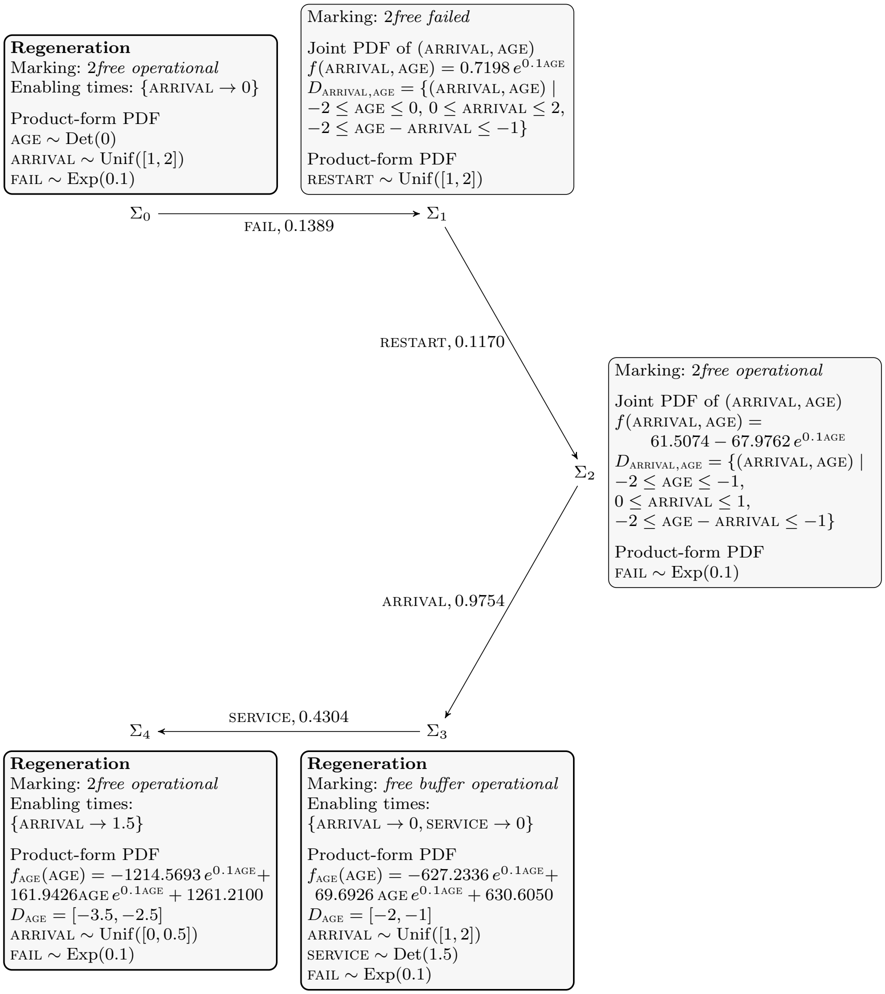

Regeneration
Marking: 2*free operational*
Enabling times: $\{\textsc{arrival} \to 0\}$
Product-form PDF
$\textsc{age} \sim \mathrm{Det}(0)$
$\textsc{arrival} \sim \mathrm{Unif}([1,2])$
$\textsc{fail} \sim \mathrm{Exp}(0.1)$
Marking: 2*free failed*
Joint PDF of $(\textsc{arrival}, \textsc{age})$
$f(\textsc{arrival}, \textsc{age}) = 0.7198\, e^{0.1\textsc{age}}$
$D_{\textsc{arrival},\textsc{age}} = \{(\textsc{arrival}, \textsc{age}) \mid -2 \leq \textsc{age} \leq 0,\ 0 \leq \textsc{arrival} \leq 2, -2 \leq \textsc{age} - \textsc{arrival} \leq -1\}$
Product-form PDF
$\textsc{restart} \sim \mathrm{Unif}([1,2])$
Σ_0
$\textsc{fail}, 0.1389$
Σ_1
$\textsc{restart}, 0.1170$
Marking: 2*free operational*
Joint PDF of $(\textsc{arrival}, \textsc{age})$
$f(\textsc{arrival}, \textsc{age}) = 61.5074 - 67.9762\, e^{0.1\textsc{age}}$
$D_{\textsc{arrival},\textsc{age}} = \{(\textsc{arrival}, \textsc{age}) \mid -2 \leq \textsc{age} \leq -1, 0 \leq \textsc{arrival} \leq 1, -2 \leq \textsc{age} - \textsc{arrival} \leq -1\}$
Product-form PDF
$\textsc{fail} \sim \mathrm{Exp}(0.1)$
Σ_2
$\textsc{arrival}, 0.9754$
Σ_4
$\textsc{service}, 0.4304$
Σ_3
Regeneration
Marking: 2*free operational*
Enabling times:
$\{\textsc{arrival} \to 1.5\}$
Product-form PDF
$f_{\textsc{age}}(\textsc{age}) = -1214.5693\, e^{0.1\textsc{age}} + 161.9426\textsc{age}\, e^{0.1\textsc{age}} + 1261.2100$
$D_{\textsc{age}} = [-3.5, -2.5]$
$\textsc{arrival} \sim \mathrm{Unif}([0, 0.5])$
$\textsc{fail} \sim \mathrm{Exp}(0.1)$
Regeneration
Marking: *free buffer operational*
Enabling times:
$\{\textsc{arrival} \to 0, \textsc{service} \to 0\}$
Product-form PDF
$f_{\textsc{age}}(\textsc{age}) = -627.2336\, e^{0.1\textsc{age}} + 69.6926\, \textsc{age}\, e^{0.1\textsc{age}} + 630.6050$
$D_{\textsc{age}} = [-2, -1]$
$\textsc{arrival} \sim \mathrm{Unif}([1,2])$
$\textsc{service} \sim \mathrm{Det}(1.5)$
$\textsc{fail} \sim \mathrm{Exp}(0.1)$

Fig. 4.1: Regeneration points and regeneration conditions for the sequence of events FAIL, RESTART, ARRIVAL, SERVICE in the queue of Fig. 3.1.

4.4 Computation of the kernels

The identification of regeneration points allows one to limit the enumeration of each transient tree to a single regenerative epoch, so that the leaves of the tree are regenerative stochastic state classes corresponding to regeneration points. The stochastic state classes from the root to a leaf node give the marking and times to fire PDF after each firing in a sequence that leads to some regeneration condition. As presented in Section 3.3, transient measures based on stochastic state classes can evaluate

1. the probability $p_{reach}(\Sigma_{leaf}, t)$ of reaching the last class Σ_{leaf} of the sequence within time t, and
2. the probability $p_{in}(\Sigma_{inner}, t)$ that, at time t, the STPN has fired all and only the transitions leading to some intermediate class Σ_{inner}.

If i and k are the regenerative conditions of the root and leaf node, respectively, the former measure contributes to the entry $G_{ik}(t)$ of the global kernel, which is the probability of reaching regeneration condition k from i within time t. On the other hand, the latter measure, when computed on intermediate classes with marking j, contributes to the entry $L_{ij}(t)$ of the local kernel, which is the transient probability of marking j at time t given the initial regeneration condition i.

By fixing a time bound t_{max} for the transient analysis of the marking process, these measures for all $t \leq t_{max}$ completely characterize regenerative epochs with initial regeneration condition i. For each regeneration condition k which is encountered in a leaf node and has not been previously discovered, a new transient tree can be enumerated from k limited to a regenerative epoch. Under the hypothesis that

- the number of distinct regeneration conditions is finite, and
- the number of classes enumerated between any two regeneration points is finite and bounded,

this procedure can be repeated until the transient trees from all the regeneration conditions—and limited to the first regenerative epoch—have been enumerated. Then, the marking process $\{M(t),\ t \geq 0\}$ is completely characterized by a finite set of transient trees, each associated with a different initial regeneration condition. The inner nodes of the trees give the transient probabilities within a regenerative epoch, while the leaf nodes give the probabilities of the next regeneration condition and the PDFs of epoch durations.

More formally, let R be the set of (reachable) regeneration conditions, and denote by $\textsc{Inner}(i)$ and $\textsc{Leaves}(i)$ the stochastic state classes associated with inner nodes and leaf nodes, respectively, in the transient tree enumerated from a regenerative stochastic state class with regeneration condition i (as defined in Lemma 4.1) and limited to the first regenerative epoch. Then, for each $t \leq t_{max}$, the (i, j)th entry of the local kernel $\mathbf{L}(t)$ is given by

$$L_{ij}(t) = \sum_{\substack{\Sigma \in \textsc{Inner}(i) \text{ s.t.} \\ \Sigma \text{ has marking } j}} p_{in}(\Sigma, t)$$

for all $i \in R$ and $j \in \mathcal{M}$, while the (i,k)th entry of the global kernel $\mathbf{G}(t)$ is given by

$$G_{ik}(t) = \sum_{\substack{\Sigma \in \textsc{Leaves}(i) \text{ s.t.} \\ \Sigma \text{ has reg. cond. } k}} p_{reach}(\Sigma, t)$$

for all $i, k \in R$.

The numerical evaluation of the local and global kernel enables the solution in the time domain of the generalized Markov renewal equations

$$\mathbf{P}(t) = \mathbf{L}(t) + \int_0^t d\mathbf{G}(u)\, \mathbf{P}(t-u)$$

where

$$P_{ij}(t) = P\{M(t) = j \mid X_0 = i\}$$

for all $t \leq t_{max}$, regeneration conditions $i \in R$, and markings $j \in \mathcal{M}$. By discretizing the time domain $[0, t_{max}]$ in $n+1$ equidistant points $t_0, t_1, \ldots, t_n$ with $t_m = (t_{max}/n)\, m$ for $m = 0, 1, \ldots, n$, Newton–Cotes formulas define the system of linear equations

$$\mathbf{P}(t_m) = \mathbf{L}(t_m) + \mathbf{G}(0)\, \mathbf{P}(t_m) + \sum_{u=1}^{m} \left(\mathbf{G}(t_u) - \mathbf{G}(t_{u-1})\right)\, \mathbf{P}(t_{m-u})$$

in the unknowns $P_{ij}(t_m)$ for all $i \in R$, $j \in \mathcal{M}$ and $m = 0, 1, \ldots, n$. The system can be solved by forward substitution if the unknowns are computed in the order $\mathbf{P}(t_0), \mathbf{P}(t_1), \ldots, \mathbf{P}(t_n)$ as

$$\mathbf{P}(t_m) = (I - \mathbf{G}(0))^{-1} \left(\mathbf{L}(t_m) + \sum_{u=1}^{m} \left(\mathbf{G}(t_u) - \mathbf{G}(t_{u-1})\right)\, \mathbf{P}(t_{m-u}) \right) \quad (4.3)$$

for $m = 0, 1, \ldots, n$.

Overall, the global and local kernels need to be evaluated at n time instants. Leveraging the transient measures presented in Section 3.3, each evaluation requires to recompute the stochastic state classes of the leaves and inner nodes of the transient trees enumerated from all regeneration conditions in R. If C is the number of stochastic state classes in all of the $|R|$ transient trees, the number of classes to enumerate is $C\left(\frac{t_{max}}{h} + 1\right)$, where $h = t_{max}/n$ is the step size used in the time discretization.

Example 4.2 (Transient analysis of a G/D/1/2/2 queue). The STPN model of the G/D/1/2/2 queue of Fig. 3.1 encounters 5 distinct regeneration conditions:

- $r_0 = \big(2\mathit{free\ operational}, \{\textsc{arrival} \to 0\}\big)$;
- $r_1 = \big(2\mathit{buffer\ operational}, \{\textsc{service} \to 0\}\big)$;
- $r_2 = \big(2\mathit{buffer\ failed}, \{\textsc{restart} \to 0\}\big)$;
- $r_3 = \big(\mathit{free\ buffer\ operational}, \{\textsc{arrival} \to 0, \textsc{service} \to 0\}\big)$;
- $r_4 = \big(2\mathit{free\ operational}, \{\textsc{arrival} \to 1.5\}\big)$.

The transient trees enumerated from these regenerations and limited to the first regenerative epoch include 69, 3, 2, 15 and 20 stochastic state classes, respectively. The set $\mathcal{M}$ of reachable markings includes 6 markings:

- $m_0 = 2\mathit{free\ operational}$;
- $m_1 = 2\mathit{buffer\ operational}$;
- $m_2 = 2\mathit{buffer\ failed}$;
- $m_3 = \mathit{free\ buffer\ operational}$;
- $m_4 = 2\mathit{free\ failed}$;
- $m_5 = \mathit{free\ buffer\ failed}$.

