# State to Function Labelled Transition Systems: A Uniform Framework for Defining Stochastic Process Calculi

*CNR-ISTI Technical Report 2011-TR-012, May 2011*

ROCCO DE NICOLA
Università di Firenze - Dipartimento di Sistemi e Informatica
DIEGO LATELLA
CNR - Istituto di Scienza e Tecnologie dell'Informazione "A. Faedo"
MICHELE LORETI
Università di Firenze - Dipartimento di Sistemi e Informatica
MIEKE MASSINK
CNR - Istituto di Scienza e Tecnologie dell'Informazione "A. Faedo"

**Abstract**

Process Algebras are one of the most successful formalisms for modeling concurrent systems and proving their properties. Due to the growing interest in the analysis of shared-resource systems, stochastic process algebras have received great attention. The main aim has been the integration of qualitative descriptions with quantitative (especially performance) ones in a single mathematical framework by building on the combination of Labeled Transition Systems and Continuous-Time Markov Chains. In this paper a unifying framework is introduced for providing the semantics of some of the most representative stochastic process languages; aiming at a systematic understanding of their similarities and differences. The used unifying framework is based on so called *State to Function Labelled Transition Systems*, FuTSs for short, that are state-transition structures where each transition is a triple of the form $(s, \alpha, \mathscr{P})$. The first and the second components are the source state and the label of the transition while the third component, $\mathscr{P}$, is the *continuation function* associating a value of a suitable type to each state $s'$. A non-zero value may represent the *cost* to reach $s'$ from $s$ via $\alpha$. The FuTSs framework is used to model representative fragments of major stochastic process calculi and the costs of *continuation function* do represent the *rate* of the exponential distribution characterizing the execution time of the performed action. In the paper, first the semantics for a simple language used to directly describe (unlabeled) CTMCs is provided, then a number of calculi that permit parallel composition of models according to the two main communication paradigms (multiparty and binary synchronization) are considered. Finally, languages where actions and rates are kept separated are modeled and the issues related to co-existence of stochastic and nondeterministic behaviors are discussed. For each calculus, the formal correspondence between the FuTSs semantics and its original SOS based semantics is established.

## 1 Introduction

Process Description Languages equipped with a formal operational semantics are one of the most successful formalisms for modeling concurrent systems and proving their properties. The operational semantics of Process Description Languages (PDLs) is defined by relying on Labeled Transition Systems (LTSs). By exploiting the so-called SOS (Structural Operational Semantics) approach, an LTS is "compositionally" associated to each term generated from the syntactic operators used to define the specific PDL. The states of the transition systems are just PDL terms while the labels of the transitions connecting states represent the possible actions, or interactions, and their effects.

PDLs often come equipped with observational mechanisms that permit equating (through behavioral equivalences) those systems that cannot be distinguished by external observations. LTSs, possibly corresponding to terms describing systems at different levels of abstraction, are then compared according to one of the many behavioral relations proposed in the literature, giving rise to so called Process Calculi (PCs). In some cases, the behavioral relations have also complete axiomatizations, in forms of equations, that exactly capture the relevant equivalences induced by the abstract operational semantics; then PCs are also called Process Algebras (PAs). Nowadays, PA, PC and PDL are often used as synonyms. In the context of the present paper, we will use the name PCs.

To define a process calculus, one starts with a set of uninterpreted action names (that might represent communication channels, synchronization actions, etc.) and with a set of elementary processes that, together with the actions, are the building blocks for forming new processes from existing ones. The basic operators are: *action prefix* ($a.\_$), *non-deterministic*, or *choice*, *composition* ($\_ + \_$) and *parallel composition* ($\_ \parallel \_$) of processes. For each operator there is a set of SOS rules describing the behavior of a system in terms of the behaviors of its components. As a result, each process term is seen as a component that can interact with other components or with the external environment.

Initially, PCs have been designed for modeling *qualitative*, i.e., functional (extensional) behavior, and their associated behavioral relations were mainly designed to assess whether two systems have comparable functional behavior, i.e., whether they could perform similar actions and induce similar changes on the current state. However, it was soon noticed that other aspects of concurrent systems, mainly related to systems performance, actions duration and probability, are at least as important as the functional ones. Thus, many variants of PCs have been introduced to take into account *quantitative* aspects of concurrent systems leading to proposals of *deterministically timed* PCs, *probabilistic* PCs, and *stochastically timed* PCs, usually referred to as *stochastic* PCs. Their semantics has then been rendered in terms of richer LTSs quotiented with new (*timed*, *probabilistic* and *stochastic*) behavioral relations.

Due to the growing interest in the analysis of shared-resource systems, stochastic PCs (SPCs) have received great attention. The main aim has been the integration of qualitative descriptions with quantitative (especially performance) ones in a single mathematical framework by building on the combination of LTSs and continuous-time Markov chains (CTMCs); the latter being one of the most successful approaches to modeling and analyzing the performance of computer systems and networks.

An overview on SPCs can be found in [22]; here, we would like to mention TIPP [19, 23], PEPA [26], EMPA [3, 1], IML [21], a language for Interactive Markov Chains (IMCs), and stochastic $\pi$-calculus [34]. More recently, also calculi for Mobile- / Service- Oriented Computing have been proposed [16, 11, 12, 33, 6, 13].

The common feature of the most prominent SPCs is that actions are enriched with rates of exponentially distributed random variables (RVs) characterizing their duration[1]. Although the same class of RVs is assumed in most SPCs, the models and the techniques underlying the definition of such calculi turn out to be significantly different in many respects.

Some differences are *conceptual*. For instance, a first distinguishing factor is, of course, the *process interaction paradigm*; although in most of the above mentioned SPCs CSP-like *multi-party* process synchronization is used, there are notable examples of SPCs with CCS-like *binary* process synchronization, like stochastic $\pi$-calculus [34] and stochastic CCS [29]. Another distinguishing factor is the *association* of rates with actions or processes. In the great majority of the proposals in the literature, rates are associated to single occurrences of actions, by means of some form of *rated action prefix* $\langle a, \lambda \rangle.P$, where $\lambda$ is the rate associated to $a$. In IML [21], instead, rates characterize process delays, by means of *rated prefix* $\lambda.P$. SPCs differ also in how the *rate of synchronizations* is defined as a function of the rates of the synchronizing actions.

Other differences, instead, are purely *technical*, in the sense that they stem from different approaches to address the same concept. The prominent example of such a situation is the modeling of the *race condition* principle, inherited from the theory of CTMCs, by means of the PCs choice operator, and its relationship to the issue of *transition multiplicity*. This principle implies that, for generic process $P$, an expression like $\langle a, \lambda \rangle.P + \langle a, \lambda \rangle.P$, where there are two different ways of executing $a$, both with (exponentially distributed

[1] In the literature, some authors assume actions have zero duration, in which case the associated RV is interpreted as a *delay*; see e.g. [19], [23], [21].

duration with) rate $\lambda$, should model the *same behavior as* $\langle a, 2\cdot\lambda\rangle.P$, since the rate of the sojourn time in state $\langle a,\lambda\rangle.P + \langle a,\lambda\rangle.P$ is $2\cdot\lambda$, and *not as* $\langle a,\lambda\rangle.P$, as it would be the case in a standard SOS interpretation of the expression. In other words, care must be taken in recording the number of different ways $P$ can be reached from $\langle a,\lambda\rangle.P + \langle a,\lambda\rangle.P$, i.e., two. This amounts to the issue of recording *transition multiplicity*. Several, significantly different, approaches have been proposed for handling transition multiplicity correctly. The proposals range from the use of *multi-relations* [26, 21], to that of *proved transition systems* [34, 19, 29], or LTS with *numbered transitions* [22], and *unique rate names* [16, 11], to mention just a few [2].

In [15], we have proposed a variant of LTSs, namely *Rate Transition Systems* (RTSs), as a tool for providing a uniform semantics to some of the most representative stochastic process languages. In LTSs, a transition is a triple $(P,\alpha,P')$ where $P$ and $\alpha$ are the source state and the label of the transition, respectively, while $P'$ is the target state reached from $P$ via the transition. In RTSs, a transition is a triple of the form $(P,\alpha,\mathscr{P})$. The first and second components are the source state and the label of the transition, as in traditional LTSs, while the third component $\mathscr{P}$ is the *continuation function* which associates a real non negative value to each state $P'$. A non-zero value represents the rate of the exponential distribution characterizing the time for the execution of the action represented by $\alpha$, necessary to reach $P'$ from $P$ via the transition. If, on the other hand, $\mathscr{P}$ maps $P'$ to 0, then state $P'$ is not reachable from $P$ via the transition. RTS continuation functions are equipped with a rich set of operations which provide a simple and clean solution to the transition multiplicity problem and make RTSs particularly suitable as a framework for the *compositional* definition of the operational semantics of fully Markovian SPCs, i.e., SPCs where non-deterministic behavior is fully replaced by the probabilistic one induced by the rates. In [14], we considered a limited number of prominent SPCs. In particular, we provided the RTS semantics for TIPP [23], EMPA [1], and PEPA [26], as representatives of the class of stochastic languages based on the CSP-like, multi-party interaction paradigm. Moreover, in [15] we considered stochastic CCS and stochastic $\pi$-calculus [34] as examples of languages based on the binary interaction paradigm.

To provide a uniform general account of the many SPCs mentioned above, in this paper we introduce *State to Function Labelled Transition Systems*, FuTSs for short, a natural generalization of RTSs. As mentioned above, in RTSs, the co-domain of continuation functions is required to be the set of non-negative real numbers, used as rates (or to represent non-reachability). In FuTSs the above constraint is removed: the co-domains of the continuation functions are only required to be *commutative semi-rings*, so that FuTSs are a *generic* framework. This provides a high level of flexibility while preserving basic properties on primitive operations like sum and multiplication. Moreover, the third component of the transition relation can be, in general, a disjoint union of sets of functions with different co-domains, thus allowing for the definition of different 'kinds' of transitions. Furthermore, also FuTSs are equipped with a rich set of (generic) operations on continuation functions, which makes the framework very well suited as semantic domain for the *compositional* definition of the operational semantics of Process Calculi, including SPCs and models where both non-deterministic behavior and stochastic delays are modeled, like in the Language of Interactive Markov Chains [21], IMCs, and (Continuous Time) Markov Decision Processes.

In this paper, by using FuTSs, we continue our program of a uniform and systematic understanding of similarities and differences of SPCs started in [14, 15]. We shall consider basically all the SPCs mentioned above[3], but we will focus only on the fragment of each calculus which is relevant for the stochastic extension, leaving out all those operators which are not directly affected by the extension. Our selection of calculi, has been guided by the aim of introducing most of the notions that have been considered for the different SPCs. We first consider a simple language for CTMCs, then we consider a number of calculi that permit parallel composition of models according to the two main communication paradigms (multi-party and binary synchronization). This will enable us to consider also the main issue related to modeling and combining the rates of active and passive actions and to preserve some generally desirable algebraic properties of the considered combinators. Finally, we consider languages where actions and rates are kept separated in the model and address the issue related to non-determinism by showing how IMCs can be

[2] It should however be noticed that, due to a minor technical imprecision (multi-relations are defined as the *least* multi-relation satisfying a set of SOS axioms and rules) both [26] and [21] obtain relations and not the intended multi-relations, thus failing to model transitions multiplicity.

[3] Only stochastic $\pi$-Calculus is not considered; its treatment of stochastic issues is the same as that for stochastic CCS. The interested reader is referred to [15].

represented by FuTSs.

We would like to remark that we do not have any aim at surveying *all* SPCs which have been proposed in the literature (the interested reader is referred to, e.g. [1, 22, 7]), neither to investigate pros and cons of the various approaches to the definition of the rates of synchronizing actions and related pragmatics (see, e.g. [25, 2]). Rather we aim at showing how the main techniques used to describe their semantics can be accommodated within a common simple unifying framework. We assume the reader to be familiar with the basic notions of SPCs; for tutorial presentations we refer to the papers mentioned above.

The formalisms we shall consider are the following:

1. a simple language of CTMCs;

2. three fully Markovian calculi that take different approaches to computing the rates resulting from multi-party (CSP-like) actions synchronization: $\text{TIPP}_k$, $\text{EMPA}_k$ and $\text{PEPA}_k$ ($k$ stands for *kernel*);

3. two calculi with binary action synchronization, which show two different approaches to computing synchronization rates within the CCS-like interaction paradigm: $\text{StoCCS}_I$ and $\text{StoCCS}_{II}$;

4. a CSP based calculus integrating Markovian behavior and purely non-deterministic one: $\text{IML}_k$.

The present paper is organized as follows: in Section 2 some preliminary notions and definitions are recalled and FuTSs are introduced; examples of how other structures like CTMCs, Discrete Time Markov Chains (DTMCs), RTSs and IMCs can be represented using FuTSs are presented as well. In Section 3 the process operators which will be used in subsequent sections for the specific SPCs are presented and briefly discussed. Section 4 introduces the FuTS semantics for a simple language for CTMCs. Section 5 addresses the issue of introducing parallel composition in the FuTS framework, thus paving the way for the next sections. Section 6 shows the FuTS semantics of significant fragments of SPCs based on the CSP interaction paradigm. The FuTS semantics of CCS-like calculi is presented in Section 7. The issue of coexisting Markovian and non-deterministic behaviors in the FuTS framework is addressed in Section 8. The original SOS definition of the relevant calculi as well as detailed proofs are reported in the appendices.

# 2 Preliminaries

In this section the notions of *commutative semi-ring* and *continuous time Markov chain* will be recalled and the *labelled function transition system* will be introduced.

## 2.1 Notation and basic definitions

We let $\mathbb{R}_{\geq 0}$ denote the set of real, non-negative numbers and, similarly, $\mathbb{R}_{>0}$ and $\mathbb{N}_{>0}$ denote the sets of positive real and natural numbers, respectively, and we let $\mathbb{B}$ denote the set of booleans. For any set $S$ we let $\wp S$ denote its power-set and $\wp_{fin} S$ the set of its *finite* subsets.

Many of the notions and definitions which we will use in the present paper are based on the concept of *commutative semi-ring*:

**Definition 2.1** *A* semi-ring *is a set* $\mathbb{S}$ *equipped with two binary operations,* $+_{\mathbb{S}}$*, usually called* sum*, and* $\cdot_{\mathbb{S}}$*, usually called* multiplication*, such that:* $(\mathbb{S}, +_{\mathbb{S}})$ *is a* commutative monoid *with neutral element* $0_{\mathbb{S}} \in \mathbb{S}$ *and* $(\mathbb{S}, \cdot_{\mathbb{S}})$ *is a* monoid *with neutral element* $1_{\mathbb{S}} \in \mathbb{S}$*. It is also assumed that multiplication* distributes *over sum and that* $0_{\mathbb{S}}$ annihilates $\mathbb{S}$ *with respect to multiplication. A semi-ring* $\mathbb{S}$ *is said* commutative *whenever also* $s_1 \cdot_{\mathbb{S}} s_2 = s_2 \cdot_{\mathbb{S}} s_1$*. Finally, we say that binary operation* $/^{\mathbb{S}}$ *is the* inverse *of* $\cdot_{\mathbb{S}}$ *if* $s_3 = s_1 /^{\mathbb{S}} s_2$ *if and only if* $s_1 = s_2 \cdot_{\mathbb{S}} s_3$*, for* $s_2 \neq 0_{\mathbb{S}}$*. Whenever clear from the context we will omit annotation* $\mathbb{S}$ *from operators.* •

Sets $\mathbb{B}$, with disjunction and conjunction, and $\mathbb{R}$, $\mathbb{N}$, with sum and products, are examples of (commutative) semirings. In $\mathbb{R}$, division is the *inverse* of product. In the sequel we let $\mathbb{C}, \mathbb{C}', \mathbb{C}_1, \ldots$ denote commutative semi-rings and $\frac{s_1}{s_2}$ denote $s_1/s_2$.