Note that marking m_0 is encountered in two distinct regeneration conditions (r_0 and r_4), while markings m_4 and m_5 are never encountered at a regeneration point.

For each $t \geq 0$, the global kernel $\mathbf{G}$ is a 5×5 matrix, while the local kernel $\mathbf{L}$ is a 5×6 matrix. From the supports of stochastic state classes, we can verify that the measures of Eq. (3.6) and Eq. (3.8) contributing to the elements of $\mathbf{G}$ and $\mathbf{L}$ converge to a constant value after time 8. As a consequence, for any time bound $t_{max} > 8$, a transient analysis with step $h = 0.1$ requires the enumeration of $109\,(8/0.1 + 1) = 8829$ stochastic state classes. Fig. 4.2 reports the transient probabilities for four conditions in the G/D/1/2/2 queue from the initial regeneration condition $r_0 = \big(2\mathit{free\ operational}, \{\textsc{arrival} \to 0\}\big)$:

- "$\mathit{buffer} = 0$", which corresponds to $P_{r_0,m_0}(t) + P_{r_0,m_4}(t)$,
- "$\mathit{buffer} = 1$", which corresponds to $P_{r_0,m_3}(t) + P_{r_0,m_5}(t)$,
- "$\mathit{buffer} = 2$", which corresponds to $P_{r_0,m_1}(t) + P_{r_0,m_2}(t)$, and
- "$\mathit{operational} = 1$", which corresponds to $P_{r_0,m_0}(t) + P_{r_0,m_1}(t) + P_{r_0,m_3}(t)$.

Each transition probability $P_{ij}(t)$ with $i \in R$ and $j \in \mathcal{M}$ was evaluated for $t = 0, 0.1, 0.2, \ldots, 20.0$ using Eq. (4.3).

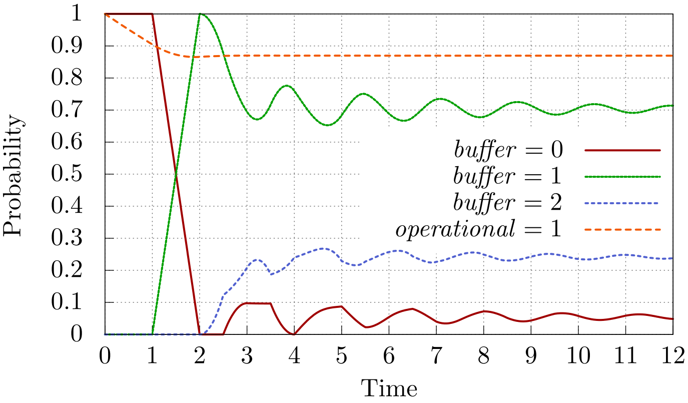


Fig. 4.2: Transient probabilities for the buffer occupation and server status in the G/D/1/2/2 queue with server breakdowns.

Chapter 5
Verification of an Interval Until Operator

Probabilistic model checking is a formal method for the analysis of quantitative properties of stochastic systems: given a probabilistic model of the system, quantitative properties expressed in a formal logic are checked automatically. This formal approach is analogous to model checking of temporal logics in transition systems: both the system model and the required property are specified formally, and the model checking algorithm can prove whether the property is satisfied or not. In addition, probabilistic model checking can also provide a probability value of the property satisfaction, and reward structures can enrich probabilistic temporal logics with "costs" or "rewards" associated with selected state transitions, or accumulated during sojourns with state-specific rates. Not only the system is modeled with a high-level formalism such as stochastic Petri nets, queueing networks, stochastic process algebras or stochastic activity networks, which avoid complex and error-prone model definitions in the state-space (an introduction to the latter formalisms can be found in Clark et al. (2007) and Sanders and Meyer (2001), for example); also the performance measures of interest can be expressed in a well-defined language and evaluated automatically, enabling an early assessment of design choices and regression verification during model refinement and evolution.

As reported in Grunske (2008), empirical evidence indicates that most probabilistic requirements occurring in the industrial practice can be formulated through a limited set of property specification patterns: among these, the fundamental patterns for the specification of transient properties are based on the probabilistic *interval until operator* $P_{\sim p}[\,\varphi_1 \mathcal{U}^{[\alpha,\beta]} \varphi_2\,]$, which imposes an upper or lower bound p on the probability that the model will be in a "goal" state satisfying φ_2 at some time in the interval $[\alpha, \beta]$ after visiting only a subset of "safe" states that satisfy φ_1.

When the underlying stochastic process of the model is a continuous-time Markov chain (as in stochastic time Petri nets with only EXP or IMM transitions), this problem can be reduced to transient analysis through the approach of Baier et al. (2003). In particular, transient analysis can be performed independently before and after α on instances of the original CTMC modified so as to turn goal states after α and illegal states into absorbing ones. This solution is justified by the fact that the system is memoryless and α is always a regeneration point. In contrast, when the underlying stochastic process can accumulate memory over time, as in the case of semi-Markov and Markov regenerative processes, the system evolution before and after α cannot be analyzed independently.

The problem could still be reduced to transient analysis by adding a deterministic timer in parallel to the model so as to represent the elapse of α and record the corresponding event in the logic state (the marking of the STPN). Unfortunately, this solution crucially affects regenerative transient analysis: it is now the deterministic timer that carries memory, destroying all regeneration points of the model before time α.

In this chapter, we present a solution based on a renewal argument specific to the interval until operator, which results in a bivariate formulation of Markov renewal equations. We provide algorithms for the evaluation of the parameters of these integral equations through the enumeration of stochastic state classes limited to the first regenerative epoch (as in regenerative transient analysis). The solution allows the verification of Boolean combinations of interval until operators on stochastic time Petri nets in which multiple GEN transitions can be started or stopped independently, but regeneration points are always encountered w.p.1 after a bounded number of firings. The repetitive structure of the underlying Markov regenerative process is exploited also before the lower bound α, providing crucial benefits for large time bounds.

A case study is also presented through the probabilistic formulation of Fischer's mutual exclusion protocol, a well-known real-time verification benchmark.

5.1 Probability space and cylinder sets

In the formulation of the probabilistic model checking problem, we will need to refer to the probability measure of selected sets of execution paths of an STPN model. We thus formalize the concept by defining the probability space $(\Omega_{m_0}, \mathcal{F}_{m_0}, Pr_{m_0,f_{\vec{\tau}_0}})$ induced by the semantics of STPNs for a given initial marking m_0 and initial times to fire PDF $f_{\vec{\tau}_0}$.

The outcomes Ω_{m_0} of the probability space are all (finite or infinite) paths

$$\omega = s_0 \xrightarrow{\gamma_1} s_1 \xrightarrow{\gamma_2} s_2 \xrightarrow{\gamma_3} \cdots$$

originating from a state $s_0 = \langle m_0, \tau_0 \rangle$ with marking m_0, where $\gamma_i \in T$ is the ith transition fired along the path and $s_i = \langle m_i, \vec{\tau}_i \rangle$ is the state reached after the firing of γ_i.

To identify a σ-algebra $\mathcal{F}_{m_0}$ of events over the paths Ω_{m_0}, let $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n)$ denote the cylinder set including all the paths with initial marking m_0 that fire the sequence of transitions $\gamma_1, \gamma_2, \ldots, \gamma_n$ at absolute times contained in the intervals $I_1, I_2, \ldots, I_n$, respectively:

$$\begin{aligned} C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n) := \{\, \omega \in \Omega_{m_0} \mid |\omega| \geq n \text{ and } \\ \forall\, 0 < k \leq n.(\omega[k] = \gamma_k \text{ and } T(k, \omega) \in I_k) \,\} \end{aligned}$$

where $|\omega|$ is the number of firings in ω (which is finite if $s_{|\omega|}$ is an *absorbing* state), $\omega[k] = \gamma_k$ for all $0 < k \leq |\omega|$ is the kth transition fired along ω, and $T(k, \omega)$ is the absolute time of the kth firing in ω, for all k:

$$T(k, \omega) := \begin{cases} \sum_{i=0}^{k-1} \min_{t \in E(m_i)} \vec{\tau}_i(t) & \text{if } k \leq |\omega|, \\ +\infty & \text{if } k > |\omega|. \end{cases}$$

Note that, in contrast with the usual definition of cylinder sets for CTMCs, constraints refer to absolute firing times rather than sojourn times in visited states. This formulation allows a simpler treatment of the dependence among subsequent sojourn times in models with underlying stochastic processes beyond the limits of semi-Markov processes.

The set of events $\mathcal{F}_{m_0}$ is defined as the smallest σ-algebra on Ω_{m_0} that contains all the cylinder sets $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n)$ for $n \in \mathbb{N}$, $\gamma_1, \gamma_2, \cdots, \gamma_n$ ranging over all the sequences of n transitions in T and $I_1, I_2, \ldots, I_n$ ranging over all the sequences of n non-empty real intervals of the form $[a, \infty)$ or $[a, b]$ with $a, b \in \mathbb{Q}$.

Proposition 5.1. *$\mathcal{F}_{m_0}$ is countable and closed with respect to intersection and complement operations.*

Proof. The elements of $\mathcal{F}_{m_0}$ are uniquely identified by finite strings alternating transitions and firing intervals $[a, \infty)$ or $[a, b]$ with $a, b \in \mathbb{Q}$: pairs of rational numbers and finite strings from a finite alphabet are countable sets, and thus $\mathcal{F}_{m_0}$ is also countable. The intersection of two cylinder sets $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n)$ and $C(m_0, \gamma'_1, I'_1, \gamma'_2, I'_2, \ldots, \gamma'_m, I'_m)$ with $n \leq m$ is non-empty only if $\gamma_i = \gamma'_i$ for $i = 1, \ldots, n$, and it corresponds to the cylinder set

$$C(m_0, \gamma_1, I_1 \cap I'_1, \gamma_2, I_2 \cap I'_2, \ldots, \gamma_n, I_n \cap I'_n, \gamma'_{n+1}, I_{n+1}, \ldots, \gamma'_m, I'_m),$$

which belongs to $\mathcal{F}_{m_0}$.

Finally, the complement of a cylinder set $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n)$ corresponds to the finite union of all cylinder sets of the form $C(m_0, \gamma'_1, I'_1, \gamma'_2, I'_2, \ldots, \gamma'_n, I'_n)$ such that either

- there exists $i \leq n$ such that $\gamma_i \neq \gamma'_i$ and $I'_i = [0, \infty)$, or
- for all $i \leq n$, $\gamma_i = \gamma'_i$ and $I'_i = [0, \inf I_i]$ or $I'_i = [\sup I_i, \infty)$.

The probability measure $Pr_{m_0,f_{\vec{\tau}_0}}$ on cylinder sets can be evaluated through the transient measures on stochastic state classes described in Section 3.3. Moreover, when GEN timers of the model are distributed according to piecewise expolynomial PDFs (on bounded or unbounded supports), these measures can be computed numerically using the Sirio package of the ORIS tool described in Carnevali et al. (2011).

Proposition 5.2 (Measure of cylinder sets). *The probability measure of a cylinder set $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n)$ is equal to the product $\prod_{i=1}^{n} \mu_i$ of succession probabilities for the sequence of stochastic state classes*

$$\Sigma_0 \xrightarrow{\gamma_1, I_1, \mu_1} \Sigma_1 \xrightarrow{\gamma_2, I_2, \mu_2} \cdots \xrightarrow{\gamma_n, I_n, \mu_n} \Sigma_n,$$

or equal to 0 if the sequence is not defined (i.e., $\exists i \leq n. \mu_i = 0$).