For generic non-empty set $S$ and commutative semi-ring $\mathbb{C}$, we let $\mathbf{TF}(S, \mathbb{C})$ denote the set of *total* functions from $S$ to $\mathbb{C}$, and we let $\mathscr{P}, \mathscr{Q}, \mathscr{R}, \ldots$ range over it.

We let $\mathbf{FTF}(S, \mathbb{C})$ denote the subset of $\mathbf{TF}(S, \mathbb{C})$ containing only those functions with *finite support*. $\mathscr{P}$ is an element of $\mathbf{FTF}(S, \mathbb{C})$ if and only if there exist $\{s_1, \ldots, s_m\} \subseteq S$, the *support* of $\mathscr{P}$, such that $\mathscr{P}\, s_j \neq 0_{\mathbb{C}}$ for $j = 1 \ldots m$ and $\mathscr{P}\, s = 0_{\mathbb{C}}$ for all $s \in S \setminus \{s_1, \ldots, s_m\}$.

**Definition 2.2 (Basic operators in $\mathbf{FTF}(S, \mathbb{C})$)** *If $s, s_1, \ldots, s_m \in S$, $\gamma_1, \ldots, \gamma_m \in \mathbb{C}$, $\mathscr{P}$ and $\mathscr{Q}$ in $\mathbf{FTF}(S, \mathbb{C})$, and $S' \subseteq S$ then:*

1. *$[s_1 \mapsto \gamma_1, \ldots, s_m \mapsto \gamma_m]_{\mathbb{C}}$ denotes the following function:*

$$[s_1 \mapsto \gamma_1, \ldots, s_m \mapsto \gamma_m]_{\mathbb{C}}\, s \ =_{\text{def}} \begin{cases} \gamma_j & \text{if } s = s_j \in \{s_1, \ldots, s_m\} \\ 0_{\mathbb{C}} & \text{otherwise} \end{cases}$$

*the $0_{\mathbb{C}}$ constant function in $\mathbf{FTF}(S, \mathbb{C})$ is rendered as $[]_{\mathbb{C}}$*

2. *function $+$ is defined as follows*

$$(\mathscr{P} + \mathscr{Q})\, s \ =_{\text{def}} \ (\mathscr{P}\, s) +_{\mathbb{C}} (\mathscr{Q}\, s)$$

3. *function $\bigoplus$ is defined as follows:*

$$\bigoplus \mathscr{P}\, S' \ =_{\text{def}} \ \sum_{s \in S'} {}_{\mathbb{C}} (\mathscr{P}\, s)$$

*where $\sum_{\mathbb{C}}$ denotes the n-ary extension of $+_{\mathbb{C}}$. We let $\oplus \mathscr{P}$ be defined as*

$$\oplus \mathscr{P} \ =_{\text{def}} \ \bigoplus \mathscr{P}\, S$$

•

Notice that $+$, $\bigoplus$ and $\oplus$ are well defined due to the definition of $\mathbf{FTF}(S, \mathbb{C})$.

## 2.2 Continuous Time Markov Chains

A CTMC is characterized by a set of states and by a *rate matrix*. The latter identifies system transitions in term of a set of random variables exponentially distributed.

**Definition 2.3 (Negative Exponential Distribution)** *A real valued random variable (RV) $D$ has a* negative exponential distribution *with* rate $\lambda \in \mathbb{R}_{>0}$ *if and only if the* probability *that $D \leq t \in \mathbb{R}$ ($\mathbb{P}\{D \leq t\}$) is $1 - e^{-\lambda \cdot t}$ for $t > 0$, and $0$ otherwise.* •

The expected value of an exponentially distributed RV with rate $\lambda$ is $\lambda^{-1}$ while its variance is $\lambda^{-2}$. Given the exponentially distributed independent RVs $D_1, \ldots, D_n$ with rates $\lambda_1, \ldots, \lambda_n$ respectively, the RV $\textsc{min}\{D_1, \ldots, D_n\}$ is exponentially distributed with rate $\lambda = \lambda_1 + \ldots + \lambda_n$, while $\mathbb{P}\{D_j = \textsc{min}\{D_1, \ldots, D_n\}\} = \frac{\lambda_j}{\lambda_1 + \ldots + \lambda_n}$.

**Definition 2.4 (CTMC)** *A* continuous time Markov chain *(CTMC) is a tuple $(S, \mathbf{R})$ where $S$ is a countable non-empty set of* states*, and $\mathbf{R} : S \to S \to \mathbb{R}_{\geq 0}$ is the* rate matrix*, where for all $s \in S$: $\sum_{s' \in S} \mathbf{R}\, s\, s' < \infty$.* •

We will often use the matrix notation $\mathbf{R}[s, s']$ for $\mathbf{R}\, s\, s'$. $\mathbf{R}[s, s'] > 0$ means that a transition from $s$ to $s'$ can be taken, while no transition from $s$ to $s'$ exists if $\mathbf{R}[s, s'] = 0$. The sojourn time at state $s$ before taking a transition is an exponentially distributed RV with rate $\sum_{s' \in S} \mathbf{R}[s, s']$ and the probability that the transition from $s$ to $s'$ is taken is $\frac{\mathbf{R}[s,s']}{\sum_{s'' \in S} \mathbf{R}[s,s'']}$.

Notice that the above definition allows $\mathbf{R}[s, s] > 0$, i.e. self-loops are allowed, while this is *not* the case in traditional definitions of CTMCs. Our choice is due to the fact that the operational semantics of the language we consider naturally leads to CTMCs with self-loops.

The following proposition, proved in Appendix A, guarantees that, as long as traditional measures of CTMCs like transient steady state probabilities are concerned, self-loops can be freely added or removed. Because, this more liberal definition does not affect the results of the analysis of CTMCs.

**Proposition 2.1** *The transient behavior of CTMC $C = (S, \mathbf{R})$ with $\mathbf{R}[\bar{s}, \bar{s}] > 0$ for some $\bar{s} \in S$ coincides with that of CTMC $\tilde{C} = (S, \tilde{\mathbf{R}})$, such that:*

$$\tilde{\mathbf{R}}[s, s'] =_{\text{def}} \begin{cases} 0 & \text{if } s = s' \\ \mathbf{R}[s, s'] & \text{otherwise} \end{cases}$$

□

## 2.3 Labelled Function Transition Systems

In this section we define a simple notion of FuTSs, which is sufficient as underlying model for all fully Markovian SPCs presented in the paper; in Section 8 the definition will be extended to that of *general* FuTSs, which support also languages and models where Markovian behaviour and pure non-determinism coexist.

In the simplest case a FuTS is tuple $(S, A, \mathbb{C}, \rightarrowtail)$ where $S$ is the set of *states*, $A$ is the set of transition *labels*, $\mathbb{C}$ is a commutative semi-ring, and $\rightarrowtail \subseteq S \times A \times \mathbf{TF}(S, \mathbb{C})$ is the *transition relation*. As usual, we let $s \overset{\alpha}{\rightarrowtail} \mathscr{P}$ denote $(s, \alpha, \mathscr{P}) \in \rightarrowtail$. Intuitively, $s_1 \overset{\alpha}{\rightarrowtail} \mathscr{P}$ and $(\mathscr{P}\ s_2) = \gamma \neq 0_{\mathbb{C}}$ means that $s_2$ is reachable from $s_1$ via (the execution of) $\alpha$ with a value $\gamma \in \mathbb{C}$. $(\mathscr{P}\ s_2) = 0_{\mathbb{C}}$ means that $s_2$ is not reachable from $s_1$ via $\alpha$.

In the context of fully Markovian SPCs, where the relevant commutative semi-ring $\mathbb{C}$ is $\mathbb{R}_{\geq 0}$, $\gamma$ may represent either the duration of the execution of an action $a$, conventionally denoted by $\delta_a$ in this paper, or just the passage of time, conventionally denoted by $\delta$ in this paper. In any case, such time interval is modeled as an exponentially distributed RV, and $\gamma (> 0)$ is the relevant rate. In the sequel, in order to emphasize the exponential nature of random delays, we use $\delta^e$ in place of $\delta$, and $\delta^e_a$ in place of $\delta_a$, letting set $\Delta_{\mathcal{A}}$ be defined as the set $\{\delta^e_a \mid a \in \mathcal{A}\}$, where $\mathcal{A}$ is a given set of *actions*.

**Definition 2.5 (Labelled Function Transition Systems)** *An $A$-*labelled function transition system *(*FuTS*) over $\mathbb{C}$ is a tuple $(S, A, \mathbb{C}, \rightarrowtail)$ where $S$ is a countable, non-empty, set of* states*, $A$ is a countable, non-empty, set of transition* labels*, $\mathbb{C}$ is a commutative semi-ring, and $\rightarrowtail \subseteq S \times A \times \mathbf{TF}(S, \mathbb{C})$ is the* transition relation*.* •

Whenever necessary or convenient an *initial* state $s_0 \in S$ will be identified, and the relevant FuTS will be the $(S, A, \mathbb{C}, \rightarrowtail, s_0)$. Henceforth, FuTSs will be denoted by $\mathcal{R}, \mathcal{R}_1, \mathcal{R}', \ldots$. Whenever $s \overset{\alpha}{\rightarrowtail} \mathscr{P}$ is a transition of $\mathcal{R}$, we call $\mathscr{P}$ *next state function*, or *continuation function*, or, simply, *continuation*. In Section 8 the definition above will be extended in order to one of general FuTS, where different type of continuations are allowed in the same FuTS.

**Definition 2.6** *Let $\mathcal{R} = (S, A, \mathbb{C}, \rightarrowtail)$ be a* FuTS*, then:*

1. *$\mathcal{R}$ is* total *if for all $s \in S$ and $\alpha \in A$ there exists $\mathscr{P} \in \mathbf{TF}(S, \mathbb{C})$ such that $s \overset{\alpha}{\rightarrowtail} \mathscr{P}$;*
2. *$\mathcal{R}$ is* deterministic *if for all $s \in S$, $\alpha \in A$, and $\mathscr{P}, \mathscr{Q} \in \mathbf{TF}(S, \mathbb{C})$ we have that the following holds: $s \overset{\alpha}{\rightarrowtail} \mathscr{P}, s \overset{\alpha}{\rightarrowtail} \mathscr{Q} \Longrightarrow \mathscr{P} = \mathscr{Q}$;*
3. *$\mathcal{R}$ is a* finite support FuTS *(*$\text{FuTS}_{FS}$ *for short) if $\rightarrowtail \subseteq S \times A \times \mathbf{FTF}(S, \mathbb{C})$.* •

Henceforth, we will consider only *total*, *deterministic* and *finate support* FuTS, since they are powerful enough to model the process calculi we are interested in. Please notice that, the fact that we consider deterministic FuTS does *not* imply that they can only be used to model deterministic behaviors (see Section 8.2).

**Remark 2.1** *It is easy to see that standard CTMCs are isomorphic with total deterministic $\{\delta^e\}$-labeled* $\text{FuTS}_{FS}$ *over $\mathbb{R}_{\geq 0}$. Similarly, by conventionally using the label $\pi$ for denoting discrete random experiments, DTMCs are isomorphic with total deterministic $\{\pi\}$-labeled* $\text{FuTS}_{FS}$ *over $[\mathbf{0}, \mathbf{1}]$, where $[\mathbf{0}, \mathbf{1}]$ is a proper commutative semi-ring built on the* $[0, 1]$ *interval and every continuation $\mathscr{P}$ is a probability distribution function, i.e. $\oplus \mathscr{P} = 1$. RTSs coincide with $\Delta_{\mathcal{A}}$-labeled* $\text{FuTS}_{FS}$ *over $\mathbb{R}_{\geq 0}$.* □

The definition of *bisimilarity* is recalled below.

| | | | |
|---|---|---|---|
| $P, Q$ | $::=$ | $\mathbf{nil}$ | [*inaction*] |
| | $\mid$ | $\lambda.P$ | [*rate prefix*] |
| | $\mid$ | $a.P$ | [*action prefix*] |
| | $\mid$ | $\langle a, \lambda \rangle.P$ | [*rated-action prefix*] |
| | $\mid$ | $\langle a, *_\omega \rangle.P$ | [*passive-action prefix*] |
| | $\mid$ | $\bar{a}^\lambda.P$ | [*rated-output-action prefix*] |
| | $\mid$ | $a^\lambda.P$ | [*rated-input-action prefix*] |
| | $\mid$ | $a^{*_\omega}.P$ | [*passive-input-action prefix*] |
| | $\mid$ | $P + P$ | [*choice composition*] |
| | $\mid$ | $P \parallel_L P$ | [*multi-party synchronization composition*] |
| | $\mid$ | $P \mid P$ | [*binary synchronization composition*] |
| | $\mid$ | $X$ | [*constant*] |

Figure 1: Syntax of Stochastic Process Calculi operators.

**Definition 2.7** *Given a* FuTS $(S, A, \mathbb{C}, \rightarrowtail)$, *we say that an equivalence relation* $B \subseteq S \times S$ *is a bisimulation relation if and only if, whenever* $(s_1, s_2) \in B$, *for all* $\alpha \in A$ *and equivalence class* $C \subseteq S$, *and* $\mathscr{P}_1$, *if* $s_1 \overset{\alpha}{\rightarrowtail} \mathscr{P}_1$ *then there exists* $\mathscr{P}_2$ *such that* $s_2 \overset{\alpha}{\rightarrowtail} \mathscr{P}_2$ *and:* $\bigoplus \mathscr{P}_1\, C = \bigoplus \mathscr{P}_2\, C$. *We say that* $s_1$ *and* $s_2$ *are* bisimilar, *written* $s_1 \sim s_2$, *if and only if* $(s_1, s_2) \in B$ *for some bisimulation relation* $B$. •

We close this section with the definition of reachable states and of FuTS generated from a state.

**Definition 2.8 (Reachable states)** *For* $\text{FuTS}_{FS}$ $\mathcal{R} = (S, A, \mathbb{C}, \rightarrowtail)$, $s \in S$, *and* $\Gamma \subseteq A$, *the set of states reachable from* $s$ *via* $\Gamma$, *denoted* $S_{s,\Gamma}$, *is the least set satisfying both conditions below:*

1. $s \in S_{s,\Gamma}$;
2. *if* $s' \in S_{s,\Gamma}$, $s' \overset{\alpha}{\rightarrowtail} \mathscr{P}$ *for some* $\alpha \in \Gamma$ *and* $\mathscr{P} \in \mathbf{FTF}(S, \mathbb{C})$ *with* $\mathscr{P}\, s'' \neq 0_{\mathbb{C}}$, *then* $s'' \in S_{s,\Gamma}$.

*We furthermore define the set of actions associated to the set of states reachable from* $s$ *via* $\Gamma$, $A_{s,\Gamma} \subseteq A$, *as follows:*

$$A_{s,\Gamma} =_{\text{def}} \{\alpha \in A \mid \exists s' \in S_{s,\Gamma}, \mathscr{P} \in \mathbf{FTF}(S, \mathbb{C}).\ s' \overset{\alpha}{\rightarrowtail} \mathscr{P} \text{ and } \mathscr{P} \neq []_{\mathbb{C}}\}$$

*and, for* $\mathscr{P} \in \mathbf{FTF}(S, \mathbb{C})$, *we let* $\mathscr{P}_{s,\Gamma} \in \mathbf{FTF}(S_{s,\Gamma}, \mathbb{C})$ *denote* $\mathscr{P}_{|(S_{s,\Gamma})}$, *i.e. the restriction of* $\mathscr{P}$ *on* $S_{s,\Gamma}$. *Finally, we consider the restricted transition relation,* $\rightarrowtail_{s,\Gamma}$, *defined as the set* $\{(s', \alpha, \mathscr{P}_{s,\Gamma}) \mid s' \in S_{s,\Gamma}, \alpha \in A_{s,\Gamma}, s' \overset{\alpha}{\rightarrowtail} \mathscr{P}\}$ •

The $\text{FuTS}_{FS}$ generated from a state $s$ is defined below:

**Definition 2.9 (Generated $\text{FuTS}_{FS}$)** *Let* $\mathcal{R} = (S, A, \mathbb{C}, \rightarrowtail)$ *be a* $\text{FuTS}_{FS}$, $s \in S$, *and* $\Gamma \subseteq A$. *The* $\text{FuTS}_{FS}$ generated *from* $s$ *and* $\Gamma$, *denoted by* $\mathcal{R}_{s,\Gamma}$, *is defined as follows:* $\mathcal{R}_{s,\Gamma} =_{\text{def}} (S_{s,\Gamma}, A_{s,\Gamma}, \mathbb{C}, \rightarrowtail_{s,\Gamma}, s)$. •

# 3 Operators of Stochastic Processes

We will now address the modeling of several SPCs proposed in the literature, with different syntax and semantics. All the considered languages are so called fully Markovian, in the sense that non-deterministic behavior is fully replaced by the probabilistic one induced by the rates. The only considered exception is IML, where it is possible to specify non-deterministic behavior. Indeed, there are two different kinds of actions: one, purely stochastic, modeling the passage time and the other modeling processes actions and interactions.

In Figure 1 we introduce a uniform syntax for all SPCs of interest. We will thus not always use the original syntax of the individual SPCs. For the sake of completeness, in Appendix B, an account of the original syntax is given for each SPC.

In the grammar of Figure 1 we assume non-empty, countable sets $\mathcal{A}$ of *actions*, ranged over by $a, a', a_1, \ldots$, $\bar{\mathcal{A}}$ of *co-actions*, ranged over by $\bar{a}, \bar{a}', \bar{a}_1, \ldots$, while $\lambda$, $\lambda'$, $\lambda_1, \ldots$, $\omega, \omega', \omega_1, \ldots$ denote *rates* and *weights* in $\mathbb{R}_{>0}$.

Term **nil** denotes the process that is unable to get involved in any action.

The syntax contains many types of action prefixes needed to address different types of calculi. *Rate prefix* $\lambda.P$, delays execution of $P$ by an interval $\delta^e$, the duration of which is an exponentially distributed RV with rate $\lambda \in \mathbb{R}_{>0}$. *Action prefix* $a.P$ starts with the execution of action $a$ and then continues with that of $P$; the execution of $a$ is *duration-less* or instantaneous, i.e. takes no time. These two prefix operators are the key ones for calculi like IML.

In *rated-action prefix* $\langle a, \lambda \rangle.P$ the duration $\delta^e_a$ of the execution of action $a$ is an exponentially distributed RV with rate $\lambda$; after completion of the execution of $a$, the behavior continues as in $P$. Rated-action prefix is typical of most SPCs which have been proposed in the literature and which are based on the *multi-party*, CSP-like, synchronization paradigm. In such languages there is often the possibility of leaving the durations of some of the action 'unspecified', in the sense that, for any such 'passive action', the actual duration is determined by the rate of the 'active' action with which it synchronizes, if any. The *passive-action prefix* operator $\langle a, *_\omega \rangle.P$ serves this purpose; weight $\omega$ is used for determining a probabilistic distribution in case there is more than one passive action which may synchronize with the same active one.

*Rated-input-action prefix* $a^\lambda.P$, and *rated-output-action prefix* $\bar{a}^\lambda.P$ are used to model CCS-like stochastic calculi, where a *binary* synchronization paradigm is used. In this calculi duration rates are associated to both to input and output actions. There are also proposals in which input actions are considered passive *by definition*, thence the *passive-input-action prefix* $a^{*_\omega}.P$. In all the above cases, after executing $a$, the process continues as $P$.

The *choice* composition operator is denoted by $P_1 + P_2$. In fully Markovian calculi the term $P_1 + P_2$ is interpreted according to the *race condition* principle of CTMCs. For instance, the sojourn time in state $\langle a, \lambda \rangle.\mathbf{nil} + \langle b, \mu \rangle.\mathbf{nil}$ is exponentially distributed with rate $\lambda + \mu$. There is a race between the execution of action $a$ and action b. The probability that the race is 'won' by $a$ ($b$, respectively) is $\frac{\lambda}{\lambda+\mu}$ ($\frac{\mu}{\lambda+\mu}$, respectively).

The *multi-party parallel* composition operator, denoted by $P_1 \parallel_L P_2$ where $L \in (\wp_{fin} \mathcal{A})$ is the synchronization (or cooperation) set, corresponds to CSP parallel composition that requires actions in $L$ to be performed synchronously and the others independently. The *binary parallel* composition, denoted by $P_1 \mid P_2$, is the parallel operator used in the CCS-based calculi, that is used to model synchronization of complementary actions.

To associate name a constant name $X$ to a process $P$, an equation of the form $X := P$ is used, where constants $X, X_1, X', \ldots$ may occur only guarded in $P$, i.e. within the scope of a prefix. A set $E$ of defining equations is *complete* and *consistent* if and only if it contains exactly one equation for each constant name of the language.

In the sequel, the FuTS semantics of a class of calculi is given. For each calculus $C$, assuming complete and consistent set of defining equations $E$, a FuTS $\mathcal{R}_C$ is defined, corresponding to the complete language of the calculus, under $E$, this set will not be always explicitly mentioned. The set of states of the FuTS coincides with the set of terms $\mathcal{P}_C$ of the calculus; the label set $\mathcal{L}_C$ typically (but not necessarily) refers to sets $\mathcal{A}$ and $\bar{\mathcal{A}}$ of actions and co-actions; the transition relation is defined by means of a logical deduction system and depends on the equations in $E$. The FuTS of *a single* process $P \in \mathcal{P}_C$, $(\mathcal{R}_C)_{P,\mathcal{L}_C}$, is obtained from $\mathcal{R}_C$ by considering only the states *generated* from $P$.

Formal semantics of SPCs are often defined by means of Structured Operational Semantics (SOS) that lead to labelled transition systems (LTSs) or *multi*-transition systems, i.e. LTSs where the transition relation is in fact a *multi*-relation. Such (multi-)transitions are usually labelled by rates $\lambda$, and/or action labels. In the presence of passive actions, transitions can be labelled by actions and weights $\omega$. Let $\rightarrow$ be the transition relation of the LTS defined according to the SOS of the calculus at hand. Henceforth, we let $\mathbf{rt}(P_1, P_2)$, $\mathbf{rt}_a(P_1, P_2)$, and $\mathbf{wt}_a(P_1, P_2)$ denote the *cumulative* rate over *all* transitions from $P_1$ to $P_2$ and the *cumulative* rate and weight, respectively, over all $a$-labelled transitions, from $P_1$ to $P_2$. Formally:

$$\mathbf{rt}(P_1, P_2) =_{\text{def}} \sum \{\!\!\{ \lambda | P_1 \xrightarrow{\lambda} P_2 \}\!\!\}$$

$$\mathbf{rt}_a(P_1, P_2) =_{\text{def}} \sum \{\!\!\{ \lambda | P_1 \xrightarrow{a,\lambda} P_2 \}\!\!\}$$

$$\text{(NIL)}\ \frac{}{\mathbf{nil} \overset{\alpha}{\rightarrowtail} []_{\mathbb{R}_{\geq 0}}} \qquad \text{(RPF1)}\ \frac{}{\lambda.P \overset{\alpha}{\rightarrowtail} [P \mapsto \lambda]}$$

$$\text{(CHO)}\ \frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}}{P + Q \overset{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}} \qquad \text{(CNS)}\ \frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, X := P}{X \overset{\alpha}{\rightarrowtail} \mathscr{P}}$$

Figure 2: Semantics Rules for the Language of CTMCs.

$$\mathbf{wt}_a(P_1, P_2) =_{\text{def}} \sum \{\!| \omega | P_1 \xrightarrow{a,*_\omega} P_2 |\!\}$$

where $\{\!| \_ |\!\}$ dentes a multi-set and $\sum\{\!||\!\} =_{\text{def}} 0$. Moreover, $\xrightarrow{\lambda}$, $\xrightarrow{a,\lambda}$ and $\xrightarrow{a,*_\omega}$ identify a generic transition, an $a$-labelled transition with rate $\lambda$, an $a$-labelled transition with weight $\omega$, respectively.

# 4 A Language for CTMCs

In this section we define a simple language for describing CTMCs like in [22]. The set $\mathcal{P}_{CTMC}$ of CTMC terms is defined by the grammar obtained by selecting from Figure 1 the following operators:

- *inaction*,
- *rate prefix*,
- *choice composition*, and
- *constant*.

CTMCs are composed of states, transitions between states, and rates characterizing transition delays. CTMCs do not contain any indication about the actions corresponding to the actual transitions. Thus, to define the operational semantics of our simple language, we denote the (action-less) passage of time with $\delta^e$. Consequently, the FuTS associated to the considered language, $\mathcal{R}_{CTMC}$, consists of the set of states $\mathcal{P}_{CTMC}$, the labels $\mathcal{L}_{CTMC} = \{\delta^e\}$ and the transition relation $\rightarrowtail$ defined in Figure 2, there $\rightarrowtail$ associates to each state $s$ and to label $\alpha$ $(= \delta^e)$, a function in $\mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$.

**Definition 4.1 (Formal semantics of the language for CTMCs)** *The formal semantics of the calculus for CTMCs is the* FuTS$_{FS}$ $\mathcal{R}_{CTMC} =_{\text{def}} (\mathcal{P}_{CTMC}, \mathcal{L}_{CTMC}, \mathbb{R}_{\geq 0}, \rightarrowtail)$ *where the transition relation* $\rightarrowtail \subseteq \mathcal{P}_{CTMC} \times \mathcal{L}_{CTMC} \times \mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$ *is the least relation satisfying the rules of Figure 2.* •

Intuitively, from the FuTS Semantics Rules in Figure 2 postulates that no state is reachable from **nil** while state $P$ is reachable from $\lambda.P$ with rate $\lambda$. The rule for choice postulates that $P + Q$ reaches a state $R$ with a rate resulting from the sum of the rates of the individual components. If one of the components cannot reach $R$, say $Q$, then $(\mathscr{Q}\ R) = 0$ and only the contribution of the other is taken into account. The presence of $X := P$ in the premises for the rule for constant definition is intended as the fact that $X := P$ is an element of the relevant set $E$ of defining equations and $X$ behaves exactly like $P$. The CTMC associated to term $X$, when $X := \lambda.X + \mu_1.\mathbf{nil} + \mu_1.\mathbf{nil}$, is the one of Figure 3.

The use of FuTSs, in particular in the rule for choice, naturally deals with *race conditions* and solves the related *transition multiplicity* issue in a simple and elegant way. In Figure 2 the possible transitions associated to $\lambda.R_1 + \mu.R_2$ are presented where continuation functions associating rates to future behaviours are represented as dotted arrows. In this simple example, the following transitions can be derived depending on the actual values of $R_1$, $R_2$, $\lambda$ and $\mu$:

- if $R_1 \neq R_2$ then $\lambda.R_1 + \mu.R_2 \overset{\delta^e}{\rightarrowtail} [R_1 \mapsto \lambda, R_2 \mapsto \mu]$;
- if $R_1 = R_2$ then $\lambda.R + \mu.R \overset{\delta^e}{\rightarrowtail} [R \mapsto \lambda + \mu]$

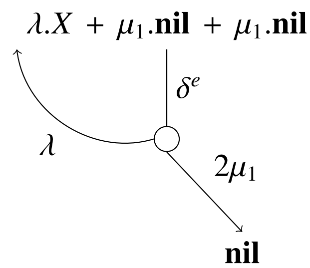



Figure 3: A CTMC

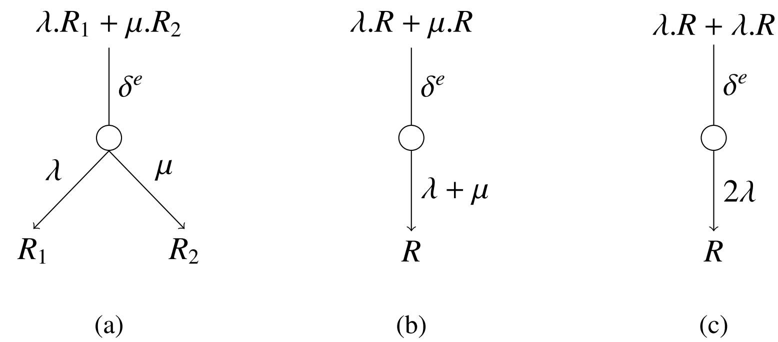



Figure 4: FuTS and race condition.

- if $R_1 = R_2$ and $\lambda = \mu$ then $\lambda.R + \lambda.R \overset{\delta^e}{\rightarrowtail} [R \mapsto 2\lambda]$.

The following proposition guarantees that the considered operational semantics is well-defined and associates to each state a finite-support total function in $\mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$.

**Proposition 4.1** *For all $P \in \mathcal{P}_{CTMC}$ and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$, if $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}$ can be derived using the rules of Figure 2, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$.* □

The following theorem, a direct consequence of the above proposition, characterizes the structure of $\mathcal{R}_{CTMC}$:

**Theorem 4.2** *$\mathcal{R}_{CTMC}$ is total and deterministic.* □

As a corollary of Theorem 4.2 we obtain that whenever $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}$, the exit rate of $P$ is given by $\oplus\mathscr{P}$ and $\mathscr{P}$ is the row of the rate matrix corresponding to $P$.

## 5 Parallel Composition

In this section we introduce some further notation and the basic ingredients necessary for dealing with process parallel composition. They will be used in the remaining sections, where major SPCs will be addressed and their specific design choices will be discussed.

A key issue to face when dealing with parallel composition operators, is the combination of the rates corresponding to the individual components contributing to a synchronization; the intuition being that a synchronization should last at least as the slowest of the synchronizing actions. A plausible choice could then be to take the maximum of the RVs associated to the involved actions. It is worth pointing that a plausible choice would be one according to which the RV associated to an action resulting from synchronization could be the MAX of the RVs.

Unfortunately, the MAX of two or more exponentially distributed RVs is not exponentially distributed. To remain in the realm of exponential distributions, different proposal have been put forward in the literature. In PEPA, the synchronization rate is the minimum of the rates of the synchronizing components according to the intuition that the rate of the slowest synchronizing action is the rate of the synchronization. Different approaches have been proposed in TIPP and EMPA. In TIPP, assuming that cooperating activities can boost each other, the synchronization rate is obtained as the product of those of the ones of the synchronizing actions. In EMPA, instead, a distinction is introduced between active and passive actions. A synchronization may take place only between a single active action and one or more passive actions. The synchronization rate is that of the active action.

Let us consider a generic process language $\mathcal{P}_C$ which provides a process parallel composition operator, denoted by, say, $\times$. The reachable states of $P_1 \times P_2$ are obtained via a suitable composition of $P_1$, $P_2$, plus the states reachable from $P_1$, and the states reachable from $P_2$. For instance, if $\times$ is the *interleaving* operator then the continuation functions of $P_1 \times P_2$ on $\alpha$-labelled transitions are obtained by composing the $\alpha$-continuations of $P_1$ (respectively $P_2$) in parallel with $P_2$ (respectively $P_1$).

Since in FuTS continuations are identified by functions in $\mathbf{FTF}(\mathcal{P}_C, \mathbb{C})$, the specific mechanism used by the various process calculi to compute synchronization rates can be expressed in terms of operators on $\mathbf{FTF}(\mathcal{P}_C, \mathbb{C})$. To provide a uniform description of the stochastic semantics of parallel composition, and in general of the considered SPCs, we introduce a set of basic operators that will be composed to capture the specific semantics of the operators of each SPC.