Proof. The event $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n)$ for $n \in \mathbb{N}$ can be expressed as $E_0 \cap E_1 \cap \cdots \cap E_n$, where $E_0 = \Omega_{m_0}$ and, for each $i > 0$,

$$E_i = \{\, \omega \in \Omega_{m_0} \mid |\omega| \geq i, \omega[i] = \gamma_i \text{ and } T(i, \omega) \in I_i \,\}$$

is the constrained set of paths imposing an absolute time only on the ith transition; thus, by the chain rule,

$$\begin{aligned} Pr_{m_0,f_{\vec{\tau}_0}}\{E_0 \cap E_1 \cap \cdots \cap E_n\} = {} & Pr_{m_0,f_{\vec{\tau}_0}}\{E_0\}\ Pr_{m_0,f_{\vec{\tau}_0}}\{E_1 \mid E_0\} \\ & Pr_{m_0,f_{\vec{\tau}_0}}\{E_2 \mid E_0, E_1\} \cdots Pr_{m_0,f_{\vec{\tau}_0}}\{E_n \mid E_0, E_1, \ldots, E_{n-1}\} \end{aligned}$$

where $Pr_{m_0,f_{\vec{\tau}_0}}$ is the probability measure over STPN paths. By induction on the definition of succession relation, for all $i \leq n$, the class Σ_i in

$$\Sigma_0 \xrightarrow{\gamma_1, I_1, \mu_1} \Sigma_1 \xrightarrow{\gamma_2, I_2, \mu_2} \cdots \xrightarrow{\gamma_n, I_n, \mu_n} \Sigma_n$$

represents the joint PDF of the absolute time and current state given the events $E_0, E_1, \ldots, E_i$, and $\mu_i = Pr_{m_0,f_{\vec{\tau}_0}}\{E_i \mid E_0, E_1, \ldots, E_{i-1}\}$. Then $Pr_{m_0,f_{\vec{\tau}_0}}\{E_0 \cap E_1 \cap \cdots \cap E_n\} = \prod_{i=1}^{n} \mu_i$ if $\mu_i > 0$ for all $i \leq n$; if some event E_i has null probability given $E_0, E_1, \ldots, E_{i-1}$ (i.e., Σ_{i-1} has no successor through γ_i at some time in I_i and thus $\mu_i = 0$), the measure of the cylinder set is zero.

5.2 Probabilistic temporal logic

We specify quantitative properties of STPNs with a probabilistic temporal logic based on an *interval until operator* with predicates over the markings of the net. The logic allows to express bounds on the probability that the marking of the STPN satisfies a goal predicate φ_2 at some time in the interval

$[\alpha,\beta]$ without violating a safety predicate φ_1. The syntax of the logic is

$$\psi ::= \textsc{true} \mid \textsc{ap} \mid \neg\psi \mid \psi \wedge \psi \mid P_{\sim p}[\,\varphi\,\mathcal{U}^{[\alpha,\beta]}\varphi\,]$$
$$\varphi ::= \textsc{true} \mid \textsc{ap} \mid \neg\varphi \mid \varphi \wedge \varphi$$

where $\sim\, \in \{<,>\}$, $p \in [0,1]$ is a probability value, $\alpha,\beta \in \mathbb{Q}_{\geqslant 0}$, and the atomic predicates are defined as $\textsc{ap} ::= g \bowtie x$ where $\bowtie\, \in \{<,\leq,=,\neq,\geq,>\}$, $x \in \mathbb{R}$ and $g\colon \mathbb{N}^P \to \mathbb{R}$ is a real-valued function over markings (e.g., "*free* > 1" for the net of Fig. 3.1).

Note that the interval until operator $P_{\sim p}[\,\varphi_1\,\mathcal{U}^{[\alpha,\beta]}\varphi_2\,]$ imposes also a lower bound α on the time for the satisfaction of φ_2, in contrast with the *bounded* until operator $P_{\sim p}[\,\varphi_1\,\mathcal{U}^{\leq\beta}\varphi_2\,]$ solved in Infante-López et al. (2001) for semi-Markov processes and in Martinez and Haverkort (2006) for Markov regenerative processes under enabling restriction.

As in Bryans et al. (2003), the logic allows the Boolean composition of interval until operators, each evaluated from a random initial state $s_0 = \langle m_0, \vec{\tau}_0\rangle$ in which m_0 is a marking and $\vec{\tau}_0$ is a times to fire vector sampled according to $f_{\vec{\tau}_0}$. Without loss of generality, we assume that all enabled transitions $E(m_0) = \{t_1, t_2, \ldots, t_n\}$ are newly enabled in the initial state, and thus $\vec{\tau}_0$ is distributed according to $f_{\vec{\tau}_0}(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} f_{t_i}(x_i)$.

Definition 5.1 (Logic semantics). Given a stochastic time Petri net $\langle P, T, A^-, A^+, A^\circ, EFT, LFT, f, w\rangle$ with initial marking m_0 and times to fire $\vec{\tau}_0$ initially distributed according to $f_{\vec{\tau}_0}$, the relations $\langle m_0, f_{\vec{\tau}_0}\rangle \models \varphi$ and $\langle m_0, f_{\vec{\tau}_0}\rangle \models \psi$ are defined inductively by

$$\begin{aligned}
\langle m_0, f_{\vec{\tau}_0}\rangle \models \textsc{true} &\iff \text{always satisfied}\\
\langle m_0, f_{\vec{\tau}_0}\rangle \models \textsc{ap} &\iff \textsc{ap} \text{ is satisfied by } m_0\\
\langle m_0, f_{\vec{\tau}_0}\rangle \models \neg\varphi &\iff \langle m_0, f_{\vec{\tau}_0}\rangle \not\models \varphi\\
\langle m_0, f_{\vec{\tau}_0}\rangle \models \neg\psi &\iff \langle m_0, f_{\vec{\tau}_0}\rangle \not\models \psi\\
\langle m_0, f_{\vec{\tau}_0}\rangle \models \varphi_1 \wedge \varphi_2 &\iff \langle m_0, f_{\vec{\tau}_0}\rangle \models \varphi_1 \wedge \langle m_0, f_{\vec{\tau}_0}\rangle \models \varphi_2\\
\langle m_0, f_{\vec{\tau}_0}\rangle \models \psi_1 \wedge \psi_2 &\iff \langle m_0, f_{\vec{\tau}_0}\rangle \models \psi_1 \wedge \langle m_0, f_{\vec{\tau}_0}\rangle \models \psi_2
\end{aligned}$$

and

$$\langle m_0, f_{\vec{\tau}_0}\rangle \models P_{\sim p}[\,\varphi_1\,\mathcal{U}^{[\alpha,\beta]}\varphi_2\,] \iff$$
$$Pr_{m_0, f_{\vec{\tau}_0}}\{\omega \in \Omega_{m_0} \mid \omega \models \varphi_1\,\mathcal{U}^{[\alpha,\beta]}\varphi_2\} \sim p \qquad (5.1)$$

where, for any path $\omega = s_0 \xrightarrow{\gamma_1} s_1 \xrightarrow{\gamma_2} \cdots$ with $s_i = \langle m_i, \vec{\tau}_i\rangle$, $m_i \in \mathbb{N}^P$ and $\vec{\tau}_i \in \mathbb{R}_{\geqslant 0}^{E(m_i)}$ for all i,

$$\begin{aligned}\omega \models \varphi_1 \mathcal{U}^{[\alpha,\beta]} \varphi_2 \iff \exists n \leq |\omega| \text{ such that } m_n \models \varphi_2 \\ \wedge \big(\forall k < n.(m_k \models \varphi_1)\big) \wedge \big(T(n,\omega) \in [\alpha,\beta] \\ \vee \ (T(n,\omega) < \alpha \wedge T(n+1,\omega) \geq \alpha \wedge m_n \models \varphi_1)\big)\,. \end{aligned} \tag{5.2}$$

Example 5.1 (G/D/1/2/2 queue). In the G/D/1/2/2 queue of Fig. 3.1, the property

$$P_{<0.4}[(\mathit{buffer} < 2)\,\mathcal{U}^{[0,7]}(\mathit{failed} = 1)] \\ \wedge P_{<0.2}[(\mathit{buffer} < 2)\,\mathcal{U}^{[2.5,7]}(\mathit{failed} = 1)]$$

is satisfied if the probability of the server being down without ever reaching its full capacity is lower than 0.4 in the interval $[0, 7]$ and lower than 0.2 in the interval $[2.5, 7]$.

The following proposition shows that, for every pair of marking predicates φ_1, φ_2, and $\alpha, \beta \in \mathbb{Q}_{\geqslant 0}$, the set of paths satisfying the interval until operator is an event of $\mathcal{F}_{m_0}$. Concretely, this means that the value of $Pr_{m_0, f_{\vec{\tau}_0}}\{\omega \in \Omega_{m_0} \mid \omega \models \varphi_1 \mathcal{U}^{[\alpha,\beta]} \varphi_2\}$ is well-defined and the semantics of Definition 5.1 can be computed.

Proposition 5.3. *For each $\varphi_1, \varphi_2 \in \mathbb{N}^P$ and $\alpha, \beta \in \mathbb{Q}_{\geqslant 0}$, the set $\{\,\omega \in \Omega_{m_0} \mid \omega \models \varphi_1 \mathcal{U}^{[\alpha,\beta]} \varphi_2\,\}$ of paths satisfying the corresponding interval until operator is a countable union of elements of $\mathcal{F}_{m_0}$.*

Proof. The cylinder sets that end on a φ_2-marking reached only through φ_1-markings are countable. Each cylinder set is in fact uniquely identified by the sequence of transitions $\gamma_1, \gamma_2, \ldots, \gamma_n$ fired from m_0, which are strings on a finite alphabet. For each cylinder set $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n)$ that ends on a φ_2-marking only through φ_1-markings, we consider (1) the cylinder set imposing only an absolute time $I_n = [\alpha, \beta]$ for the nth transition, and (2) if the marking reached after γ_n satisfies also φ_1, the cylinder sets $C(m_0, \gamma_1, I_1, \gamma_2, I_2, \ldots, \gamma_n, I_n, \gamma_{n+1}, I_{n+1})$ for each $\gamma_{n+1} \in T$ imposing a bound $I_n = [0, \alpha)$ for the firing of γ_n and a bound $I_{n+1} = [\alpha, \infty)$ for the firing of γ_{n+1}. The countable union of these cylinder sets is an event of $\mathcal{F}_{m_0}$ including all and only the successful paths.

5.3 Markov renewal equations for the interval until operator

Given the predicates φ_1 and φ_2, a real interval $[\alpha, \beta]$ with $\alpha, \beta \in \mathbb{Q}_{\geqslant 0}$, and a regeneration condition $i = (m, \vec{d})$, we define

$$\Omega_i(\alpha, \beta) := \{\,\omega \in \Omega_m \mid \omega \models \varphi_1 \mathcal{U}^{[\alpha,\beta]} \varphi_2\,\}$$

to be the set of paths that start from the marking of the regeneration condition i and satisfy the until operator, and we denote by $p_i(\alpha,\beta) := Pr_i\{\,\Omega_m(\alpha,\beta)\,\}$ its probability measure in the probability space $(\Omega_m, \mathcal{F}_m, Pr_{m,\vec{d}})$ for the paths with initial marking m when times to fire are initially distributed according to a PDF $f_{\vec{\tau}_0}$ with the product-form described in Lemma 4.1 for enabling times $\vec{d}$ and τ_{age} equal to zero (i.e., with $f_{age}(x_{age}) = \delta(x_{age})$ and $D_{age} = [0,0]$).

For each path $\omega = s_0 \xrightarrow{\gamma_1} s_1 \xrightarrow{\gamma_2} \cdots$ in $\Omega_i(\alpha,\beta)$, with $s_n = \langle m_n, \vec{\tau}_n\rangle$, $m_n \in N^P$ and $\vec{\tau}_n \in \mathbb{R}_{\geqslant 0}^{E(m_n)}$, we indicate as

$$\textsc{reg}(\omega) := \min\{n \in \mathbb{N} \mid s_{n-1} \xrightarrow{\gamma_n} s_n \text{ is a regeneration }\}$$

the index of the first regeneration along the path, and we indicate as

$$\begin{aligned}\textsc{ok}(\omega) := \min\{n \in \mathbb{N} \mid\ & n \leq |\omega| \,\wedge\, (m_n \models \varphi_2) \,\wedge \\ & \big(\forall k < n.(m_k \models \varphi_1)\big) \,\wedge\, \big(T(n,\omega) \in [\alpha,\beta] \,\vee \\ & (T(n,\omega) < \alpha \,\wedge\, T(n+1,\omega) \geq \alpha \,\wedge\, m_n \models \varphi_1)\big)\,\}\end{aligned}$$

the index of the first conclusive state satisfying the until operator. Moreover, we indicate with $t_{\textsc{reg}}(\omega) := T(\textsc{reg}(\omega),\omega)$ the time of the first regeneration in ω, and with $(m_{\textsc{reg}(\omega)}, \vec{d}_{\textsc{reg}(\omega)})$ the corresponding regeneration condition.

The probability $p_i(\alpha,\beta)$ can then be decomposed so as to separately account for paths in $\Omega_i(\alpha,\beta)$ that satisfy the until operator under different timings of the first regeneration. To this end, we distinguish paths that satisfy the until operator before reaching a regeneration from those that encounter the first regeneration before α, or between α and β, and then satisfy the until operator:

$$\begin{aligned}\Omega_i^L(\alpha,\beta) &:= \{\,\omega \in \Omega_i(\alpha,\beta) \mid \textsc{ok}(\omega) < \textsc{reg}(\omega)\,\},\\ \Omega_i^G(\alpha,\beta) &:= \{\,\omega \in \Omega_i(\alpha,\beta) \mid \textsc{ok}(\omega) \geq \textsc{reg}(\omega) \wedge t_{\textsc{reg}}(\omega) < \alpha\,\}, \text{ and}\\ \Omega_i^H(\alpha,\beta) &:= \{\,\omega \in \Omega_i(\alpha,\beta) \mid \textsc{ok}(\omega) \geq \textsc{reg}(\omega) \wedge t_{\textsc{reg}}(\omega) \in [\alpha,\beta]\,\}\,.\end{aligned}$$

Given these sets of successful paths, the following result holds.

Proposition 5.4. *For any regeneration condition i and $\alpha,\beta \in \mathbb{Q}_{\geqslant 0}$, we have $p_i(\alpha,\beta) = Pr_i\{\Omega_i^L(\alpha,\beta)\} + Pr_i\{\Omega_i^G(\alpha,\beta)\} + Pr_i\{\Omega_i^H(\alpha,\beta)\}$.*

Proof. The sets $\Omega_i^L(\alpha,\beta)$, $\Omega_i^G(\alpha,\beta)$, $\Omega_i^H(\alpha,\beta)$ comprise a partition of the set of paths $\Omega_i(\alpha,\beta)$: on the one hand, they are clearly disjoint; on the other hand, to prove that they cover $\Omega_i(\alpha,\beta)$ it is sufficient to consider that $\forall\omega \in \Omega_i(\alpha,\beta)$, $t_{\textsc{reg}}(\omega) > \beta$ implies $\textsc{ok}(\omega) < \textsc{reg}(\omega)$.

The probability measure of the three sets $\Omega_i^L(\alpha,\beta)$, $\Omega_i^G(\alpha,\beta)$ and $\Omega_i^H(\alpha,\beta)$ can be expressed in terms of three *kernels* that depend on the behavior of the stochastic process within the first epoch of regeneration.

The measure of $\Omega_i^L(\alpha,\beta)$ is directly defined as the *local kernel* $L_i^{\varphi_1,\varphi_2}(\alpha,\beta) := Pr\{\,\Omega_i^L(\alpha,\beta)\,\}$, which evaluates the probability measure of paths that satisfy the until operator before reaching a regeneration. In contrast, the measures of $\Omega_i^G(\alpha,\beta)$ and $\Omega_i^H(\alpha,\beta)$ are not limited to a regenerative epoch and require the following propositions.

Proposition 5.5. *The measure $Pr_i\{\,\Omega_i^G(\alpha,\beta)\,\}$ is equal to*

$$\sum_k \int_{x\in[0,\alpha)} dG_{ik}^{\varphi_1}(x)\, p_k(\alpha - x, \beta - x) \tag{5.3}$$

where $k = (m, \vec{d})$ ranges over all reachable regeneration conditions and the global kernel *$G_{ik}^{\varphi_1}(x)$ is defined as*

$$\begin{aligned} G_{ik}^{\varphi_1}(x) := Pr_i\{\,\omega \in \Omega_i \mid t_{\text{REG}}(\omega) \le x \\ \wedge\ (m_{\text{REG}(\omega)}, \vec{d}_{\text{REG}(\omega)}) = k\ \wedge\ (\forall j < \text{REG}(\omega)).(m_j \models \varphi_1)\,\}\,. \end{aligned} \tag{5.4}$$

Proof. For each $\omega \in \Omega_i^G(\alpha,\beta)$, it must be $\text{OK}(\omega) \ge \text{REG}(\omega)$ and $t_{\text{REG}}(\omega) < \alpha$ (i.e., ω encounters a goal state after reaching a regeneration point before time α). According to Lemmas 4.2 and 4.3, the process evolution after the regeneration point depends only on the regeneration condition $(m_{\text{REG}(\omega)}, \vec{d}_{\text{REG}(\omega)})$, and the remaining time for satisfying the until operator is reduced by $t_{\text{REG}}(\omega)$; since the only condition required by Eq. (5.2) for states $\langle m_j, \vec{\tau}_j\rangle$ with $j < \text{OK}(\omega)$ is $m_j \models \varphi_1$, and $\text{OK}(\omega) \ge \text{REG}(\omega)$ for $\omega \in \Omega_i^G(\alpha,\beta)$, the measure $Pr_i\{\,\Omega_i^G(\alpha,\beta)\,\}$ is equal to

$$\int_{X(\alpha)} p_{(m_{\text{REG}(\omega)}, \vec{d}_{\text{REG}(\omega)})}\big(\alpha - t_{\text{REG}}(\omega), \beta - t_{\text{REG}}(\omega)\big)\, dPr_i(\omega) \tag{5.5}$$

where $X(\alpha) := \{\omega \in \Omega_i \mid t_{\text{REG}}(\omega) < \alpha \wedge (\forall j < \text{REG}(\omega)).(m_j \models \varphi_1)\,\}$. In Eq. (5.5), the measure of each path reaching its first regeneration point before α without violating φ_1 is multiplied by the probability that the until operator will be satisfied from the regeneration in the remaining time. By conditioning on all reachable regeneration conditions $(m_{\text{REG}(\omega)}, \vec{d}_{\text{REG}(\omega)}) = k$ and times $t_{\text{REG}}(\omega) = x < \alpha$ of the first regeneration, we obtain Eq. (5.3), where the global kernel represents the probability of reaching a regeneration within time x with regeneration condition k, always satisfying φ_1 in previous states.

Proposition 5.6. *The measure $Pr_i\{\,\Omega_i^H(\alpha,\beta)\,\}$ is equal to*

$$\sum_k \int_{x\in[\alpha,\beta]} dH_{ik}^{\varphi_1,\varphi_2}(\alpha, x)\, p_k(0, \beta - x) \tag{5.6}$$

where $k = (m, \vec{d})$ ranges over all reachable regeneration conditions and the conditional global kernel *$H_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$ is defined as*

$$H_{ik}^{\varphi_1,\varphi_2}(\alpha,x) := Pr_i\{\,\omega \in \Omega_i \mid t_{\text{REG}}(\omega) \in [\alpha,x] \ \wedge\ (m_{\text{REG}(\omega)}, \vec{d}_{\text{REG}(\omega)}) = k \ \wedge\ (\forall j < \text{REG}(\omega)).\big(m_j \models \varphi_1 \ \wedge\ (m_j \models \varphi_2) \Rightarrow T(j,\omega) < \alpha\big)\,\}\,. \quad (5.7)$$

Proof. For each $\omega \in \Omega_i^H(\alpha,\beta)$, it must be that $\text{OK}(\omega) \geq \text{REG}(\omega)$ and $t_{\text{REG}}(\omega) \in [\alpha,\beta]$ (i.e., ω encounters a conclusive state after reaching a regeneration point at some time in $[\alpha,\beta]$). The proof is analogous to the case of Proposition 5.5: in this case, states $(m_j, \vec{\tau}_j)$ with $j < \text{REG}(\omega)$ must satisfy φ_1, but not φ_2 when reached after time α, in order to guarantee that $\text{OK}(\omega) \geq \text{REG}(\omega)$.

Propositions 5.5 and 5.6 comprise an important result, as they apply renewal arguments on the satisfaction of the interval until operator and distinguish the properties that must be satisfied by paths before a regeneration point in $[0,\alpha)$ or in $[\alpha,\beta]$. We can now present our main result, which follows directly from Propositions 5.4 to 5.6 and shows that the measure $p_{(m_0,\vec{0})}(\alpha,\beta)$ of paths satisfying the until operator from the initial regeneration condition $(m_0,\vec{0})$ can be computed from the measures $p_i(\alpha,\beta)$ for all possible i, α, β.

Theorem 5.1. *The measures $p_i(\alpha,\beta)$ for all $i = (m,\vec{d})$, each corresponding to the probability that the model satisfies the interval until operator $\varphi_1\,\mathcal{U}^{[\alpha,\beta]}\varphi_2$ from the initial marking m with PDF of GEN timers given by the deterministic enabling times $\vec{d}$, are given by the system of integral equations*

$$\begin{aligned} p_i(\alpha,\beta) = {} & L_i^{\varphi_1,\varphi_2}(\alpha,\beta) \\ & + \sum_k \int_{x\in[0,\alpha)} dG_{ik}^{\varphi_1}(x)\, p_k(\alpha-x,\beta-x) \\ & + \sum_k \int_{x\in[\alpha,\beta]} dH_{ik}^{\varphi_1,\varphi_2}(\alpha,x)\, p_k(0,\beta-x) \end{aligned} \qquad (5.8)$$

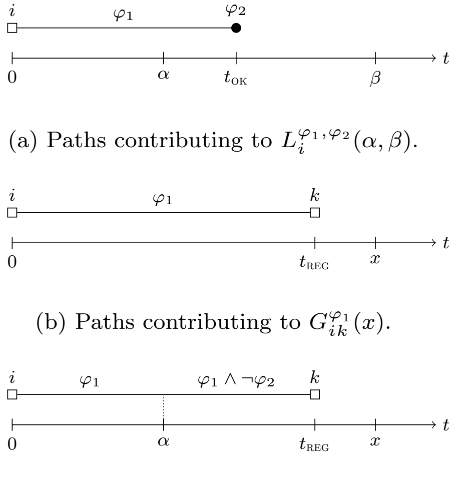
where i and k range over all reachable regeneration conditions.

The theorem comprises a bivariate generalization of Markov renewal equations with three kernels that result from a renewal argument specific to the interval until operator: the model can satisfy φ_2 between α and β either

- without regenerations,
- reaching the first regeneration before α, or
- reaching the first regeneration in $[\alpha,\beta]$.

As illustrated in Fig. 5.1, φ_1 must be always satisfied; additionally, also $\neg\varphi_2$ must be satisfied between α and the first regeneration point in paths that satisfy the until operator only after a regeneration.

The bivariate unknowns $p_i(\alpha,\beta)$ allow to take into account both a minimum and maximum time for the satisfaction of φ_2; after a regeneration at time x with regeneration condition k, the success probability is given by the solution from k with reduced time constrains: $p_k(\alpha-x,\beta-x)$ if $x<\alpha$ and



(a) Paths contributing to $L_i^{\varphi_1,\varphi_2}(\alpha,\beta)$.

(b) Paths contributing to $G_{ik}^{\varphi_1}(x)$.

(c) Paths contributing to $H_{ik}^{\varphi_1,\varphi_2}(\alpha,x)$.

Fig. 5.1: Constraints on paths contributing to the kernels.

$p_k(0,\beta-x)$ if $x \geq \alpha$. In the next section, we will show that the numerical solution of the integral equations for $p_i(\alpha,\beta)$ requires a number of unknowns $p_k(x,y)$ that grows linearly with respect to β, similarly to the required values of $L_i^{\varphi_1,\varphi_2}$ and $G_{ik}^{\varphi_1}$; in contrast, the number of required values of $H_{ik}^{\varphi_1,\varphi_2}$ grows linearly with the product $\alpha(\beta-\alpha)$, as illustrated in Fig. 5.2.