**Parallel aggregation** Let $\times$ be a parallel composition operator of a generic process language $\mathcal{P}_C$, and let $\mathscr{P}$ and $\mathscr{Q}$ be functions in $\mathbf{FTF}(\mathcal{P}_C, \mathbb{C})$, $\mathscr{P} \otimes^{\mathbb{C}}_{\times} \mathscr{Q}$ denotes the *parallel aggregation* of $\mathscr{P}$ and $\mathscr{Q}$ according to $\times$. This operator is used to obtain a function associating $(\mathscr{P} s_1) \cdot_{\mathbb{C}} (\mathscr{Q} s_2)$ to each term of the form $s_1 \times s_2$ and $0_{\mathbb{C}}$ to all the others.

Notice that $\otimes^{\mathbb{C}}_{\times} : \mathbf{TF}(S, \mathbb{C}) \to \mathbf{TF}(S, \mathbb{C}) \to \mathbf{TF}(S, \mathbb{C})$ is parametric with respect to the specific parallel operator used in the algebra.

The general definition of $\_ \otimes^{\mathbb{C}}_{\times} \_$ is the following:

$$(\mathscr{P} \otimes^{\mathbb{C}}_{\times} \mathscr{Q})\, s \;=_{\text{def}} \begin{cases} (\mathscr{P}\, s_1) \cdot_{\mathbb{C}} (\mathscr{Q}\, s_2), & \text{if } \exists s_1, s_2 \in S.\ s = s_1 \times s_2 \\ 0_{\mathbb{C}}, & \text{otherwise} \end{cases}$$

In the sequel we will use $\otimes^{\mathbb{C}}_{\|_L}$ and $\otimes^{\mathbb{C}}_{|}$ when multi-parties ( $\|_L$ ) and binary composition ($|$) are used, respectively. Moreover, we will omit the semi-ring $\mathbb{C}$ whenever clear from the context. Notice moreover that injectivity of $\times$ is essential in the above definition.

**Renormalization** Parallel aggregation combines functions to describe behaviour of cooperating processes, however, to compute the rate associated to specific transitions, the aggregated values might have to be *renormalized*. Let $\mathscr{P}$ be a function in $\mathbf{TF}(S, \mathbb{C})$ and $x, y \in \mathbb{C}$, we let $\mathscr{P} \cdot \frac{x}{y}$ denote the function associating to each $s \in S$ $(\mathscr{P} s) \cdot_{\mathbb{C}} \frac{x}{y}$ when $y \neq 0_{\mathbb{C}}$ and $0_{\mathbb{C}}$ otherwise.

Function $\_ \cdot_{\mathbb{C}} \_/\_ : \mathbf{TF}(S, \mathbb{C}) \to \mathbb{C} \to \mathbb{C} \to S \to \mathbb{C}$ is formally defined as follows:

$$(\mathscr{P} \cdot_{\mathbb{C}} \frac{x}{y})\, s \;=_{\text{def}} \begin{cases} (\mathscr{P}\, s) \cdot_{\mathbb{C}} (x/^{\mathbb{C}} y), & \text{if } y \neq 0_{\mathbb{C}} \\ 0_{\mathbb{C}}, & \text{otherwise} \end{cases}$$

**Characteristic functions** Let $\mathbb{C}$ be a commutative semi-ring and $S$ a set of states, we let $\mathcal{X}_{\mathbb{C}}$ be the function associating to each $s \in S$ function $[s \mapsto 1_{\mathbb{C}}]$ yielding value $1_{\mathbb{C}}$ in $s$ and $0_{\mathbb{C}}$ in all the other states in $S$.

The following proposition follows directrly from definition of $\mathcal{X}_{\mathbb{C}}$, $(\_ \otimes_{\times} \_)$, and $\_ \cdot_{\mathbb{C}} \_/\_$:

**Proposition 5.1** *For every countable non empty sets $S$ and commutative semi-ring $\mathbb{C}$,* $\mathbf{FTF}(S, \mathbb{C})$ *is* closed *under the operations $\mathcal{X}_{\mathbb{C}}$, $(\_ \otimes_{\times} \_)$, and $\_ \cdot_{\mathbb{C}} \_/\_$, i.e.*

- *for each* $s \in S$*,* $(\mathcal{X}_{\mathbb{C}}\ s) \in \mathbf{FTF}(S, \mathbb{C})$*,*
- *for each* $\mathscr{P}, \mathscr{Q} \in \mathbf{FTF}(S, \mathbb{C})$*,* $(\mathscr{P} \otimes_{\times} \mathscr{Q}) \in \mathbf{FTF}(S, \mathbb{C})$
- *for each* $\mathscr{P} \in \mathbf{FTF}(S, \mathbb{C})$ *and* $x, y \in \mathbb{C}$*,* $\mathscr{P} \cdot_{\mathbb{C}} x/y \in \mathbf{FTF}(S, \mathbb{C})$*.*

## 5.1 Parallel Composition of CTMCs

To show how the operators introduced above can be used, we extend the language of CTMC considered in Section 4 with the parallel operator $\parallel$, where $P_1 \parallel P_2$ identifies the *interleaving* between $P_1$ and $P_2$.

In a process term $P_1 \parallel P_2$, components $P_1$ and $P_2$ do not cooperate and the reachable states are those reachable from $P_1$ (respectively, $P_2$) composed in parallel with $P_2$ (respectively $P_1$). This means that if $P_1 \stackrel{\delta^e}{\rightarrowtail} \mathscr{P}$ and $P_2 \stackrel{\delta^e}{\rightarrowtail} \mathscr{Q}$, the states reachable from $P_1 \parallel P_2$ are obtained by *combining* $\mathscr{P}$ and $\mathscr{Q}$ respectively with $P_2$ and $P_1$ by using operators $\otimes_{\parallel}$ $\mathcal{X}_{\mathbb{R}}$ and introduced in the previous section:

- expression $\mathscr{P} \otimes_{\parallel} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P_2)$ characterizes all those states which are parallel compositions where the left-side component is a state reachable from $P_1$ and the right-side component is $P_2$ itself, and assigns to such states the same rate assigned to their left-side components, while
- expression $(\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P_1) \otimes_{\parallel} \mathscr{P}_2$ characterizes all those states which are parallel compositions where the left-side component is $P_1$ itself and the right-side component is a state reachable from $P_2$, and assigns to such states the same rate assigned to their right-side component;

these two expressions have to be summed up, in order to correctly compute the rates of all reachable states:

$$(\mathscr{P}_1 \otimes_{\parallel} (\mathcal{X}_{\mathbb{C}}\, P_2)) + ((\mathcal{X}_{\mathbb{C}}\, P_1) \otimes_{\parallel} \mathscr{P}_2). \tag{1}$$

In conclusion, the rule governing behaviour of parallel composed processes is the following:

$$(\text{PAR})\,\frac{P_1 \stackrel{\delta^e}{\rightarrowtail} \mathscr{P},\ P_2 \stackrel{\delta^e}{\rightarrowtail} \mathscr{Q}}{P_1 \parallel P_2 \stackrel{\delta^e}{\rightarrowtail} (\mathscr{P} \otimes_{\parallel} (\mathcal{X}_{\mathbb{R}}\, P_2)) + ((\mathcal{X}_{\mathbb{R}}\, P_1) \otimes_{\parallel} \mathscr{Q})}$$

For instance, if we consider term $\lambda_1.\mathbf{nil} \parallel \lambda_2.\mathbf{nil}$, then we have that:

$$\lambda_1.\mathbf{nil} \stackrel{\delta^e}{\rightarrowtail} [\mathbf{nil} \mapsto \lambda_1] \qquad \lambda_2.\mathbf{nil} \stackrel{\delta^e}{\rightarrowtail} [\mathbf{nil} \mapsto \lambda_2]$$

By applying rule (PAR) above, the following derivation can be proved:

$$\frac{\lambda_1.\mathbf{nil} \stackrel{\delta^e}{\rightarrowtail} [\mathbf{nil} \mapsto \lambda_1] \qquad \lambda_2.\mathbf{nil} \stackrel{\delta^e}{\rightarrowtail} [\mathbf{nil} \mapsto \lambda_2]}{\lambda_1.\mathbf{nil} \parallel \lambda_2.\mathbf{nil} \stackrel{\delta^e}{\rightarrowtail} [\mathbf{nil} \mapsto \lambda_1] \otimes_{\parallel} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\lambda_2.\mathbf{nil}) +_{\mathbb{R}_{\geq 0}} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\lambda_1.\mathbf{nil}) \otimes_{\parallel} [\mathbf{nil} \mapsto \lambda_2]}$$

where:

$$\begin{array}{rl} & [\mathbf{nil} \mapsto \lambda_1] \otimes_{\parallel} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\lambda_2.\mathbf{nil}) +_{\mathbb{R}_{\geq 0}} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\lambda_1.\mathbf{nil}) \otimes_{\parallel} [\mathbf{nil} \mapsto \lambda_2] \\ = & [\mathbf{nil} \mapsto \lambda_1] \otimes_{\parallel} [\lambda_2.\mathbf{nil} \mapsto 1_{\mathbb{R}_{\geq 0}}] +_{\mathbb{R}_{\geq 0}} [\lambda_1.\mathbf{nil} \mapsto 1_{\mathbb{R}_{\geq 0}}] \otimes_{\parallel} [\mathbf{nil} \mapsto \lambda_2] \\ = & [\mathbf{nil} \parallel \lambda_2.\mathbf{nil} \mapsto \lambda_1] +_{\mathbb{R}_{\geq 0}} [\lambda_1.\mathbf{nil} \parallel \mathbf{nil} \mapsto \lambda_2] \\ = & [\mathbf{nil} \parallel \lambda_2.\mathbf{nil} \mapsto \lambda_1\ ,\ \lambda_1.\mathbf{nil} \parallel \mathbf{nil} \mapsto \lambda_2] \end{array}$$

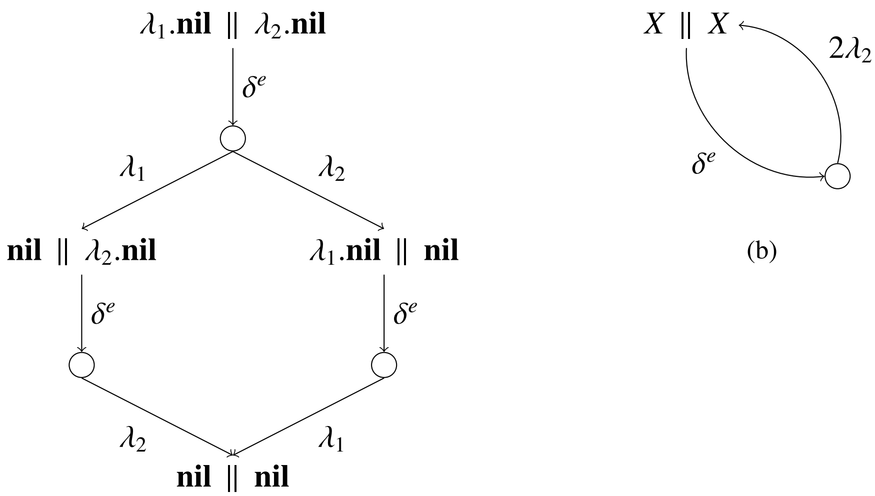
stating that term $\lambda_1.\mathbf{nil} \parallel \lambda_2.\mathbf{nil}$ can reach $\mathbf{nil} \parallel \lambda_2.\mathbf{nil}$ with rate $\lambda_1$ and $\lambda_1.\mathbf{nil} \parallel \mathbf{nil}$ with rate $\lambda_2$. The FuTS generated by $\lambda_1.\mathbf{nil} \parallel \lambda_2.\mathbf{nil}$ to term is reported in Figure 5 (a).

Let us now consider the term

$$X \parallel X \quad \text{where } X := \lambda.X$$

that represents a subtle situation where it is crucial to take into account all possible derivations. We have that $X \stackrel{\delta^e}{\rightarrowtail} [X \mapsto \lambda]$ and, by applying rule (PAR):

$$\frac{X \stackrel{\delta^e}{\rightarrowtail} [X \mapsto \lambda] \qquad X \stackrel{\delta^e}{\rightarrowtail} [X \mapsto \lambda]}{X \parallel X \stackrel{\delta^e}{\rightarrowtail} [X \mapsto \lambda] \otimes_{\parallel} (\mathcal{X}_{\mathbb{R}_{\geq 0}}X) +_{\mathbb{R}_{\geq 0}} (\mathcal{X}_{\mathbb{R}_{\geq 0}}X) \otimes_{\parallel} [X \mapsto \lambda]}$$




(a)

(b)

Figure 5: Parallel composition of CTMCs.

and, with derivations similar to the above ones, we have $X \parallel X \overset{\delta^e}{\rightarrowtail} [X \parallel X \mapsto 2\lambda]$. Thus, while according to the standard semantics of process algebras $X \parallel X$ would be considered equivalent to $X$, in SPCs $X \parallel X$ is modelled as twice faster than $X$. The use of summation to compose the next state functions associated to each of the two parallel components, guarantees the correct calculation of the rates of all transitions. The FuTS associated to $X \parallel X$ is shown in Figure 5 (b).

# 6 Fully Markovian CSP-based Calculi

In this section we focus on the SPCs with multi-party synchronization, namely TIPP, EMPA, and PEPA. We address only their fragments which are relevant for stochastic behavior, ignoring operators like *relabeling* or *hiding*. We will call such fragments $\text{TIPP}_k$, $\text{EMPA}_k$, and $\text{PEPA}_k$, the subscript $k$ standing for *kernel*. The relevant commutative semi-ring will be $\mathbb{R}_{\geq 0}$.

## 6.1 $\text{TIPP}_k$

The kernel we consider referres to the version of TIPP presented in [23][4], in Appendix B.2 we recall the original SOS for the fragment $\text{TIPP}_k$ we focus on. The considered operators are:

- *inaction*,
- *rated-action prefix*,
- *choice*,
- *multi-party synchronization*, and
- *constant*.

Consequently, the set $\mathcal{P}_{TIPP_k}$ of $\text{TIPP}_k$ terms is defined by the grammar obtained by selecting from Figure 1 the specific productions for the above operators.

[4] Here, synchronization rates are defined as the product of those of the synchronizing actions, as opposed to the original definition of TIPP, given in [19], where, instead, such rates are the MAX of the rates of the components.

$$\text{(RAPF1)}\ \frac{}{\langle a,\lambda\rangle.P \overset{\delta_a^e}{\rightarrowtail} [P\mapsto\lambda]} \qquad \text{(RAPF2)}\ \frac{\alpha\neq\delta_a^e}{\langle a,\lambda\rangle.P \overset{\alpha}{\rightarrowtail} []_{\mathbb{R}_{\geq 0}}}$$

$$\text{(PAR1)}\ \frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}, (n\,\alpha)\notin L}{P \parallel_L Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes_{\parallel_L} (\mathcal{X}_{\mathbb{R}}\, Q)) + ((\mathcal{X}_{\mathbb{R}}\, P) \otimes_{\parallel_L} \mathscr{Q})} \qquad \text{(PAR2}_\text{T}\text{)}\ \frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}, (n\,\alpha)\in L}{P \parallel_L Q \overset{\alpha}{\rightarrowtail} \mathscr{P} \otimes_{\parallel_L} \mathscr{Q}}$$

Figure 6: Additional semantics rules for $\text{TIPP}_k$

In TIPP, time delays are not separated from actions and, in fact, the race condition principle governs the choice between the initial actions of processes composed using the $+$ operator. $\mathcal{P}_{TIPP_k}$ is the set of states, $\mathcal{L}_{TIPP_k}$ is the set $\Delta_{\mathcal{A}}$ introduced in Section 2, while the transition relation $\rightarrowtail$ is the one induced by the rules (NIL), (CHO) and (CNS) of Figure 2, and the rules of Figure 6, where function $n$ associates to a transition label its action ($n : \mathcal{L}_{TIPP_k} \to \mathcal{A}$, with $n\,\delta_a^e \ =_{\text{def}}\ a$) while functions $\mathcal{X}_{\mathbb{R}}$ and $\otimes^{\mathbb{R}}_{\parallel_L}$ are those introduced in the previous section.