5.4 Numerical integration and kernels evaluation

The kernels can be evaluated through the enumeration of stochastic state classes limited to the first regeneration along sequences of discrete events; Eq. (5.8) can then be solved numerically in the time domain through techniques such as Newton–Cotes formulas or Runge–Kutta methods described in Brunner and van der Houwen (1986). Given a step size h and discretizing the temporal domain $[0,\beta]$ into points $t_n = nh$, with $\alpha = \bar{a}h$ and $\beta = \bar{b}h$, Newton–Cotes formulas define the linear system

$$\begin{aligned}\vec{p}(t_a,t_b) = {} & \vec{L}^{\varphi_1,\varphi_2}(t_a,t_b) \\ & + \sum_{m=0}^{a} w_m \, d\mathbf{G}^{\varphi_1}(t_m) \cdot \vec{p}(t_{a-m},t_{b-m}) \\ & + \sum_{m=a}^{b} w_m \, d\mathbf{H}^{\varphi_1,\varphi_2}(t_a,t_m) \cdot \vec{p}(0,t_{b-m})\end{aligned} \tag{5.9}$$

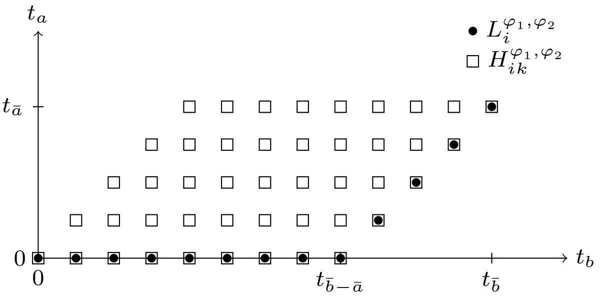

t_a
● $L_i^{\varphi_1,\varphi_2}$
□ $H_{ik}^{\varphi_1,\varphi_2}$
$t_{\bar{a}}$
0
0
$t_{\bar{b}-\bar{a}}$
$t_{\bar{b}}$
t_b

Fig. 5.2: Required values of $L_i^{\varphi_1,\varphi_2}(t_a, t_b)$ and $H_{ik}^{\varphi_1,\varphi_2}(t_a, t_b)$.

in the unknowns $\vec{p}(0, t_b)$ for $b = 0, \ldots, \bar{b}-\bar{a}$, and $\vec{p}(t_a, t_{a+\bar{b}-\bar{a}})$ for $a = 1, \ldots, \bar{a}$, where, for first degree formulas (*trapezoidal rule*), $w_m = h/2$ for $m = 0$, $m = a$, or $m = b$, and $w_m = h$ otherwise. For regular MRPs $d\mathbf{G}(0) = 0$ and $d\mathbf{H}(0, 0) = 0$, and Eq. (5.9) can be solved by forward substitution; in particular,

$$\vec{p}(0, t_b) = \vec{L}^{\varphi_1,\varphi_2}(0, t_b) + \sum_{m=1}^{b} w_m \, d\mathbf{H}^{\varphi_1,\varphi_2}(0, t_m) \cdot \vec{p}(0, t_{b-m})$$

for $b = 0, \ldots, \bar{b} - \bar{a}$, and

$$\begin{aligned} \vec{p}(t_a, t_b) = {} & \vec{L}^{\varphi_1,\varphi_2}(t_a, t_b) \\ & + \sum_{m=1}^{a} w_m \, d\mathbf{G}^{\varphi_1}(t_m) \cdot \vec{p}(t_{a-m}, t_{b-m}) \\ & + \sum_{m=a}^{b} w_m \, d\mathbf{H}^{\varphi_1,\varphi_2}(t_a, t_m) \cdot \vec{p}(0, t_{b-m}) \end{aligned} \tag{5.10}$$

for $a = 1, \ldots, \bar{a}$ and $b = a + \bar{b} - \bar{a}$. By evaluating the unknowns $\vec{p}(t_a, t_b)$ in this order, the solution $\vec{p}(t_{\bar{a}}, t_{\bar{b}})$ can be computed as a direct sum that requires:

- local kernel values $\vec{L}^{\varphi_1,\varphi_2}(0, t_b)$ for $b = 0, \ldots, \bar{b} - \bar{a}$ and $\vec{L}^{\varphi_1,\varphi_2}(t_a, t_b)$ for $a = 1, \ldots, \bar{a}$, $b = a + \bar{b} - \bar{a}$;
- global kernel values $d\mathbf{G}^{\varphi_1}(t_m)$ for $m = 1, \ldots, \bar{a}$;
- conditional global kernel values $d\mathbf{H}^{\varphi_1,\varphi_2}(t_a, t_m)$ for $a = 0, \ldots, \bar{a}$ and $m = a, \ldots, a + \bar{b} - \bar{a}$.

Values of $L_i^{\varphi_1,\varphi_2}(t_a, t_b)$, $dG_{ik}^{\varphi_1}(t_m)$, and $dH_{ik}^{\varphi_1,\varphi_2}(t_a, t_m)$ can be derived from the transient stochastic tree enumerated from the regeneration condition i, halting on either (1) regenerative nodes, (2) nodes not satisfying φ_1, (3) nodes with minimum reaching time τ_{age} greater than β w.p.1. In the enumeration, the successors of a class Σ, indicated as $\textsc{Successors}(\Sigma)$, are derived according to the calculus of stochastic state classes described in Chapter 3. In particular, each class Σ_n derived through the successions $\Sigma_{i-1} \xRightarrow{\gamma_i,\mu_i} \Sigma_i$ for $i = 1, \ldots, n$ is associated with the probability measure $\eta(\Sigma_n) = \prod_{i=1}^{n} \mu_i$ of the cylinder set of paths that perform the sequence of discrete events $\gamma_1, \ldots, \gamma_n$. Additional constraints on paths can be imposed by restricting the set of values of the times to fires of classes; in particular, given a transient stochastic state class $\Sigma = \langle m, D, f \rangle$ and the intervals I_1 and I_2, we indicate as $\Sigma_{in \in I_1, out \in I_2} = \langle m, D_{in \in I_1, out \in I_2}, f_{in \in I_1, out \in I_2} \rangle$, where

$$\begin{aligned} D_{in \in I_1,\, out \in I_2} &:= \big\{ (x_{age}, \vec{x}) \in D \mid -x_{age} \in I_1 \\ &\qquad\qquad \text{and } \big(\min_{i \neq age, *} x_i \big) - x_{age} \in I_2 \big\}, \\ \eta(\Sigma_{in \in I_1,\, out \in I_2}) &:= \eta(\Sigma) \int_{D_{in \in I_1,\, out \in I_2}} f(x_{age}, \vec{x}) \, dx_{age} \, d\vec{x}\,, \text{ and} \\ f_{in \in I_1,\, out \in I_2}(x_{age}, \vec{x}) &:= f(x_{age}, \vec{x}) \, \frac{\eta(\Sigma)}{\eta(\Sigma_{in \in I_1,\, out \in I_2})}\,, \end{aligned}$$

the class Σ conditioned to event imposing that the last firing happened at some time in I_1 and the next firing will happen at some time in I_2. Correspondingly, $\eta(\Sigma_{in \in I_1,\, out \in I_2})$ represents the measure of the cylinder set of paths where the firings that enter and leave Σ occur in the intervals I_1 and I_2, respectively. In the following, the superfluous restrictions $in \in [0, +\infty)$ and $out \in [0, +\infty)$ will be omitted in the notation.

Local kernel values $L_i^{\varphi_1,\varphi_2}(t_a, t_b)$*.* The algorithm in Fig. 5.3 evaluates $L_i^{\varphi_1,\varphi_2}(\alpha, \beta)$ by enumerating the transient tree from regeneration condition i. Specifically, Γ is the frontier set containing classes to be processed and p accumulates the value of $L_i^{\varphi_1,\varphi_2}(\alpha, \beta)$. For each non-regenerative class Σ selected from Γ, three cases are possible, depending on the satisfaction of φ_1 and φ_2:

- A state in a class $\neg\varphi_1 \wedge \varphi_2$ (line 8), contributes to the probability p if and only if it is reached in $[\alpha, \beta]$; according to this, p is incremented by the measure of the subset of Σ restricted with the constraint $in \in [\alpha, \beta]$.
- A state in a class $\varphi_1 \wedge \neg\varphi_2$ (line 10) does not contribute to p, but its successors can, provided that they are reached within β; therefore, the successors of Σ that are reached within β are added to Γ.
- A state in a class $\varphi_1 \wedge \varphi_2$ (line 12) can contribute to either p or the frontier Γ: p is incremented by the measure of the states in Σ that are reached within $[\alpha, \beta]$, or reached before α and left after α; the successors of Σ are added to Γ if and only if they are reached before α.

In the derivation of the values of $L_i^{\varphi_1,\varphi_2}(t_a, t_b)$ needed for the integration, the algorithm is executed for each pair $(0, t_b)$ with $b = 0, \ldots, \bar{b} - \bar{a}$ and (t_a, t_b) with $a = 1, \ldots, \bar{a}$ and $b = a + \bar{b} - \bar{a}$.

Global kernel values $dG_{ik}^{\varphi_1}(t_m)$*.* The values $dG_{ik}^{\varphi_1}(t_m)$ for $m = 1, \ldots, \bar{a}$ can be derived from the transient tree enumerated from regeneration condition i, stopping on any regenerative class, or on any $(\neg\varphi_1)$-class, or after the time bound α. The value $dG_{ik}^{\varphi_1}(t_m)$ can then be obtained by summing up, over each regenerative class n reached in the transient tree with regeneration k, the PDF value $f_{age}^{n}(-t_m)$ of the absolute reaching time multiplied by $\eta(n)$, i.e., $dG_{ik}^{\varphi_1}(t_m) = \sum_n \eta(n)\, f_{age}^{n}(-t_m)$.

Conditional global kernel values $dH_{ik}^{\varphi_1,\varphi_2}(t_a, t_m)$*.* The values $dH_{ik}^{\varphi_1,\varphi_2}(t_a, t_m)$ can be computed as $(H_{ik}^{\varphi_1,\varphi_2}(t_a, t_m) - H_{ik}^{\varphi_1,\varphi_2}(t_a, t_{m-1}))/h$, where the values $H_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$ are derived from the transient tree enumerated from regenerative condition i stopping on regenerations, on $(\neg\varphi_1)$-classes, and on classes reached after β. The evaluation also discards states in φ_2 classes that are left after α, since $H_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$ provides the measure of the set of paths that end on regeneration condition k after visiting only φ_1-states and without visiting any φ_2-state after α. The algorithm in Fig. 5.4 evaluates $H_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$ for all k by enumerating the transient tree from regeneration condition i; similarly to Fig. 5.3, Γ is the frontier set and p_k accumulates the value of $H_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$. For each state class Σ selected from Γ:

- A state in a regenerative class with regeneration condition k (line 6), contributes to the probability p_k of $H_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$ if and only if it is reached

```
EVALUATE-$L_i^{\varphi_1,\varphi_2}(\alpha,\beta)$
 1  $\Sigma_0 =$ initial state class with regeneration condition $i$
 2  $p \leftarrow 0$
 3  $\Gamma \leftarrow \{\Sigma_0\}$
 4  while $\Gamma \neq \varnothing$
 5       do select and remove a class $\Sigma = \langle m, D, f\rangle$ from $\Gamma$
 6          if $m \models \neg\varphi_1 \wedge \neg\varphi_2$ or $\Sigma$ is regenerative
 7             then discard $\Sigma$
 8          elseif $m \models \neg\varphi_1 \wedge \varphi_2$
 9             then $p \leftarrow p + \eta(\Sigma_{in\in[\alpha,\beta]})$
10          elseif $m \models \varphi_1 \wedge \neg\varphi_2$
11             then $\Gamma \leftarrow \Gamma \cup$ SUCCESSORS$(\Sigma_{out\in[0,\beta]})$
12          elseif $m \models \varphi_1 \wedge \varphi_2$
13             then $p \leftarrow p + \eta(\Sigma_{in\in[\alpha,\beta]})$
14                            $+ \eta(\Sigma_{in\in[0,\alpha),out\in[\alpha,+\infty)})$
15                  $\Gamma \leftarrow \Gamma \cup$ SUCCESSORS$(\Sigma_{out\in[0,\alpha)})$
16  return $p$
```

Fig. 5.3: Algorithm evaluating $L_i^{\varphi_1,\varphi_2}(\alpha, \beta)$.

```
EVALUATE-$\vec{H}_i^{\varphi_1,\varphi_2}(\alpha, x)$
 1  $\Sigma_0 =$ initial state class with regeneration condition $i$
 2  $p_k \leftarrow 0$ for each regeneration condition $k$
 3  $\Gamma \leftarrow \{ \Sigma_0 \}$
 4  while $\Gamma \neq \varnothing$
 5        do select and remove a class $\Sigma = \langle m, D, f \rangle$ from $\Gamma$
 6            if $\Sigma$ is regenerative with regeneration condition $k$
 7               then $p_k \leftarrow p_k + \eta(\Sigma_{in \in [0,x]})$
 8            elseif $m \models \varphi_1 \wedge \neg\varphi_2$
 9               then $\Gamma \leftarrow \Gamma \cup$ SUCCESSORS$(\Sigma_{out \in [0,x]})$
10            elseif $m \models \varphi_1 \wedge \varphi_2$
11               then $\Gamma \leftarrow \Gamma \cup$ SUCCESSORS$(\Sigma_{out \in [0,\alpha)})$
12  return $\vec{p}$
```

Fig. 5.4: Algorithm evaluating $\vec{H}_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$.

before time x; according to this, p_k is incremented by the measure of the subset of Σ restricted with the constraint $in \in [0, x]$.