Rule (PAR1) for interleaving ensures that all behaviours of $P \parallel_L Q$ after $\alpha$, whenever $(n\,\alpha) \notin L$, are either of the form $R \parallel_L Q$ where $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ and $(\mathscr{P}\,R) > 0$, for some $\mathscr{P}$, or of the form $P \parallel_L R$ where $Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}$ and $(\mathscr{Q}\,R) > 0$, for some $\mathscr{Q}$. Rule ($\text{PAR2}_\text{T}$) for synchronization, instead, implements the *rate multiplication* principle of TIPP: if $(n\,\alpha) \in L$, $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$, $Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}$, $(\mathscr{P}\,R_P) = \lambda_P > 0$, and $(\mathscr{Q}\,R_Q) = \lambda_Q > 0$, then $P \parallel_L Q$ evolves, via $\alpha$, to $R_P \parallel_L R_Q$ with rate $\lambda_P \cdot \lambda_Q$.

The following proposition can be easily proven by derivation induction:

**Proposition 6.1** *For all $P \in \mathcal{P}_{TIPP_k}$, $\alpha \in \mathcal{L}_{TIPP_k}$ and $\mathscr{P} \in$ **TF**$(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using rules (NIL), (CHO) and (CNS) of Figure 2, plus those of Figure 6, then $\mathscr{P} \in$ **FTF**$(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$.* □

**Definition 6.1** *The formal semantics of $\text{TIPP}_k$ is the* $\text{FuTS}_{FS}$

$$\mathcal{R}_{TIPP_k} \ =_{\text{def}}\ (\mathcal{P}_{TIPP_k}, \mathcal{L}_{TIPP_k}, \mathbb{R}_{\geq 0}, \rightarrowtail)$$

*where $\rightarrowtail \subseteq \mathcal{P}_{TIPP_k} \times \mathcal{L}_{TIPP_k} \times$ **FTF**$(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$ is the least relation induced by rules (NIL), (CHO) and (CNS) of Figure 2 and by the rules in Figure 6.* •

The following theorem characterizes the structure of $\mathcal{R}_{TIPP_k}$.

**Theorem 6.2** *$\mathcal{R}_{TIPP_k}$ is total and deterministic.* □

**Theorem 6.3** FuTS *semantics of $\text{TIPP}_k$ coincides with the one in [23], i.e. for all $P, Q \in \mathcal{P}_{TIPP_k}$, $\alpha \in \mathcal{L}_{TIPP_k}$, and unique $\mathscr{P} \in$ **FTF**$(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$ such that $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$: $(\mathscr{P}\,Q) = \mathbf{rt}_{(n\,\alpha)}(P, Q)$, where $\mathbf{rt}_{(n\,\alpha)}(P, Q)$ is computed over the LTS characterized by the $\text{TIPP}_k$ SOS semantics.*

A CTMC can be easily derived for each $P \in \mathcal{P}_{TIPP_k}$. The set of its states is $(\mathcal{P}_{TIPP_k})_{P,\mathcal{L}_{TIPP_k}}$, while the rate matrix is defined as follows:

$$\mathbf{R}\,Q_1\,Q_2 \ =_{\text{def}} \sum_{Q_1 \overset{\alpha}{\rightarrowtail}_{P,\mathcal{L}_{TIPP_k}} \mathscr{P}} (\mathscr{P}\,Q_2)$$

Notice that, by (syntactical) construction, for all $P \in \mathcal{P}_{TIPP_k}$ the set $\{\alpha \mid \exists \mathscr{P}.\ P \overset{\alpha}{\rightarrowtail} \mathscr{P}\}$ is finite; thus the above sum always converges.

$$\text{(PAPF1)}\frac{}{\langle a,*_\omega\rangle.P \stackrel{\delta^e_{a*}}{\rightarrowtail} [P\mapsto\omega]} \qquad \text{(PAPF2)}\frac{\alpha\neq\delta^e_{a*}}{\langle a,*_\omega\rangle.P \stackrel{\alpha}{\rightarrowtail} []_{\mathbb{R}_{\geq 0}}}$$

$$\text{(PAR2}_{\text{E}}\text{)}\frac{P \stackrel{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}, Q \stackrel{\delta^e_{a*}}{\rightarrowtail} \mathscr{Q}, a\in L}{P \,\|_L\, Q \stackrel{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}\otimes_{\|_L} \mathscr{Q}\cdot\frac{(\oplus\mathscr{P})+(\oplus\mathscr{Q})}{(\oplus\mathscr{P})\cdot(\oplus\mathscr{Q})}}$$

$$\text{(PAR3}_{\text{E}}\text{)}\frac{P \stackrel{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_o, P \stackrel{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}_i, Q \stackrel{\delta^e_{a*}}{\rightarrowtail} \mathscr{Q}_i, Q \stackrel{\delta^e_{a}}{\rightarrowtail} \mathscr{Q}_o, a\in L}{P \,\|_L\, Q \stackrel{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_o\otimes_{\|_L} \mathscr{Q}_i\cdot\frac{1}{\oplus\mathscr{Q}_i} + \mathscr{P}_i\otimes_{\|_L} \mathscr{Q}_o\cdot\frac{1}{\oplus\mathscr{P}_i}}$$

Figure 7: Additional semantics rules for $\text{EMPA}_k$

## 6.2 $\text{EMPA}_k$

In this section we consider EMPA [1], but restrict our attention to the features of the exponentially timed kernel of EMPA and do not address other features of EMPA such as priorities, probabilities and immediate actions. In Appendix B.3 we recall the SOS for the fragment $\text{EMPA}_k$ we focus on. The considered operators are:

- *inaction*,
- *rated-action prefix*,
- *passive-action prefix*,
- *choice*,
- *multi-party synchronization*, and
- *constant*.

The set $\mathcal{P}_{EMPA_k}$ of $\text{EMPA}_k$ terms is induced by the grammar obtained by selecting from Figure 1 the specific productions for the above operators, where $\omega \in \mathcal{W}_{EMPA} \;=_{\text{def}}\; \mathbb{R}_{>0}$.

Similarly to TIPP, EMPA associates delays to actions. The label set $\mathcal{L}_{EMPA_k}$ includes set $\Delta_{\mathcal{A}}$. Moreover, to model EMPA interactions that forbid synchronization between active actions, we let $\mathcal{L}_{EMPA_k}$ include labels explicitly indicating execution of passive actions. We let $\delta^e_{a*}$ denote the execution of passive action $a$ and let $\Delta_{\mathcal{A}*}$ be $\{\delta^e_{a*} \mid a \in \mathcal{A}\}$. Thus we have $\mathcal{L}_{EMPA_k} \;=_{\text{def}}\; \Delta_{\mathcal{A}} \cup \Delta_{\mathcal{A}*}$ and, like in TIPP, we use function $n : \mathcal{L}_{EMPA_k} \to \mathcal{A}$ to obtain the action involved in the actual transition. The transition relation $\rightarrowtail$ is the one induced by rules (NIL), (CHO) and (CNS) of Figure 2, rules (RAPF1), (RAPF2) and (PAR1) of Figure 6, and the rules of Figure 7.

In this SPC, each synchronization is obtained as the interaction of a single *active action* with a set of *passive* ones. The rate of the synchronization is that of the active action; passive actions are equipped with *weights*

Rules ($\text{PAPF1}_{\text{E}}$) and ($\text{PAPF2}_{\text{E}}$) are self explanatory. Rule (PAR1) for interleaving ensures that all behaviors of $P \;\|_L\; Q$, after $\alpha$, whenever $(n\,\alpha) \notin L$, are of the form $R \;\|_L\; Q$ where $P \stackrel{\alpha}{\rightarrowtail} \mathscr{P}$ and $(\mathscr{P}\,R) > 0$, for some $\mathscr{P}$, or of the form $P \;\|_L\; R$ where $Q \stackrel{\alpha}{\rightarrowtail} \mathscr{Q}$ and $(\mathscr{Q}\,R) > 0$, for some $\mathscr{Q}$. Notice that $\alpha$ can also be a delay of a passive action, $\delta^e_{a*}$ for some $a$, in which case $\mathscr{P}$ or $\mathscr{Q}$ yields weights. Rule ($\text{PAR2}_{\text{E}}$) models the "passive side" of EMPA's *asymmetry* principle for synchronization: for $a \in L$, if $P \stackrel{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}$, $Q \stackrel{\delta^e_{a*}}{\rightarrowtail} \mathscr{Q}$, $(\mathscr{P}\,R_P) = \omega_P > 0$, and $(\mathscr{Q}\,R_Q) = \omega_Q > 0$, then $P \;\|_L\; Q$ evolves to $R_P \;\|_L\; R_Q$ with weight $\omega_P \cdot \omega_Q \cdot \frac{(\oplus\mathscr{P})+(\oplus\mathscr{Q})}{(\oplus\mathscr{P})\cdot(\oplus\mathscr{Q})}$, under the assumption that the total weight of $a$ in $P$ (*i.e.* $\oplus\,\mathscr{P}$) and the total weight of $a$ in $Q$ (*i.e.* $\oplus\,\mathscr{Q}$) are positive; otherwise $P \;\|_L\; Q$ leads to $[]_{\mathbb{R}_{\geq 0}}$ via $\delta^e_{a*}$. The normalization factor $\frac{(\oplus\mathscr{P})+(\oplus\mathscr{Q})}{(\oplus\mathscr{P})\cdot(\oplus\mathscr{Q})}$ is chosen, in EMPA, in such a way that the total weight of $a$ in $P \;\|_L\; Q$ is indeed $(\oplus\mathscr{P}) + (\oplus\mathscr{Q})$.

The second rule for synchronization, ($\mathrm{PAR3_E}$), implements the asymmetry principle of EMPA: the transitions concerning the active role of $a$ in $P$, i.e. $P \overset{\delta_a^e}{\rightarrowtail} \mathscr{P}_o$, are paired with the transitions concerning the "passive role" of $a$ in $Q$, i.e. $Q \overset{\delta_{a*}^e}{\rightarrowtail} \mathscr{Q}_i$, and the resulting function $\mathscr{P}_o \;\; \|_L \;\; \mathscr{Q}_i$ is normalized with the positive weight of $a$ in $Q$, (i.e. $\oplus\mathscr{Q}_i$), and vice-versa.

The following proposition can be easily proven by derivation induction:

**Proposition 6.4** *For all $P \in \mathcal{P}_{EMPA_k}$, $\alpha \in \mathcal{L}_{EMPA_k}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules consisting of rules (NIL), (CHO) and (CNS) of Figure 2 plus rules (RAPF1) and (RAPF2) of Figure 6 and the rules in Figure 7, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$.* □

**Definition 6.2** *The formal semantics of EMPA$_k$ is the* FuTS$_{FS}$

$$\mathcal{R}_{EMPA_k} \;=_{\mathrm{def}}\; (\mathcal{P}_{EMPA_k}, \mathcal{L}_{EMPA_k}, \mathbb{R}_{\geq 0}, \rightarrowtail)$$

*where the transition relation $\rightarrowtail \subseteq \mathcal{P}_{EMPA_k} \times \mathcal{L}_{EMPA_k} \times \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ is the least relation satisfying the set of rules consisting only of the rules (NIL), (CHO) and (CNS) of Figure 2 plus rules (RAPF1) and (RAPF2) of Figure 6 and the rules of Figure 7.* •

**Theorem 6.5** *$\mathcal{R}_{EMPA_k}$ is total and deterministic.*

**Theorem 6.6** FuTS *semantics of EMPA$_k$ coincides with the semantics given in [1], i.e. for all $P, Q \in \mathcal{P}_{EMPA_k}$, $\delta_a^e, \delta_{a*}^e \in \mathcal{L}_{EMPA_k}$, and unique functions $\mathscr{P}, \mathscr{P}' \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ such that $P \overset{\delta_a^e}{\rightarrowtail} \mathscr{P}$ and $P \overset{\delta_{a*}^e}{\rightarrowtail} \mathscr{P}'$, the following holds: $(\mathscr{P}\, Q) = \mathbf{rt}_a(P, Q)$, $(\mathscr{P}'\, Q) = \mathbf{wt}_a(P, Q)$, and $(\oplus\mathscr{P}') = weight(P, a)$, where $\mathbf{rt}_a(P, Q)$, $\mathbf{wt}_a(P, Q)$, and $weight(P, a)$ are computed over the LTS characterized by the EMPA$_k$ SOS semantics.* □

The CTMC associated to $P \in \mathcal{P}_{EMPA_k}$ is built by considering only the transitions associated to active actions. Consequently, the set of states is $(\mathcal{P}_{EMPA_k})_{P,\Delta_{\mathcal{A}}}$, while the rate matrix is defined as follows:

$$\mathbf{R}\, Q_1\, Q_2 \;=_{\mathrm{def}} \sum_{Q_1 \overset{\alpha}{\rightarrowtail}_{P,\Delta_{\mathcal{A}}} \mathscr{P}} (\mathscr{P}\, Q_2)$$

We close this section by noting that the original syntax of EMPA contains also *immediate actions*, i.e. actions with no durations. We will show how to deal with this kind of actions when we consider the language IML (see Section 8) that clearly separates non-determinism and time and hence all its actions are untimed.

## 6.3 PEPA$_k$

We consider now the kernel calculus PEPA$_k$ of PEPA [26], consisting of

- *rated-action prefix*,
- *choice composition*,
- *multi-party synchronization*, and
- *constant*.

The principle regulating the synchronization rate of PEPA processes is the so called *minimal rate*, where, essentially, the rate of an action which is the result of the synchronization of two component processes is the MIN of the rates of synchronizing actions. Whenever a component process may perform the same action in several different ways, the cumulative, so called *apparent*, rate has to be considered when modelling compositions. The kernel we consider in this section is adequate for illustrating the *minimal apparent rate* principle; therefore, we leave out other features of PEPA like hiding and relabeling. In Appendix B.3 we recall the original SOS for the fragment PEPA$_k$ we focus on.

$$(\text{PAR2}_\text{P})\frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}, (n\,\alpha) \in L}{P \,\|_L\, Q \overset{\alpha}{\rightarrowtail} \mathscr{P} \,\|_L\, \mathscr{Q} \cdot \frac{\text{MIN}\{\oplus\mathscr{P}, \oplus\mathscr{Q}\}}{\oplus\mathscr{P} \cdot \oplus\mathscr{Q}}}$$

Figure 8: Synchronization rule for $\text{PEPA}_k$

The set $\mathcal{P}_{PEPA_k}$ of $\text{PEPA}_k$ terms is defined by the grammar obtained by selecting from Figure 1 the specific productions for the above mentioned operators.

Also in PEPA delays are associated to actions. Consequently, we let the label set $\mathcal{L}_{PEPA_k}$ be again the set $\Delta_{\mathcal{A}}$, ranged over by $\alpha, \alpha_1, \alpha', \ldots$; function $n$ for PEPA is defined as expected: $n : \mathcal{L}_{PEPA_k} \to \mathcal{A}$ with $n\,\delta^e_a \;=_{\text{def}}\; a$. The relevant set of states is $\mathcal{P}_{PEPA_k}$ while the set of rules defining the transition relation $\rightarrowtail$ is composed of rules (CHO) and (CNS) of Figure 2, (RAPF1), (RAPF2) and (PAR1) of Figure 6 and the rule of Figure 8.

Rule ($\text{PAR2}_\text{P}$) for cooperation implements the *minimal apparent rate* principle of PEPA: if $(n\,\alpha) \in L$, $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$, $Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}$, $(\mathscr{P}\, R_P) = \lambda_P > 0$, and $(\mathscr{Q}\, R_Q) = \lambda_Q > 0$, then $P \;\|_L\; Q$ evolves to $R_P \;\|_L\; R_Q$ with rate $\frac{\lambda_P}{\oplus\mathscr{P}} \cdot \frac{\lambda_Q}{\oplus\mathscr{Q}} \cdot \text{MIN}\{\oplus\mathscr{P}, \oplus\mathscr{Q}\}$.