- A state in a class $\varphi_1 \wedge \neg\varphi_2$ (line 8) does not contribute to any p_k, but its successors can, provided that they are reached within x; therefore, the successors of Σ that are reached within x are added to Γ.
- The successors of a state in a class $\varphi_1 \wedge \varphi_2$ (line 10) can contribute to $H_{ik}^{\varphi_1,\varphi_2}(\alpha, x)$ if the state is left before time α; according to this, the successors of Σ that are reached before α are added to Γ.

In the derivation of the values of $H_{ik}^{\varphi_1,\varphi_2}(t_a, t_m)$ needed for the integration, the algorithm is repeated for each pair (t_a, t_m) with $a = 0, \ldots, \bar{a}$ and $m = a - 1, \ldots, a + \bar{b} - \bar{a}$.

Overall, for each regeneration condition i, the transient tree enumeration is performed:

- $\frac{\alpha}{h} + 1$ times for the evaluation of $L_i^{\varphi_1,\varphi_2}(t_a, t_{a+\bar{b}-\bar{a}})$ with $a = 0, 1, \ldots, \bar{a}$;
- once for the evaluation of the global kernel values $dG_{ik}^{\varphi_1}(t_m)$ for $m = 1, \ldots, \bar{a}$ and all k;
- $(\frac{\alpha}{h} + 1)(\frac{\beta - \alpha}{h} + 2)$ times for the evaluation of the conditional global kernel values $H_{ik}^{\varphi_1,\varphi_2}(t_a, t_m)$ for $a = 0, 1, \ldots, \bar{a}$, $m = a - 1, \ldots, a + \bar{b} - \bar{a}$, and all k.

If $|R|$ is the number of reachable regeneration conditions, the number of transient tree enumerations is thus $O\big(\frac{\alpha}{h} \frac{\beta-\alpha}{h} |R|\big)$. The advantage with respect to Horváth et al. (2011), where model checking is performed with a single transient tree enumeration, lies in the reduced depth of these transient trees: the computation of successors for the leaves of the tree now halts not only on $\neg\varphi_1$ classes, but also on regenerative ones. Notably, both the worst-case space and time required for the computation of a successor class grow exponentially with the depth of the predecessor in the tree, as discussed in Carnevali et al. (2009); when the time bound β is large and regenerations are reached in a limited number of discrete events, the repeated enumeration of shallow trees becomes extremely beneficial. With respect to regenerative transient analysis of Chapter 4 and Horváth et al. (2012), which is based on univariate Markov renewal equations that require $O\big(\frac{\beta}{h}|R|\big)$ transient tree enumerations (or equivalent operations), the bivariate formulation of Eq. (5.8) requires a higher number of repetitions in order to explicitly account for the minimum time bound α. Nonetheless, this approach preserves regenerations before α that would be destroyed by a simple reduction to transient analysis, as discussed in the next section.

5.5 Eliminating the lower bound α

The availability of deterministic transitions in STPN models can be leveraged to remove the lower bound α for the satisfaction of φ_2 and reduce the evaluation of an interval until operator to a first-passage problem. We present this alternative approach by discussing the effect of $\alpha = 0$ on Eq. (5.8), and then the consequences of extending the model with an additional transition with duration equal to α.

The complexity of Eq. (5.8) is largely reduced if the until operator does not restrict the minimum time for the acceptance of the conclusive condition φ_2, i.e., if $\alpha = 0$. In this case, both Eq. (5.8) and its kernels are simplified: the local kernel $L_i^{\varphi_1,\varphi_2}(\alpha,\beta)$ becomes the probability that, starting from the regenerative state i, a φ_2-state is encountered before the first regeneration and not later than β. Moreover, the second term of Eq. (5.8) gives a null contribution. Finally, the conditional global kernel $H^{\varphi_1,\varphi_2}(\alpha,x)$ becomes the probability that a regeneration k is reached before x after visiting only states that satisfy $\varphi_1 \wedge \neg\varphi_2$.

The two kernels $L_i^{\varphi_1,\varphi_2}(0,\beta)$ and $H^{\varphi_1,\varphi_2}(0,x)$ can be derived from the transient trees rooted in regenerative classes reached within the first regenerative epoch, not later than β and through executions that visit only classes satisfying $\varphi_1 \wedge \neg\varphi_2$: the local kernel $L_i^{\varphi_1,\varphi_2}(0,\beta)$ is derived from the transient tree rooted in a class with regeneration condition i and limited to the first regeneration, or to time β, or to the first conclusive state that satisfies φ_2 or $\neg\varphi_1$; finally, $H_{ik}^{\varphi_1,\varphi_2}(0,x)$ is derived through the analysis of behaviors that reach the first regeneration within time β and visiting only states that satisfy $\varphi_1 \wedge \neg\varphi_2$.

This construction basically comprises an application of the strategy of Baier et al. (2003) to the context of non-Markovian processes. In fact, restrictions made in the enumeration of transient trees correspond to manipulations performed on the underlying stochastic process to turn any state that satisfies φ_2 or $\neg\varphi_1$ into an absorbing state.

The case $[0, \beta]$ can be lifted to solve the case $[\alpha, \beta]$ by exploiting the ability of STPNs to represent DET transitions. Following the same principle of techniques such as Donatelli et al. (2009) and Chen et al. (2009), which reduce probabilistic model checking to the analysis of a synchronous composition of the model with a specification automaton, the STPN model can be extended with a DET transition t with static density $f_t(x) = \delta(x-\alpha)$ and φ_2 can be restricted to $\varphi_2' = \varphi_2 \wedge \{t \text{ has fired}\}$. In so doing, regenerations before α are not exploited, since the added DET transition t is enabled and carries memory. Only after the firing of t at time α, the regenerative approach will be fully exploited in the analysis. This approach based on transient analysis is thus well-suited only for cases with a small α with respect to the duration of regenerative epochs.

Example 5.2 (G/D/1/2/2 queue). The property of Example 5.1 is not satisfied. In fact, the measure of paths that satisfy $\varphi_1 \, \mathcal{U}^{[\alpha,\beta]} \varphi_2$ from the initial marking $2\mathit{free}\,\mathit{operational}$ with $\varphi_1 = (\mathit{buffer} < 2)$ and $\varphi_2 = (\mathit{failed} = 1)$ corresponds to $0.3313 < 0.4$ for $\alpha = 0$ and $\beta = 7$, and to $0.2359 > 0.2$ for $\alpha = 2.5$ and $\beta = 7$. In the latter case, when limited to φ_1-markings, the model can reach only 3 distinct regenerations before time $\beta = 7$ and the corresponding transient trees include a total of 44 classes. In contrast, if a transition with deterministic value $\alpha = 2.5$ is added to the model, a total of 130 classes need to be enumerated. In particular, the transient tree enumerated from the initial regeneration includes 115 classes: this larger number is a consequence of the deterministic timer added to the initial state, which results in a higher number of transition firings required to reach the first regeneration.

5.6 Case study: Fischer's mutual exclusion protocol

We illustrate the proposed technique with reference to a stochastic model of n concurrent processes $P_1, P_2, \ldots, P_n$ accessing a critical section with Fischer's protocol, discussed in Lynch and Shavit (1992). The protocol ensures mutual exclusion using atomic read and write operations on a shared communication variable id taking the values $0, 1, \ldots, n$. When $id = 0$, each process P_i can attempt the access to the critical section. To this end, it performs the (time-consuming) write operation $id \leftarrow i$, waits for a time not lower than a given $t_{max} > 0$, and then reads id again: if $id = i$, it can access the critical section, and write $id \leftarrow 0$ on exit; whereas, if $id \neq i$, it has to wait until $id = 0$ to attempt again.

Fischer's protocol is a typical benchmark for real-time model checking, as it neatly illustrates the interaction between concurrency and firm timing: mutual exclusion is guaranteed provided that the waiting time t_{max} is not lower than the maximum time required by the write operation of any process. This condition inherently requires a model with multiple concurrent

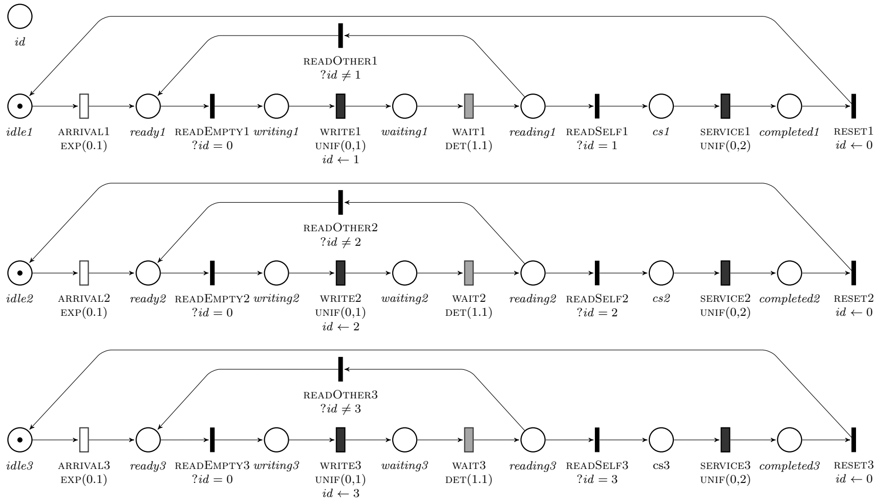


Fig. 5.5: STPN model of three processes accessing a critical section with Fischer's mutual exclusion protocol.

timers with upper and lower bounds. While the protocol has been verified in the qualitative perspective using real-time model checkers such as Kronos of Daws et al. (1996) and Uppaal of Bengtsson et al. (1996), randomized versions have been analyzed in closed-form in Gafni and Mitzenmacher (2001) or through simulation in Katoen et al. (2004); Deavours et al. (2002) only with timed activities modeled through exponential or gamma distributions. In this case, due to unbounded PDF supports, mutual exclusion can be violated with probability greater than zero.

We analyze quantitative properties in a stochastic model of the protocol allowing concurrency among GEN timers with bounded supports, thus enforcing with certainty the requirement of mutual exclusion. Fig. 5.5 illustrates an STPN model with three processes P_1, P_2, P_3 (the same scheme can be extended to any number of processes). The shared variable is encoded by the marking of place *id* (initially equal to zero). Each process P_i eventually leaves the idle condition with transition ARRIVAL$_i$ (EXP with rate 0.1), and enters the contention by reaching place $\mathit{writing}_i$ as soon as $\mathit{id} = 0$ (IMM transition READEMPTY$_i$ with enabling function $?\mathit{id} = 0$); it then sets the shared variable to its own identifier (as specified by the update function $\mathit{id} \leftarrow i$) at the end of a write operation (transition WRITE$_i$, with duration uniformly distributed over $[0, 1]$), and sojourns in a waiting state (place $\mathit{waiting}_i$) for a time higher than the maximum time that any process can spend writing to *id* (transition WAIT$_i$, DET equal to 1.1). When the wait completes, process P_i reads *id* again to ensure that its write was the last one (place $\mathit{reading}_i$): if $\mathit{id} \neq i$, the control goes back to the initial state

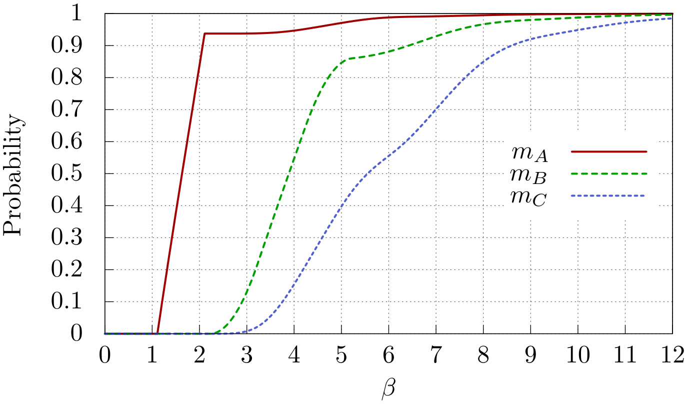


Fig. 5.6: The probability measure of paths satisfying TRUE $\mathcal{U}^{[0,\beta]}(cs_1 = 1)$ as a function of β, for markings $m_A = ready_1\ idle_2\ idle_3$, $m_B = 3id\ ready_1\ idle_2\ waiting_3$, $m_C = 3id\ ready_1\ writing_2\ waiting_3$.

of contention $ready_i$ (IMM transition READOTHER$_i$); whereas, if the shared variable is still equal to the process identifier (i.e., $id = i$), P_i enters the critical section cs_i (IMM transition READSELF$_i$), performs its service (transition SERVICE$_i$, uniform over $[0, 2]$), and then resets the shared variable (IMM transition RESET$_i$), returning to the idle state.