The following proposition can be easily proven by derivation induction:

**Proposition 6.7** *For all $P \in \mathcal{P}_{PEPA_k}$, $\alpha \in \mathcal{L}_{PEPA_k}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules composed only of rules (CHO) and (CNS) of Figure 2, rules (RAPF1), (RAPF2) and (PAR1) of Figure 6 and rule ($\text{PAR2}_P$) of Figure 8, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$.* □

**Definition 6.3** *The formal semantics of $\text{PEPA}_k$ is the* $\text{FuTS}_{FS}$

$$\mathcal{R}_{PEPA_k} \;=_{\text{def}}\; (\mathcal{P}_{PEPA_k}, \mathcal{L}_{PEPA_k}, \mathbb{R}_{\geq 0}, \rightarrowtail)$$

*where $\rightarrowtail \subseteq \mathcal{P}_{PEPA_k} \times \mathcal{L}_{PEPA_k} \times \mathbf{FTF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$ is the least relation satisfying the set of rules consisting only of rules (CHO) and (CNS) of Figure 2, (RAPF1), (RAPF2) and (PAR1) of Figure 6 and rule ($\text{PAR2}_P$) of Figure 8.* •

**Theorem 6.8** *$\mathcal{R}_{PEPA_k}$ is total and deterministic.*

As a corollary of Theorem 6.8, we get that whenever $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$, the apparent rate of $\alpha$ in $P$, i.e. the exit rate, $r_\alpha(P)$, of $P$ relative to $\alpha$, is given by $\oplus\mathscr{P}$.

**Theorem 6.9** FuTS *semantics of $\text{PEPA}_k$ coincides with the semantics given in [26], i.e. for all $P, Q \in \mathcal{P}_{PEPA_k}$, $\alpha \in \mathcal{L}_{PEPA_k}$, and unique $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$ such that $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ the following holds: $(\mathscr{P}\, Q) = \mathbf{rt}_{(n\,\alpha)}(P, Q)$ where $\mathbf{rt}_{(n\,\alpha)}(P, Q)$ is computed over the LTS characterized by the $\text{PEPA}_k$ SOS semantics.* □

A CTMC can be associated to each $P \in \mathcal{P}_{PEPA_k}$ in a similar way as for $\text{TIPP}_k$.

We close this section by observing that *PEPA passive actions* [26] can be easily dealt with also in the FuTS approach. One way for doing this is to proceed as in [26], considering functions in $\mathbf{TF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0} \cup \{*_\omega \mid \omega \in \mathbb{N}_{>0}\})$, using the equations for weights[5] introduced in [26]. For details we refer the reader to [15].

# 7 Fully Markovian CCS-based Calculi

The SPCs that have been considered in previous sections all rely on an operator for multi-party (n-ary) parallel composition borrowed from Hoare's CSP [27]. In this section, we shall instead consider stochastic extensions of CCS [32] that, instead, make use of binary parallel composition. A synchronization between processes $P$ and $Q$ running in parallel occurs when $P$ *sends* a signal over a channel (action $\bar{a}$) and $Q$ *receives* a signal over the same channel (action $a$). While there have been many variants of stochastic calculi based on the CSP interaction paradigm, very few proposals have been put forward for the CCS

[5] $*_{\omega_1} + *_{\omega_2} = *_{\omega_1+\omega_2}$, $\frac{*_{\omega_1}}{*_{\omega_2}} = \frac{\omega_1}{\omega_2}$. Moreover, weights are ordered as follows: $x < *_\omega$ $(\forall x \in \mathbb{R}_{>0})$, $*_{\omega_1} < *_{\omega_2}$ if $\omega_1 < \omega_2$.

based one. Moreover, all of them are inspired by [34], that introduces a stochastic extension of $\pi$-calculus, a calculus for mobility that generalizes CCS and guarantees a sophisticated handling of channel names and their visibility. In this section, we consider two stochastic extensions of CCS that in general terms take inspiration from the approach presented in [34].

The first variant, named StoCCS$_{AA}$, assignes an *active* role to both input and output actions. Following the same approach used in [29], we consider two alternatives ways of computing the rate of a binary synchronization. First, like in TIPP, the rate of a synchronization is obtained as the multiplication of the rates of the involved input and output actions. Then, the rate of a synchronization is computed *a-là* PEPA, like in [34] for the $\pi$-calculus, and is obtained as the minimum between the total input and the total output rates over the same channel. We shall see that in the second case associativity of parallel composition is lost. Indeed, in [29] it is shown that associativity of CCS parallel composition can be guaranteed only when the multiplicative approach is used.

The second variant, named StoCCS$_{AP}$, follows an approach similar to the calculus EMPA that we considered in Section 6.2: Is is assumed that *output actions* have an *active* role while *input actions* are *passive*. The rate of a synchronization is then the one of the involved output action. This simple stochastic extension permits highlighting some of the intricacies related to stochastic extensions of CCS-like calculi. We will see that differently from the multiparty approach, where synchronizations have a *local* meaning, in the binary approach, synchronizations play a *global role*. This means that, to guarantee expected properties like associativity of parallel composition, re-normalizations of computed synchronization rates have to be performed.

We will conclude this section with a discussion about associativity of parallel binary composition when both input and output actions are considered active and the minimum rate principle is used. It will be shown that, by paying the price of additional intricacies an associative binary parallel composition operator implementing the minimum rate principle can be defined. This somehow contradicts the result presented in [29] but we shall discuss the additional ingredients that permit overcoming the obstacles.

In the context of stochastic extensions of CCS-like languages, *proved transitions* are used; transitions are labelled also with an encoding of the derivation which provides a proof of the transition in a SOS deduction system and *uniquely* identifies the transition within the set of those coming out of a state. Letting $\theta, \theta'$ be proof encodings, we define

$$\mathbf{rt}_a(P_1, P_2) \ =_{\text{def}} \ \sum\{\lambda \mid P_1 \xrightarrow{\theta a, \lambda} P_2\}$$

for $a \in \mathcal{A} \cup \bar{\mathcal{A}}$, and

$$\mathbf{rt}_{\langle a \,\|\, \bar{a}\rangle}(P_1, P_2) \ =_{\text{def}} \ \sum\{\lambda \mid P_1 \xrightarrow{\langle \theta a, \theta' \bar{a}\rangle, \lambda} P_2\}.$$

## 7.1 Active-active synchronization

In this section we consider StoCCS$_{AA}$ and the two variants for calculating synchronization rates. The syntax of the language is obtained by considering the following operators:

- *inaction*,
- *rated-output-action prefix*,
- *rated-input-action prefix*,
- *choice*, and
- *binary synchronization*.

Consequently, the set $\mathcal{P}_{StoCCS_{AA}}$ of StoCCS$_{AA}$ terms is defined by the grammar obtained by selecting from Figure 1 the productions specific of the above mentioned operators.

We let the label set $\mathcal{L}_{StoCCS_{AA}}$ be defined as the set $\Delta_{\mathcal{A}} \cup \Delta_{\bar{\mathcal{A}}} \cup \Delta_{\hat{\mathcal{A}}}$, where $\Delta_{\bar{\mathcal{A}}}$ and $\Delta_{\hat{\mathcal{A}}}$ are the sets $\{\delta^e_{a} \mid a \in \bar{\mathcal{A}}\}$ and $\{\delta^e_{\hat{a}} \mid a \in \mathcal{A}\}$, respectively. The latter denote channel synchronizations. Please notice that, while in CCS the result of a synchronization is labeled as $\tau$, here the synchronization of two StoCCS$_{AA}$

$$(\text{IN1}_A)\frac{}{a^\lambda.P \overset{\delta^e_a}{\rightarrowtail} [P\mapsto\lambda]} \qquad (\text{IN2}_A)\frac{\alpha \neq \delta^e_a}{a^\lambda.P \overset{\alpha}{\rightarrowtail} []_{\mathbb{R}_{\geq 0}}}$$

$$(\text{OUT1})\frac{}{\bar{a}^\lambda.P \overset{\delta^e_{\bar{a}}}{\rightarrowtail} [P\mapsto\lambda]} \qquad (\text{OUT2})\frac{\alpha \neq \delta^e_{\bar{a}}}{\bar{a}^\lambda.P \overset{\alpha}{\rightarrowtail} []_{\mathbb{R}_{\geq 0}}}$$

$$(\text{PAR1})\frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}, \alpha\neq\delta^e_{\hat{a}}}{P|Q \overset{\alpha}{\rightarrowtail} (\mathscr{P}\otimes_{|}(\mathcal{X}_{\mathbb{R}_{\geq 0}} Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P)\otimes_{|}\mathscr{Q})}$$

Figure 9: Additional semantic rules for $\text{StoCCS}_{AA}$

$$\frac{P \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{P}, P \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_i, P \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}_o, Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{Q}, Q \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{Q}_i, Q \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{Q}_o}{P|Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} (\mathscr{P}\otimes_{|}(\mathcal{X}_{\mathbb{R}_{\geq 0}} Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P)\otimes_{|}\mathscr{Q}) + \mathscr{P}_i\otimes_{|}\mathscr{Q}_o + \mathscr{P}_o\otimes_{|}\mathscr{Q}_i}$$

Figure 10: Rule ($\text{PAR}_{\text{AA-}*}$) for multiplicative synchronization for $\text{StoCCS}_{AA}$

processes over channel $a$ is labeled as $\hat{a}$. The additional information, i.e. the action name, is crucial for establishing associativity results.

The transition relation $\rightarrowtail$ is the one induced by rules (NIL) and (CHO) of Figure 2, rules reported in Figure 9, and one out of the two rules in Figure 10 and Figure 11. Rules ($\text{IN1}_\text{P}$) and ($\text{IN2}_\text{P}$) (respectively ($\text{OUT1}_\text{A}$) and ($\text{OUT2}_\text{A}$)) govern the behaviour of passive input actions (respectively active output actions), while rule (PAR1) describes interleaving behaviour of parallel processes. Notice that, rule (PAR1) can only be applied when involved label $\alpha$ is not a synchronization, i.e. when $\alpha \neq \delta^e_{\hat{a}}$.

Rules for synchronization deserve a few remarks. In the following we will define two different rules, named ($\text{PAR}_{\text{AA-}*}$) and ($\text{PAR}_{\text{AA-min}}$) that consider multiplicative rates and minimum apparent rate respectively.

In CCS we have two-party synchronizations, thus possible continuations of $P \mid Q$ after $\hat{a}$, i.e. after a synchronization over channel $a$, are:

1. the continuations of $P$ after $\hat{a}$, in parallel with $Q$;
2. the continuations of $Q$ after $\hat{a}$, in parallel with $P$;
3. the continuations of $P$ after $a$ in parallel with continuations of $Q$ after $\bar{a}$;
4. the continuations of $P$ after $\bar{a}$in parallel with the continuations of $Q$ after $a$.

Let continuation functions $\mathscr{P}$, $\mathscr{P}_i$ and $\mathscr{P}_o$ (respectively $\mathscr{Q}$, $\mathscr{Q}_i$ and $\mathscr{Q}_o$) be associated to $P$ (respectively $Q$) after a synchronization, an input and an output over channel $a$. When the multiplicative approach is used to compute rate of synchronization, continuations (1)-(4) described above are computed by using parallel aggregation ($\otimes_|$) and characteristic functions ($\mathcal{X}_\mathbb{R}$) introduced in Section 5:

1. ($\mathscr{P} \otimes_| (\mathcal{X}_{\mathbb{R}_{\geq 0}} Q)$), which characterizes the continuation of $P$, in parallel with $Q$;
2. (($\mathcal{X}_{\mathbb{R}_{\geq 0}} P) \otimes_| \mathscr{Q}$), which characterizes the continuation of $Q$, in parallel with $P$;
3. $\mathscr{P}_i \otimes_| \mathscr{Q}_o$, which is the continuation after composing inputs in $P$ with outputs in $Q$;
4. $\mathscr{P}_o \otimes_| \mathscr{Q}_i$, which is the continuation after composing outputs in $P$ with inputs in $Q$.

Notice that in the latter two cases, synchronization rates are obtained as the multiplication between the involved output and input rates. Finally, rule ($\text{PAR}_{\text{AA-}*}$) is shown in Figure 10.

A more complicated rule has to be used to handle synchronizations where the minimum rate principle is used. Let $\mathscr{P}_i$ be the continuation of $P$ after an input over $a$, and $\mathscr{Q}_o$ be the continuation of $Q$ after a $\bar{a}$, then the rate of a synchronization on channel $a$ between inputs in $P$ and outputs in $Q$ is obtained as the minimum between the total rate of inputs in $P$ ($\oplus\mathscr{P}_i$) and the total rate of outputs in $Q$ ($\oplus\mathscr{Q}_o$). The

$$\frac{P \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{P},\ P \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_i,\ P \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}_o,\ Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{Q},\ Q \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{Q}_i,\ Q \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{Q}_o}{P|Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} (\mathscr{P} \otimes_| (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_| \mathscr{Q}) + \mathscr{P}_i \otimes_| \mathscr{Q}_o \cdot \frac{\textsc{min}\{\oplus \mathscr{P}_i, \oplus \mathscr{Q}_o\}}{\oplus \mathscr{P}_i \cdot \oplus \mathscr{Q}_o} + \mathscr{P}_o \otimes_| \mathscr{Q}_i \cdot \frac{\textsc{min}\{\oplus \mathscr{P}_o, \oplus \mathscr{Q}_i\}}{\oplus \mathscr{P}_o \cdot \oplus \mathscr{Q}_i}}$$

Figure 11: Rule ($\mathrm{PAR}_{\text{AA-M}}$) for minimum rate synchronization for $\mathrm{StoCCS}_{AA}$

synchronizations between output in $P$ and input in $Q$ are dealt with similarly. A we know, a specific process $P'$ is reached from $P$, after input over $a$, with rate $(\mathscr{P}_i\, P')$; similarly, $Q'$ is reached from $Q$, after output $\bar{a}$, with rate $(\mathscr{Q}_o\, Q')$; thus the probability that such a specific interaction takes place is $\frac{(\mathscr{P}_i\, P')\cdot(\mathscr{Q}_o\, Q')}{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_o}$. Hence, the final synchronization rate is obtained as $\frac{(\mathscr{P}_i\, P')\cdot(\mathscr{Q}_o\, Q')}{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_o}\textsc{min}\{\oplus\mathscr{P}_i, \oplus\mathscr{Q}_o\}$. Notice that, $(\mathscr{P}_i\, P') \cdot (\mathscr{Q}_o\, Q')$ is the synchronization rate obtained according to the multiplicative rate approach. This means that, when the minimum rate principle is used, rule ($\mathrm{PAR}_{\text{AA-}*}$) has to be modified so to re-normalize $\mathscr{P}_i \otimes_| \mathscr{Q}_o$ and $\mathscr{P}_o \otimes_| \mathscr{Q}_i$ with $\frac{\textsc{min}\{\oplus\mathscr{P}_i, \oplus\mathscr{Q}_o\}}{\oplus\mathscr{P}_i \cdot \oplus\mathscr{Q}_o}$ and $\frac{\textsc{min}\{\oplus\mathscr{P}_o, \oplus\mathscr{Q}_i\}}{\oplus\mathscr{P}_o \cdot \oplus\mathscr{Q}_i}$ respectively. Rule ($\mathrm{PAR}_{\text{AA-M}}$) is formally defined in Figure 11. Notice that neither $P$ nor $Q$ plays any role in the synchronization occurring locally in the other process.

**Proposition 7.1** *For all $P \in \mathcal{P}_{StoCCS_{AA}}$, $\alpha \in \mathcal{L}_{StoCCS_{AA}}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0})$ if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules consisting of rules (NIL) and (CHO) of Figure 2, plus rules in Figure 9, and using one out of ($\mathrm{PAR}_{\text{AA-M}}$) and ($\mathrm{PAR}_{\text{AA-}*}$), then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0})$.* □

Below we define the two FuTS corresponding to the two variants of the $\mathrm{StoCCS}_{AA}$ when the multiplicative or the minimum rate synchronization approach is taken.

**Definition 7.1**

1. *We let $\mathcal{R}^*_{StoCCS_{AA}}$ be the* $\mathrm{FuTS}_{FS}$ *such that:*

$$\mathcal{R}^*_{StoCCS_{AA}} =_{\mathrm{def}} (\mathcal{P}_{StoCCS_{AA}}, \mathcal{L}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0}, \rightarrowtail)$$

   *where the transition relation $\rightarrowtail \subseteq \mathcal{P}_{StoCCS_{AA}} \times \mathcal{L}_{StoCCS_{AA}} \times \mathbf{FTF}(\mathcal{P}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0})$ is the least relation satisfying the set of rules consisting only of the rules (NIL) and (CHO) of Figure 2, the rules in Figure 9, and tule ($\mathrm{PAR}_{AA\text{-}*}$).*

2. *We let $\mathcal{R}^M_{StoCCS_{AA}}$ be the* $\mathrm{FuTS}_{FS}$ *such that*

$$\mathcal{R}^M_{StoCCS_{AA}} =_{\mathrm{def}} (\mathcal{P}_{StoCCS_{AA}}, \mathcal{L}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0}, \rightarrowtail)$$

   *where the transition relation $\rightarrowtail \subseteq \mathcal{P}_{StoCCS_{AA}} \times \mathcal{L}_{StoCCS_{AA}} \times \mathbf{FTF}(\mathcal{P}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0})$ is the least relation satisfying the set of rules consisting only of the rules (NIL) and (CHO) of Figure 2, the rules in Figure 9, and tule ($\mathrm{PAR}_{AA\text{-}M}$).*

•

**Theorem 7.2** *Both $\mathcal{R}^*_{StoCCS_{AA}}$ and $\mathcal{R}^M_{StoCCS_{AA}}$ are total and deterministic.*

The following theorem establishes the formal correspondence between the FuTS semantics of $\mathrm{StoCCS}_{AA}$ and the semantics definition given in [29].

**Theorem 7.3** *The two operational semantics of $StoCCS_{AA}$ induced by $\mathcal{R}^*_{StoCCS_{AA}}$ and $\mathcal{R}^M_{StoCCS_{AA}}$ coincide with those proposed in [29], i.e. in both multiplicative and minimum rate approach, for all $P, Q \in \mathcal{P}_{StoCCS_{AA}}$, $\alpha \in \mathcal{L}_{StoCCS_{AA}}$, and unique $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{StoCCS_I}, \mathbb{R}_{\geq 0})$ such that $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ the following holds:*

$$(\mathscr{P}\ Q) = \begin{cases} \mathbf{rt}_a(P, Q),\ \text{if } \alpha = \delta^e_a \text{ with } a \in \mathcal{A} \cup \bar{\mathcal{A}} \\ \\ \mathbf{rt}_{\langle a|\bar{a}\rangle}(P, Q) + \mathbf{rt}_{\langle \bar{a}|a\rangle}(P, Q),\ \text{if } \alpha = \delta^e_{\hat{a}} \text{ with } a \in \mathcal{A} \end{cases}$$

□

As pointed out in [29], Stochastic CCS with a *minimal rate*, PEPA-like approach, semantics suffers of non-associativity of CCS parallel composition with respect to strong Markovian bisimilarity[6]. Intuitively, the problem with parallel composition is that terms which differ only for the grouping of parallel components generate transitions which correspond to "the same" interactions but assign *different* rates to "the same" continuation behaviours.

As an example of non associativity (taken from [29]), let us consider $P = (P_1 \mid P_2) \mid P_3$ and $Q = P_1 \mid (P_2 \mid P_3)$ where $P_1 = a^{\lambda}.\mathbf{nil}$, $P_2 = a^{\lambda}.\mathbf{nil}$ and $P_3 = \bar{a}^{\lambda}.\mathbf{nil}$. We have that, process $P$ gives rise to a single $\delta^{e}_{\hat{a}}$-labelled transition leading to the continuation function:

$$[(P_1 \mid \mathbf{nil}) \mid \mathbf{nil} \mapsto \frac{\lambda}{2}, (\mathbf{nil} \mid P_2) \mid \mathbf{nil} \mapsto \frac{\lambda}{2}]$$

while process $Q$ gives rise to a single $\delta^{e}_{\hat{a}}$-labelled transition leading to the continuation function:

$$[P_1 \mid (\mathbf{nil} \mid \mathbf{nil}) \mapsto \lambda, \mathbf{nil} \mid (P_2 \mid \mathbf{nil}) \mapsto \lambda]$$

Clearly, process $P$ after an interaction between $P_2$ and $P_3$, reaches $(P_1 \mid \mathbf{nil}) \mid \mathbf{nil}$ with rate $\frac{\lambda}{2}$, while process $Q$ after the same interaction, reaches $P_1 \mid (\mathbf{nil} \mid \mathbf{nil})$, with rate $\lambda$. Thus, the *stochastic* behavior of $P$ and $Q$ is clearly different. The basic reason for the difference is the fact that in $Q$, differently from $P$, the rate of the synchronization between $P_2$ and $P_3$ is computed without taking into account the presence of the input action in $P_1$.

From the results in [29] it follows that it is impossible to define an SGSOS semantics that guarantees the associativity of CCS parallel composition. In the next section we will discuss how this problem could be overcome.

## 7.2 Active-passive synchronization

We introduce now StoCCS$_{AP}$, the stochastic extension of CCS where it is assumed that output actions have an *active* role while input actions are considered as *passive*. The duration of a synchronization is determined by the rate assigned to the participating output action. *Input actions* are annotated with *weights*, i.e. positive integers that are only used for determining the probability that a specific input is selected among the possible ones when a complementary output is executed. This approach is inspired by the notion of *passive actions* of EMPA discussed in Section 6.2.

The set $\mathcal{P}_{StoCCS_{AP}}$ of StoCCS$_{AP}$ terms are obtained by considering:

- *inaction*,
- *rated-output-action prefix*,
- *passive-input-action prefix*,
- *choice*,
- *binary synchronization*, and
- *constants*

Consequently, the set $\mathcal{P}_{StoCCS_{AP}}$ of StoCCS$_{AP}$ terms is defined by the grammar obtained by selecting from Figure 1 the productions specific of the above mentioned operators where the additional constraint is imposed that the two processes in a nondeterministic term of the form $P + Q$ cannot offer alternative input and output actions on the same channel. In other words, processes of the form $\bar{a}^{\lambda}.P_1 + a^{*\omega}.Q_1$ are not allowed. This is mainly for the sake simplicity; otherwise the computation of the synchronization rates would be more complicated because we would have to exclude from the computation input actions alternative to the active output ones.

The label set $\mathcal{L}_{StoCCS_{AP}}$ is the same as the one for StoCCS$_{AA}$, i.e. $\Delta_{\mathcal{A}} \cup \Delta_{\bar{\mathcal{A}}} \cup \Delta_{\hat{\mathcal{A}}}$. The transition relation $\rightarrowtail$ is the one induced by rules (NIL) and (CHO) of Figure 2, plus rules (OUT1), (OUT2) and (PAR1) in

[6] PEPA parallel composition/cooperation operator *is*, instead, associative.

$$(\text{IN1}_\text{P})\frac{}{a^{*\omega}.P \overset{\delta_a^e}{\rightarrowtail} [P \mapsto \omega]} \qquad (\text{IN2}_\text{P})\frac{\alpha \neq \delta_a^e}{a^{*\omega}.P \overset{\alpha}{\rightarrowtail} []_{\mathbb{N}_{\geq 0}}}$$

Figure 12: Semantic rules for $\text{StoCCS}_{AP}$

Figure 9, rules $(\text{IN1}_\text{P})$, $(\text{IN2}_\text{P})$ (modeling the behaviour of passive input actions) in Figure 12 plus the rule governing the synchronization that will be introduced below.

The rule for synchronization deserves more attention and specific motivations. The next states of $P \mid Q$ after $\hat{a}$, i.e. after a synchronization over channel $a$ has taken place, can be obtained by composing next state functions $\mathscr{P}$, $\mathscr{P}_i$ and $\mathscr{P}_o$ (respectively $\mathscr{Q}$, $\mathscr{Q}_i$ and $\mathscr{Q}_o$) associated to $P$ (respectively $Q$) after a synchronization, after an input and after an output over channel $a$ by using parallel aggregation ($\otimes_|$), renormalization ($\_ \cdot_\mathbb{R} \_/\_$) and characteristic functions ($\mathcal{X}_\mathbb{R}$) introduced in Section 5.

A straightforward implementation of the synchronization rule would take into account the following components when calculating the continuations of $P \mid Q$ after $\hat{a}$:

1. the continuations of $P$ after $\hat{a}$, in parallel with $Q$ ($\mathscr{P} \otimes_|^\mathbb{R} (\mathcal{X}_\mathbb{R}\, Q)$);
2. the continuations of $Q$ after $\hat{a}$, in parallel with $P$ ($(\mathcal{X}_{\hat{a}}\, P) \otimes_|^\mathbb{R} \mathscr{Q}$);
3. the continuations of $P$ after $a$ in parallel with the continuations of $Q$ after $\bar{a}$, renormalized w.r.t. the total weight of inputs in $Q$ $\left(\frac{\mathscr{P}_i \otimes_|^\mathbb{R} \mathscr{Q}_o}{\oplus \mathscr{P}_i}\right)$;
4. the continuations of $P$ after $\bar{a}$ in parallel with the continuations of $Q$ after $a$, renormalized w.r.t. the total weight of inputs in $P$ $\left(\frac{\mathscr{P}_o \otimes_|^\mathbb{R} \mathscr{Q}_i}{\oplus \mathscr{Q}_i}\right)$.

Renormalization in (3) and (4) is needed to correctly compute the relative probability of selecting a specific input action.

Formally, the rule corresponding to the composition of the four components intuitively described above is the following:

$$\frac{P \overset{\delta_{\hat{a}}^e}{\rightarrowtail} \mathscr{P}, P \overset{\delta_a^e}{\rightarrowtail} \mathscr{P}_i, P \overset{\delta_{\bar{a}}^e}{\rightarrowtail} \mathscr{P}_o, Q \overset{\delta_{\hat{a}}^e}{\rightarrowtail} \mathscr{Q}, Q \overset{\delta_a^e}{\rightarrowtail} \mathscr{Q}_i, Q \overset{\delta_{\bar{a}}^e}{\rightarrowtail} \mathscr{Q}_o}{P|Q \overset{\delta_{\hat{a}}^e}{\rightarrowtail} (\mathscr{P} \otimes_| (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_| \mathscr{Q}) + \frac{\mathscr{P}_i \otimes_| \mathscr{Q}_o}{\oplus \mathscr{P}_i} + \frac{\mathscr{P}_o \otimes_| \mathscr{Q}_i}{\oplus \mathscr{Q}_i}}$$

Unfortunately, when this synchronization rule is considered, *associativity of parallel composition* is again lost. To see this, just reconsider, in the context of $\text{StoCCS}_{AP}$, the example discussed in the previous section.

Here, we let $P =_\text{def} (P_1 \mid P_2) \mid P_3$ and $Q =_\text{def} P_1 \mid (P_2 \mid P_3)$ where $P_1 =_\text{def} a^{*\omega}.\mathbf{nil}$, $P_2 =_\text{def} a^{*\omega}.\mathbf{nil}$ and $P_3 =_\text{def} \bar{a}^\lambda.\mathbf{nil}$. As before, $P$ reaches both $(\mathbf{nil} \mid P_2) \mid \mathbf{nil}$ and $(P_1 \mid \mathbf{nil}) \mid \mathbf{nil}$ with rate $\frac{\lambda}{2}$ while $Q$ reaches $P_1 \mid (\mathbf{nil} \mid \mathbf{nil})$ and $\mathbf{nil} \mid (P_2 \mid \mathbf{nil})$ with rate $\lambda$. However, when using active-passive approach, this problem can be easily overcome. Indeed, it is sufficient to modify the synchronization rule in such way that:

- the rates of the synchronizations occurring in $P$ and $Q$ are updated to take into account the possible inputs of $P$ and $Q$.
- the synchronizations between outputs in $P$ and inputs in $Q$ (and vice-versa) are renormalized by considering the *total* weight of *inputs* in $P$ and $Q$.

The new rule for synchronization is reported in Figure 13.

Applying the new semantics to the above example, we get:

$$P_1 \mid P_2 \overset{\delta^e_{\hat{a}}}{\rightarrowtail} []$$

$$P_1 \mid P_2 \overset{\delta^e_{a}}{\rightarrowtail} [\mathbf{nil} \mid P_2 \mapsto \omega, P_1 \mid \mathbf{nil} \mapsto \omega]$$

$$P_1 \mid P_2 \overset{\delta^e_{\bar{a}}}{\rightarrowtail} []$$

and, consequently

$$(P_1 \mid P_2) \mid P_3 \overset{\delta^e_{\hat{a}}}{\rightarrowtail} [(\mathbf{nil} \mid P_2) \mid \mathbf{nil} \mapsto \frac{\lambda}{2}, (P_1 \mid \mathbf{nil}) \mid \mathbf{nil} \mapsto \frac{\lambda}{2}]$$

We get, moreover

$$P_2 \mid P_3 \overset{\delta^e_{\hat{a}}}{\rightarrowtail} [\mathbf{nil} \mid \mathbf{nil} \mapsto \lambda]$$

$$P_2 \mid P_3 \overset{\delta^e_{a}}{\rightarrowtail} [\mathbf{nil} \mid P_3 \mapsto \omega]$$

$$P_2 \mid P_3 \overset{\delta^e_{\bar{a}}}{\rightarrowtail} [P_2 \mid \mathbf{nil} \mapsto \lambda]$$

Consequently, we have also

$$P_1 \mid (P_2 \mid P_3) \overset{\delta^e_{\hat{a}}}{\rightarrowtail} [\mathbf{nil} \mid (P_2 \mid \mathbf{nil}) \mapsto \frac{\lambda}{2}, P_1 \mid (\mathbf{nil} \mid \mathbf{nil}) \mapsto \frac{\lambda}{2}]$$

**Proposition 7.4** *For all $P \in \mathcal{P}_{StoCCS_{AP}}$, $\alpha \in \mathcal{L}_{StoCCS_{AP}}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{StoCCS_{AP}}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules consisting of rules (NIL) and (CHO) of Figure 2, rules (OUT1), (OUT2) and (PAR1) in Figure 9, rules (IN1$_P$), (IN2$_P$) in Figure 12 and rule (PAR2$_{AP}$) of Figure 13, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{StoCCS_{AP}}, \mathbb{R}_{\geq 0})$.* □

**Definition 7.2** *The formal semantics of StoCCS$_{AP}$ is the* FuTS$_{FS}$

$$\mathcal{R}_{StoCCS_{AP}} =_{\text{def}} (\mathcal{P}_{StoCCS_{AP}}, \mathcal{L}_{StoCCS_{AP}}, \mathbb{R}_{\geq 0}, \rightarrowtail)$$

*where $\rightarrowtail \subseteq \mathcal{P}_{StoCCS_{AP}} \times \mathcal{L}_{StoCCS_{AP}} \times \mathbf{FTF}(\mathcal{P}_{StoCCS_{AP}}, \mathbb{R}_{\geq 0})$ is the least relation satisfying the set of rules consisting of rules (NIL) and (CHO) of Figure 2, rules (OUT1), (OUT2) and (PAR1) in Figure 9, rules (IN1$_P$), (IN2$_P$) in Figure 12 and rule (PAR2$_{AP}$) of Figure 13.* •

**Theorem 7.5** *$\mathcal{R}_{StoCCS_{AP}}$ is total and deterministic.*

In [15] it has been shown that, by using the above rule, associativity of parallel composition is guaranteed:

**Theorem 7.6** *For all $P, Q, R \in \mathcal{P}_{StoCCS_{AP}}$, $(P \mid Q) \mid R \sim P \mid (Q \mid R)$* □

This result is not in contradiction with the one presented in [29], where it is proved that associativity of parallel composition does not hold for CCS-like calculi if one uses PEPA-like minimum rate synchronization. Our result relies on the introduction of distinct labels for synchronization transitions ($\delta^e_{\hat{a}}$), which keeps track of the interaction channel. This is necessary to properly compute renormalization while taking into account possible new inputs popping up along the derivation. The synchronization labels in [29] are just by $\tau$: due to this crucial information is lost.

$$(\text{PAR2}_{\text{AP}})\frac{P \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{P}, P \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_i, P \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}_o, Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{Q}, Q \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{Q}_i, Q \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{Q}_o}{P|Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \frac{(\mathscr{P} \otimes_| (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) \cdot \oplus \mathscr{P}_i}{\oplus \mathscr{P}_i + \oplus \mathscr{Q}_i} + \frac{((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_| \mathscr{Q}) \cdot \oplus \mathscr{Q}_i}{\oplus \mathscr{P}_i + \oplus \mathscr{Q}_i} + \frac{\mathscr{P}_i \otimes_| \mathscr{Q}_o}{\oplus \mathscr{P}_i + \oplus \mathscr{Q}_i} + \frac{\mathscr{P}_o \otimes_| \mathscr{Q}_i}{\oplus \mathscr{P}_i + \oplus \mathscr{Q}_i}}$$

Figure 13: Rule (PAR2$_{\text{AP}}$) for rate synchronization for StoCCS$_{AP}$

Notice that when deterministic FuTS$_{FS}$ with $\mathbb{R}_{>0}$-valued continuations are considered (like in the case of considered stochastic process calculi), strong Markovian bisimilarity, denoted by $\sim_M$, coincides with the bisimilarity relation of Definition 2.7. We refer the readers interested in further details of Markovian bisimilarity to, e.g. [15, 26, 7].