To illustrate the analysis, we consider a deadline requirement prescribing that the latency for the access of P_1 to the critical section be (1) not higher than β (which we call base deadline) with probability greater than p, and (2) not higher than $\beta_r > \beta$ (which we call relaxed deadline) with probability greater than $p_r > p$. This property can be encoded as the Boolean conjunction of two *probabilistic existence* properties

$$P_{>p}[\,\text{TRUE}\ \mathcal{U}^{[0,\beta]}(cs_1 = 1)\,]\ \wedge P_{>p_r}[\,\text{TRUE}\ \mathcal{U}^{[0,\beta_r]}(cs_1 = 1)\,]\,. \tag{5.11}$$

Fig. 5.6 reports the measure

$$Pr_{(m,\vec{0})}\{\omega \in \Omega_m \mid \omega \models \text{TRUE}\,\mathcal{U}^{[0,\beta]}(cs_1 = 1)\}$$

as a function of β, for $m \in \{m_A, m_B, m_C\}$ where: $m_A = ready_1\, idle_2\, idle_3$ (which occurs when P_1 becomes ready while the other processes are idle), $m_B = 3id\, ready_1\, idle_2\, waiting_3$ (which occurs when P_1 becomes ready and P_3 has just set the shared variable, closing the access to the contention), $m_C = 3id\, ready_1\, writing_2\, waiting_3$ (which occurs when P_1 becomes ready while both P_2 and P_3 are in the contention, with P_2 writing to id and P_3 waiting to check id after a write operation that closed the access to the contention). As intuitive, the latency of P_1 increases when the initial condition is changed from m_A to m_B, and then from m_B to m_C. Properties in the form of Eq. (5.11) are decided by comparing the probability measure computed for a given value of β with the threshold p. For example, with the initial condition m_A, for $\beta = 2$, $p = 0.90$, $\beta_r = 6$ and $p_r = 0.95$, we have that

$$P_{>p}[\,\text{TRUE}\ \mathcal{U}^{[0,\beta]}(cs_1 = 1)\,] = \text{FALSE}$$

and

$$P_{>p_r}[\,\text{TRUE}\ \mathcal{U}^{[0,\beta_r]}(cs_1 = 1)\,] = \text{TRUE}.$$

This indicates that the relaxed deadline is met with the required probability, but the base deadline is not.

Additional until properties allow to evaluate the probability measure of subsets of paths, so as to help the understanding of the role of different design parameters in the overall distribution of latency. For instance, the *probabilistic until* pattern

$$P_{>p}[\,(\textstyle\sum_{i\neq 1} cs_i = 0)\ \mathcal{U}^{[0,\beta]}(cs_1 = 1)\,] \tag{5.12}$$

evaluated from the initial marking m_A formulates a requirement on the measure of probability of the set of behaviors where P_1 is the first process accessing the critical section. In a practical perspective, this property expresses a bound p on the probability that P_1 is not overtaken in the access to the critical section by some process that was initially idle. The corresponding probability measure is determined by the trade-off between the rapidity of P_1 in completing the write operation (and thus preventing the access to contention by other processes) and the number $n-1$ and rate λ with which other processes enter the ready state. Results of the evaluation show that the probability of no-overtaking depends on the total offered load $(n-1)\lambda$, but it is relatively immune to the number of processes that produce this offered load. For instance: if $(n-1)\lambda$ is kept equal to 0.2 (to 0.1) while varying $n-1$ from 2 to 8, the probability of no-overtaking remains equal to 0.93 (to 0.96) with a variation lower than 0.001 (lower than 0.001).

When P_1 is overtaken, the service time of the overtaking process plays a twofold role: it determines the time that P_1 must wait before the next attempt, and it also determines the probability that more processes can enter the contention. The latter effect suggests that service times should be sufficiently low with respect to the total load of the system. For a quantitative assessment of the concept, we can formulate a requirement on the maximum probability that at least k processes enter the contention during the first failed attempt of P_1:

$$P_{<p}[\,(cs_1 = 0)\ \mathcal{U}^{[0,\beta]}(\textstyle\sum_{i=2}^{n} completed_i = 1\ \wedge \sum_{i=2}^{n} ready_i \geq k)\,]\,. \tag{5.13}$$

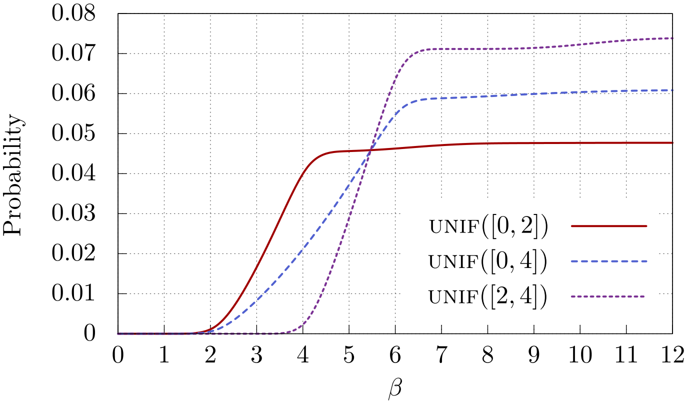


Fig. 5.7: The probability of paths satisfying the until operator $(cs_1 = 0)\ \mathcal{U}^{[0,\beta]}(\sum_{i=2}^{n} completed_i = 1 \ \wedge\ \sum_{i=2}^{n} ready_i \geq k)$ as a function of β, from marking $m_A = ready_1\, idle_2\, idle_3$ and for service times uniform on $[0, 2]$, $[0, 4]$ and $[2, 4]$, respectively.

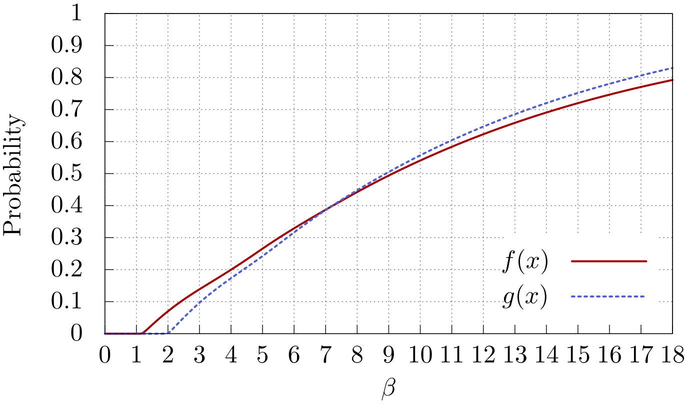


Fig. 5.8: The probability measure of paths satisfying TRUE $\mathcal{U}^{[0,\beta]}(cs_1 = 1)$ from $m_0 = idle_1\, idle_2\, idle_3$ when WRITE$_1$ is distributed according to truncated Erlang PDF $f(x) = xe^{-20x}/400$ over $[0, 1]$ (mean value 0.1) or its symmetrical $g(x) = f(1 - x)$ (mean value 0.9).

Fig. 5.7 reports the probability measure associated with this property (as a function of β, from the initial state m_A) for three service time distributions when $n = 3$ and $k = 1$.

While the impact of service times is intuitive, the role of writing time distributions is subtle: due to the *last-write-wins* policy of Fischer's protocol, a shorter writing time favors P_1 in keeping concurrent processes out of the contention, but, in case of contention, a longer writing time will favor P_1 in being the last process that completes its write to *id*, and thus the first one entering the critical section. To give a quantitative insight into this mechanism, we consider a setting in which the writing times of P_2 and P_3 are distributed uniformly over $[0, 1]$ (mean value 0.5), while the writing time of P_1 has either a truncated Erlang PDF $f(x) = xe^{-20x}/400$ over $[0, 1]$ (mean value 0.1) or its symmetrical $g(x) = (1 - x)e^{-20(1-x)}/400$ over $[0, 1]$ (mean value 0.9). Fig. 5.8 shows that, for $\beta < 7$, a faster writing time PDF $f(x)$ results in a higher probability that P_1 will reach the critical section from the initial marking $m_0 = \mathit{ready_1}\ \mathit{ready_2}\ \mathit{ready_3}$, while the slower PDF $g(x)$ is advantageous when $\beta > 7$. This result captures the following intuition: while the shorter mean value of f favors process P_1 in the first attempt, the longer mean value of g makes P_1 more competitive in trials subsequent to an initial overtaking; until time 7, the gain in the first attempt prevails, but after time 7, the competitive advantage in subsequent trails becomes more relevant. In this interpretation perspective, it is worth noting that the unbiased distribution with mean value 0.5 is always worse than one of the two biased distributions f and g.

As a last example, we evaluate the probability that process P_1 is in the critical section within a given time window $[\alpha, \beta]$ after an execution in which P_3 has never accessed the critical section. This property might be of interest in a problem of real-time testing where the system can be observed only within an interval $[\alpha, \beta]$ and the test case requires P_1 in the critical section without prior accesses of P_3. The requirement can be formulated as the probabilistic interval until

$$P_{>p}[\,(cs_3 = 0)\ \mathcal{U}^{[\alpha,\beta]}(cs_1 = 1)\,] \qquad (5.14)$$

and verified for given values of α, β and p so as to determine at which time α it is best to start the observation, or what is the minimum duration of $\beta - \alpha$ to obtain a probability of conclusive execution higher than a given threshold p. Fig. 5.9 plots the probability measure of paths satisfying $(cs_3 = 0)\ \mathcal{U}^{[\alpha,\beta]}(cs_1 = 1)$ for different values of α and duration $\beta - \alpha$ of the observation window. For each of the window sizes $\beta - \alpha \in \{0.1, 0.5, 1.0, 1.5, 2.0\}$, the probability measure $p_{(m_0,\vec{0})}(\alpha, \beta) := Pr_{(m_0,\vec{0})}\{\omega \in \Omega_{m_0} \mid \omega \models (cs_3 = 0)\ \mathcal{U}^{[\alpha,\beta]}(cs_1 = 1)\}$ is evaluated in Fig. 5.9 for $\alpha = 0, 0.1, \ldots, 4$ so as to select an optimum time α to start the observation. Note that, for each value δ of $\beta - \alpha$, the measures $p_{(m_0,\vec{0})}(\alpha, \alpha + \delta)$ for $\alpha = 0, 0.1, \ldots, 4$ are computed as the by-product of a single solution of Eq. (5.8) for the evaluation of $p_{(m_0,\vec{0})}(4, 4 + \delta)$ with step 0.1. For this measure, using a preliminary implementation, we report 17 distinct regeneration conditions reachable under $\varphi_1 = (cs_3 = 0)$, whose transient trees (limited to the first regeneration point) include 904 stochastic state classes. For $\alpha = 4$ and $\delta = 1$, adopting a step size $h = 0.1$, the enumeration of each

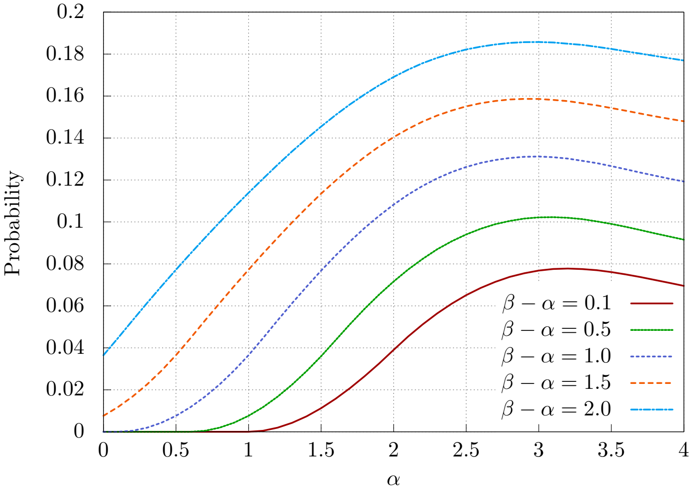


Fig. 5.9: The probability measure of paths satisfying $(cs_3 = 0)\ \mathcal{U}^{[\alpha,\beta]}(cs_1 = 1)$ from marking $m_0 = idle_1\ idle_2\ idle_3$.

tree is repeated $\frac{4}{0.1} + 1 = 41$ times for the evaluation of $L_i^{\varphi_1,\varphi_2}$, once for the evaluation of $dG_{ik}^{\varphi_1}(t_m)$, and $(\frac{4}{0.1} + 1)(\frac{1}{0.1} + 2) = 492$ times for the evaluation of $H_{ik}^{\varphi_1,\varphi_2}$, resulting in $482,736$ enumerated classes. As a comparison, the reduction to transient analysis through a deterministic timer $\alpha = 4$ described in Section 5.5 requires more than 8 million classes to be enumerated; of these, more than 99% belong to the enumerations of the initial transient tree, in which a regeneration is reached only after the elapse of α. Fig. 5.10 reports the total number of classes enumerated with the two approaches as a function of α for $\beta - \alpha = 1$.

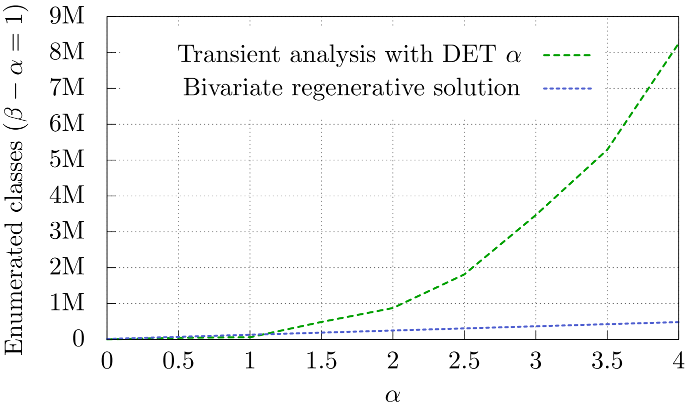


Fig. 5.10: Enumerated classes in the computation of the measure of paths satisfying $(cs_3 = 0)\ \mathcal{U}^{[\alpha,\beta]}(cs_1 = 1)$ from marking $m_0 = \mathit{idle_1}\ \mathit{idle_2}\ \mathit{idle_3}$.

Chapter 6
Conclusion

Markov renewal theory is the key to the analysis of non-Markovian stochastic systems over large time periods. We presented a solution for the transient analysis of systems in which multiple generally distributed timers can be started or stopped independently, but regenerations are encountered in a bounded number of discrete events. Notably, the approach introduced a novel concept of regeneration, extending the class of models amenable to state-of-the-art analytical or numerical techniques.

Although related to transient analysis, the verification of an interval until operator $\varphi_1 \mathcal{U}^{[\alpha,\beta]} \varphi_2$ in regenerative stochastic systems presents major challenges, both theoretical and practical, that cannot leverage existing approaches for CTMCs nor established results of Markov renewal theory.

Stochastic models with concurrent GEN timers accumulate memory over time: the state at time α does not summarize, in general, the past evolution of the system, and the process cannot be verified independently before and after α, in contrast to CTMCs (in which every time instant, and thus α, is a regeneration point). On the other hand, the reduction to a first-passage analysis problem requires the introduction of a deterministic timer in order to account for the minimum time α for the satisfaction of φ_2. Unfortunately, this approach crucially affects regenerative transient analysis: it is now the deterministic timer that carries memory until its elapse, in order to characterize the state distribution of the system at time α. Regeneration points before α are thus inevitably lost, forcing the enumeration of all sequences of discrete events before α.

To tackle this problem, we provided a solution based on the bivariate extension of Markov renewal equations, explicitly accounting for a satisfaction interval $[\alpha, \beta]$. The result is based on the formal definition of the probability space of STPN paths, which allowed to establish the theoretical relation between cylinder sets of paths and stochastic state classes; enumeration of stochastic state classes was in turn the basis for algorithms computing the kernels of bivariate Markov renewal equations.

The computation of the kernels requires to repeat the enumeration of stochastic state classes from each regeneration point for a number of times linear in $\alpha(\beta - \alpha)$, but each enumeration is limited to the first regenerative epoch and regeneration points are exploited also before α: since the number of feasible events grows exponentially with the time bound, repeating the analysis of a fixed number of shallow trees can produce considerable benefits when the time bound is large. Moreover, the enumeration is always restricted to paths satisfying the safety condition φ_1, and the underlying stochastic process is required to encounter regeneration points w.p.1 in a bounded number of events only on paths that always satisfy φ_1.

The benefits of the approach were demonstrated by a preliminary implementation in the analysis of a probabilistic model of Fischer's mutual exclusion protocol, a typical benchmark for real-time model checking. Notably, quantitative properties were analyzed in a stochastic model that guarantees the correctness of the protocol due to generally distributed timers with bounded supports. The construction of these results highlighted important problems of the probabilistic model checking of transient properties in regenerative systems, and can serve as the basis for further analysis techniques.

References

Ajmone Marsan, M., Balbo, G., Conte, G., Donatelli, S., and Franceschinis, G. (1995). *Modelling with Generalized Stochastic Petri Nets.* Wiley Series in Parallel Computing. John Wiley and Sons.

Ajmone Marsan, M., Conte, G., and Balbo, G. (1984). A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems. *ACM Trans. Comput. Syst.*, 2(2):93–122.

Baier, C., Haverkort, B., Hermanns, H., and Katoen, J.-P. (2003). Model-checking algorithms for continuous-time Markov chains. *IEEE Trans. Softw. Eng.*, 29(6):524–541.

Ballarini, P., Bertrand, N., Horváth, A., Paolieri, M., and Vicario, E. (2013). Transient Analysis of Networks of Stochastic Timed Automata using Stochastic State Classes. In *QEST'13*, volume 8054 of *LNCS*, pages 355–371. Springer.

Bengtsson, J., Larsen, K., Larsson, F., Pettersson, P., and Yi, W. (1996). *UPPAAL tool suite for automatic verification of real-time systems.* Springer.

Berthomieu, B. and Diaz, M. (1991). Modeling and Verification of Time Dependent Systems Using Time Petri Nets. *IEEE Trans. Softw. Eng.*, 17(3):259–273.

Bobbio, A., Puliafito, A., and Telek, M. (2000). A modeling framework to implement preemption policies in non-Markovian SPNs. *IEEE Trans. Softw. Eng.*, 26(1):36–54.

Brunner, H. and van der Houwen, P. (1986). *The numerical solution of Volterra equations*, volume 268. North-Holland Amsterdam.

Bryans, J., Bowman, H., and Derrick, J. (2003). Model checking stochastic automata. *ACM Trans. Comput. Logic*, 4(4):452–492.

Bucci, G., Carnevali, L., Ridi, L., and Vicario, E. (2010). Oris: a tool for modeling, verification and evaluation of real-time systems. *Int. J. on Softw. Tools for Techn. Transfer*, 12(5):391–403.

Carnevali, L., Grassi, L., and Vicario, E. (2009). State-density functions over DBM domains in the analysis of non-Markovian models. *IEEE Trans. Softw. Eng.*, 35(2):178–194.

Carnevali, L., Ridi, L., and Vicario, E. (2011). A framework for simulation and symbolic state space analysis of non-Markovian models. In *SAFECOMP'11*, volume 6894 of *LNCS*, pages 409–422. Springer.

Çinlar, E. (1975). Markov renewal theory: A survey. *Management Science*, 21(7):727–752.

Chen, T., Han, T., Katoen, J., and Mereacre, A. (2009). Quantitative Model Checking of Continuous-Time Markov Chains Against Timed Automata Specifications. In *LICS'09*, pages 309–318.

Choi, H., Kulkarni, V. G., and Trivedi, K. S. (1994). Markov regenerative stochastic Petri nets. *Perform. Eval.*, 20(1-3):337–357.

Ciardo, G., Blakemore, A., Chimento, P. F., Muppala, J. K., and Trivedi, K. S. (1993). Automated Generation and Analysis of Markov Reward Models Using Stochastic Reward Nets. In *Linear Algebra, Markov Chains, and Queueing Models*, volume 48 of *The IMA Volumes in Mathematics and its Applications*, pages 145–191. Springer.

Ciardo, G., German, R., and Lindemann, C. (1994). A characterization of the stochastic process underlying a stochastic Petri net. *IEEE Trans. Softw. Eng.*, 20(7):506–515.

Clark, A., Gilmore, S., Hillston, J., and Tribastone, M. (2007). Stochastic process algebras. In *Formal Methods for Performance Evaluation*, volume 4486 of *LNCS*, pages 132–179. Springer.

Daws, C., Olivero, A., Tripakis, S., and Yovine, S. (1996). The Tool KRONOS. In *Hybrid Systems III*, pages 208–219. Springer.

Deavours, D. D., Clark, G., Courtney, T., Daly, D., Derisavi, S., Doyle, J. M., Sanders, W. H., and Webster, P. G. (2002). The Mobius framework and its implementation. *IEEE Trans. Softw. Eng.*, 28(10):956–969.

Dill, D. L. (1990). Timing assumptions and verification of finite-state concurrent systems. In *AVMFSS'89*, volume 407 of *LNCS*, pages 197–212. Springer.

Donatelli, S., Haddad, S., and Sproston, J. (2009). Model checking timed and stochastic properties with CSL$^{\mathrm{TA}}$. *IEEE Trans. Softw. Eng.*, 35(2):224–240.

Gafni, E. and Mitzenmacher, M. (2001). Analysis of timing-based mutual exclusion with random times. *SIAM Journal on Computing*, 31(3):816–837.

German, R., Logothetis, D., and Trivedi, K. (1995). Transient analysis of Markov regenerative stochastic Petri nets: a comparison of approaches. In *International Workshop on Petri Nets and Performance Models (PNPM)*, pages 103–112.

Gross, D. and Miller, D. R. (1984). The randomization technique as a modeling tool and solution procedure for transient Markov processes. *Operations Research*, 32(2):343–361.

Grunske, L. (2008). Specification patterns for probabilistic quality properties. In *ICSE'08*, pages 31–40.

Haas, P. (2002). *Stochastic Petri Nets: Modelling, Stability, Simulation.* Springer.

Haas, P. J. and Shedler, G. S. (1986). Regenerative stochastic Petri nets. *Perform. Eval.*, 6(3):189–204.

Haas, P. J. and Shedler, G. S. (1989). Stochastic Petri net representation of discrete event simulations. *IEEE Trans. Softw. Eng.*, 15(4):381–393.

Horváth, A., Paolieri, M., Ridi, L., and Vicario, E. (2011). Probabilistic model checking of non-Markovian models with concurrent generally distributed timers. In *QEST'11*, pages 131–140. IEEE CS.

Horváth, A., Paolieri, M., Ridi, L., and Vicario, E. (2012). Transient analysis of non-Markovian models using stochastic state classes. *Perform. Eval.*, 69(7-8):315–335.

Infante-López, G., Hermanns, H., and Katoen, J.-P. (2001). Beyond Memoryless Distributions: Model Checking Semi-Markov Chains. In *PAPM-PROBMIV'01*, volume 2165 of *LNCS*, pages 57–70. Springer.

Katoen, J.-P., Bohnenkamp, H., Klaren, R., and Hermanns, H. (2004). Embedded software analysis with MOTOR. In *Formal Methods for the Design of Real-Time Systems*, pages 268–293. Springer.

Kulkarni, V. (1995). *Modeling and analysis of stochastic systems.* Chapman & Hall.

Lynch, N. and Shavit, N. (1992). Timing-based mutual exclusion. In *Real-Time Systems Symposium, 1992*, pages 2–11. IEEE.

Martinez, J. M. and Haverkort, B. R. (2006). CSL Model Checking of Deterministic and Stochastic Petri Nets. In *MMB'06 13th GI/ITG Conference*, pages 1–18. IEEE CS.

Puliafito, A., Scarpa, M., and Trivedi, K. S. (1998). Petri nets with k simultaneously enabled generally distributed timed transitions. *Perform. Eval.*, 32(1):1–34.

Sanders, W. H. and Meyer, J. F. (2001). Stochastic activity networks: Formal definitions and concepts. In *Lectures on Formal Methods and Performance Analysis*, pages 315–343. Springer.

Stewart, W. J. (1995). *Introduction to the Numerical Solution of Markov Chains.* Princeton University Press.

Vicario, E. (2001). Static analysis and dynamic steering of time-dependent systems. *IEEE Trans. Softw. Eng.*, 27(8):728–748.

Vicario, E., Sassoli, L., and Carnevali, L. (2009). Using stochastic state classes in quantitative evaluation of dense-time reactive systems. *IEEE Trans. Softw. Eng.*, 35(5):703–719.