The CTMC of the *interactions* within process $P \in \mathcal{P}_{StoCCS_{AP}}$ can be derived as expected. The set of states is $(\mathcal{P}_{StoCCS_{AP}})_{P,\Delta_{\hat{\mathcal{A}}}}$; the rate matrix is defined as follows, for each $Q_1, Q_2 \in (\mathcal{P}_{StoCCS_{AP}})_{P,\Delta_{\hat{\mathcal{A}}}}$:

$$\mathbf{R}\, Q_1\, Q_2 \;=_{\text{def}} \sum_{Q_1 \overset{\alpha}{\rightarrowtail}_{P,\Delta_{\hat{\mathcal{A}}}} \mathscr{P}} (\mathscr{P}\, Q_2)$$

Also the semantics we consider in Section 7.1 can be *fixed* to obtain associativity of CCS parallel composition operator. Like above we have to replace (PAR$_{\text{AA-M}}$) by the following one:

$$\frac{P \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{P}, P \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_i, P \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}_o, Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{Q}, Q \overset{\delta^e_{a}}{\rightarrow} \mathscr{Q}_i, Q \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{Q}_o}{P|Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} (\mathscr{P} \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) \cdot \rho_P + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{|} \mathscr{Q}) \cdot \rho_Q + \mathscr{P}_i \otimes_{|} \mathscr{Q}_o \cdot \rho + \mathscr{P}_o \otimes_{|} \mathscr{Q}_i \cdot \rho}$$

where $\rho_Q$, $\rho_P$ are the values used to recompute the rate of synchronizations occurring in $P$ and $Q$, while $\rho$ is used to compute synchronization rates.

$$\rho_P \;=_{\text{def}}\; \frac{\text{MIN}\{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_i, \oplus\mathscr{P}_o + \oplus\mathscr{Q}_o\}}{\text{MIN}\{\oplus\mathscr{P}_i, \oplus\mathscr{P}_o\}} \cdot \frac{\oplus\mathscr{P}_i}{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_i} \cdot \frac{\oplus\mathscr{P}_o}{\oplus\mathscr{P}_o + \oplus\mathscr{Q}_o}$$

$$\rho_Q \;=_{\text{def}}\; \frac{\text{MIN}\{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_i, \oplus\mathscr{P}_o + \oplus\mathscr{Q}_o\}}{\text{MIN}\{\oplus\mathscr{Q}_i, \oplus\mathscr{Q}_o\}} \cdot \frac{\oplus\mathscr{Q}_i}{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_i} \cdot \frac{\oplus\mathscr{Q}_o}{\oplus\mathscr{P}_o + \oplus\mathscr{Q}_o}$$

$$\rho \;=_{\text{def}}\; \frac{\text{MIN}\{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_i, \oplus\mathscr{P}_o + \oplus\mathscr{Q}_o\}}{(\oplus\mathscr{P}_i + \oplus\mathscr{Q}_i) \cdot (\oplus\mathscr{P}_o + \oplus\mathscr{Q}_o)} \cdot \frac{\oplus\mathscr{Q}_i}{\oplus\mathscr{P}_i + \oplus\mathscr{Q}_i} \cdot \frac{\oplus\mathscr{Q}_o}{\oplus\mathscr{P}_o + \oplus\mathscr{Q}_o}$$

The following Theorem holds:

**Theorem 7.7** *For all* $P, Q, R \in \mathcal{P}_{StoCCS_{AA}}$, $(P \mid Q) \mid R \sim P \mid (Q \mid R)$ □

# 8 Dealing with non-determinism

In this section we deal with models where stochastic and non-deterministic behaviors coexist. We shall consider the Language of Interactive Markov Chains (IML) [21]. Before doing this we have to consider a more general framework and we have to extend FuTS so to have multiple semi-rings as codomains of the continuations.

## 8.1 Labelled Function Transition Systems and non-determism

Interactive Markov Chains (IMCs) [21] have been introduced to model non-fully stochastic systems, i.e. systems where non-determinism and stochastic behaviours coexist. Indeed, the key feature of IMCs is a definite, clear distinction between transitions modeling (instantaneous) actions, called *interactive transitions*, and transitions modeling the passage of time, called *Markovian transitions* because durations are, as usual, characterized by exponentially distributed random variables.

**Definition 8.1** *An Interactive Markov Chain is a tuple* $(S, A, \rightarrow, \dashrightarrow, s_0)$ *where* $S$ *is a nonempty but finite set of states,* $A$ *a finite set of actions,* $\rightarrow \subseteq S \times A \times S$ *the set of* interactive transitions*,* $\dashrightarrow \subseteq S \times \mathbb{R}_{>0} \times S$ *the set of* Markov transitions*, and* $s_0 \in S$ *the initial state.* •

An obvious way to describe IMCs with FuTS is to consider as codomain a semi-ring of the form $\mathbb{R} \cup \{\infty\}$, where classical operations on reals are extended as follows: $\forall x \neq 0 : x \cdot \infty = \infty$, $0 \cdot \infty = 0$ and $\forall x. x + \infty = \infty$. Elements in $\mathbb{R}$ indicate rates of *Markovian transitions*, while $\infty$ characterizes *interactive* ones; following the intuition that the latter, being immediate, have an infinite rate. This, simple, choice has however disadvantage of confusing non-determinism and stochasticity. For this reason, we introduce a generalization of FuTS that enable us to define functions whose codomains is the disjoint union ($\uplus$) of commutative semi-rings.

In the specific case we will consider codomains of the form $\mathbb{B} \uplus \mathbb{R}$, where continuations in $\mathbf{FTF}(S, \mathbb{B})$ are used to model non-deterministic transitions, while functions in $\mathbf{FTF}(S, \mathbb{R}_{\geq 0})$ will be used to describe stochastic behaviours. Notice that, FuTS used in the previous sections are simply a special case of the one considered here.

**Definition 8.2 (Labelled Function Transition Systems)** *An $A$-*labelled function transition system *(*FuTS*)* over $\{\mathbb{C}_j\}_{j=1}^k$ *is a tuple* $(S, A, \{\mathbb{C}_j\}_{j=1}^k, \rightarrowtail)$ *where $S$ is a countable, non-empty, set of* states*, $A$ is a countable, non-empty, set of transition* labels*,* $\{\mathbb{C}_j\}_{j=1}^k$ *is a* finite *family of commutative semi-rings, and* $\rightarrowtail \subseteq S \times A \times \biguplus_{j=1}^k \mathbf{TF}(S, \mathbb{C}_j)$ *is the* transition relation*.* •

Whenever necessary or convenient an *initial* state $s_0 \in S$ will be identified, and the relevant FuTS will be the extended tuple $(S, A, \{\mathbb{C}_j\}_{j=1}^k, \rightarrowtail, s_0)$. As in the previous sections, FuTSs will be denoted by $\mathcal{R}, \mathcal{R}_1, \mathcal{R}', \ldots$. Furthermore, all notational conventions as well as definitions, e.g. *total*, *deterministic*, *finite support* FuTS, are extended in a natural way.

## 8.2 A Language of Interactive Markov Chains

In this section, we focus on Interactive Markov Chains (IMCs); in particular we provide a FuTS semantics of Hermanns' Language for IMCs (IML).

Also for IMCs we let the cumulative transition rate from $P_1$ to $P_2$ be denoted by $\mathbf{rt}(P_1, P_2)$. As usual, we restrict our attention to a significative kernel of the full calculus; in Appendix B.6 we recall the SOS for the fragment $\mathrm{IML}_k$ we focus on. The considered operators are:

- *inaction*,
- *rate prefix*,
- *action prefix*,
- *choice*,
- *multi-party synchronization*, and
- *constant*.

Consequently, the set $\mathcal{P}_{IML_k}$ of $\mathrm{IML}_k$ terms is defined by the grammar obtained by selecting from Figure 1 the productions for the above mentioned operators. Notice that, due to the distinction between actions and delays of IMCs, $\mathrm{IML}_k$ has two different prefix operators, namely rate prefix, for delays, and action prefix for actions.

The relevant label set $\mathcal{L}_{IML_k} =_{\mathrm{def}} \mathcal{A} \cup \{\delta^e\}$ includes untimed actions in $\mathcal{A}$ and the delay label $\delta^e$. Function $n : \mathcal{L}_{IML_k} \to \mathcal{A} \cup \{\epsilon\}$ is defined by $n\, a =_{\mathrm{def}} a$ and $n\, \delta^e =_{\mathrm{def}} \epsilon$, assuming $\epsilon \notin \mathcal{A}$.

The relevant set of states is $\mathcal{P}_{IML_k}$ while the set of rules defining the transition relation $\rightarrowtail$ is composed of rules (CHO) and (CNS) of Figure 2, and of the rules of Figure 14, where we let function $\mathcal{X} : \mathcal{L}_{IML_k} \to \mathcal{P}_{IML_k} \to (\mathbf{TF}(\mathcal{P}_{IML_k}, \mathbb{B}) \uplus \mathbf{TF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0}))$ be:

$$\mathcal{X}\, \alpha \;=_{\mathrm{def}}\; \begin{cases} \mathcal{X}_{\mathbb{B}}, & \text{if } \alpha \in \mathcal{A} \\ \mathcal{X}_{\mathbb{R}_{\geq 0}}, & \text{if } \alpha = \delta^e \end{cases}$$

$$(\text{NIL1}_{\text{I}})\frac{}{\mathbf{nil} \overset{\delta^e}{\rightarrowtail} []_{\mathbb{R}_{\geq 0}}} \qquad (\text{NIL2}_{\text{I}})\frac{\alpha \in \mathcal{A}}{\mathbf{nil} \overset{\alpha}{\rightarrowtail} []_{\mathbb{B}}}$$

$$(\text{RPF1}_{\text{I}})\frac{}{\lambda.P \overset{\delta^e}{\rightarrowtail} [P \mapsto \lambda]} \qquad (\text{RPF2}_{\text{I}})\frac{\alpha \in \mathcal{A}}{\lambda.P \overset{\alpha}{\rightarrowtail} []_{\mathbb{B}}}$$

$$(\text{AP1}_{\text{I}})\frac{}{a.P \overset{\delta^e}{\rightarrowtail} []_{\mathbb{R}_{\geq 0}}} \qquad (\text{AP2}_{\text{I}})\frac{}{a.P \overset{a}{\rightarrowtail} [P \mapsto 1_{\mathbb{B}}]} \qquad (\text{AP3}_{\text{I}})\frac{\alpha \in \mathcal{A} \setminus \{a\}}{a.P \overset{\alpha}{\rightarrowtail} []_{\mathbb{B}}}$$

$$(\text{PAR1})\frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}, (n\,\alpha) \notin L}{P \parallel_L Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\ \alpha\ Q)) + ((\mathcal{X}\ \alpha\ P) \otimes^{\alpha}_{\parallel_L} \mathscr{Q})} \qquad (\text{PAR2})\frac{P \overset{\alpha}{\rightarrowtail} \mathscr{P}, Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}, (n\,\alpha) \in L}{P \parallel_L Q \overset{\alpha}{\rightarrowtail} \mathscr{P} \otimes^{\alpha}_{\parallel_L} \mathscr{Q}}$$

Figure 14: Additional semantic rules for $\text{IML}_k$

and the specific function $\otimes^{\alpha}_{\parallel_L}$ be $\otimes^{\mathbb{B}}_{\parallel_L}$ if $\alpha \in \mathcal{A}$ and $\otimes^{\mathbb{R}}_{\parallel_L}$ if $\alpha = \delta^e$. Notably, operators $\mathcal{X}$ and $\otimes^{\alpha}_{\parallel_L}$ are used to avoid type mismatches in the continuation formula while guaranteeing that the *same* rules for interleaving (and, similarly, for) synchronization are used, regardless of the type of continuation functions.

Please notice that the semantic rules for $\text{IML}_k$ are *the same* as those for $\text{TIPP}_k$ except for parameter $\alpha$ in the functions above. Actually, one could use these generalized functions also in the definitions of $\text{TIPP}_k$ and of all the other SPCs. Then *the same* rule format, and in the case of $\text{IML}_k$ and $\text{TIPP}_k$ exactly the same rules, could be used for all the calculi. We preferred to use different formats for the sake of readability.

The following proposition can be easily proven by derivation induction:

**Proposition 8.1** *For all $P \in \mathcal{P}_{IML_k}$, $\alpha \in \mathcal{L}_{IML_k}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0} \uplus \mathbb{B})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set consisting of rules (CHO) and (CNS) of Figure 2 and of the rules of Figure 14, then the following holds: $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0} \uplus \mathbb{B})$.* □

**Proposition 8.2** *For all $P \in \mathcal{P}_{IML_k}$, $\alpha \in \mathcal{L}_{IML_k}$ and $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0} \uplus \mathbb{B})$ such that $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set consisting of rules (CHO) and (CNS) of Figure 2 and of the rules of Figure 14, then the following holds: (i) if $\alpha \in \mathcal{A}$ then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$; (ii) if $\alpha = \delta^e$ then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0})$.* □.

**Definition 8.3** *The formal semantics of $\text{IML}_k$ is the* $\text{FuTS}_{FS}$

$$\mathcal{R}_{IML_k} =_{\text{def}} (\mathcal{P}_{IML_k}, \mathcal{L}_{IML_k}, \{\mathbb{B}, \mathbb{R}_{\geq 0}\}, \rightarrowtail)$$

*where the transition relation $\rightarrowtail \subseteq \mathcal{P}_{IML_k} \times \mathcal{L}_{IML_k} \times (\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B}) \uplus \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0}))$ is the least relation satisfying the set consisting of rules (CHO) and (CNS) of Figure 2 and of the rules of Figure 14.* •

The following theorem characterizes the structure of $\mathcal{R}_{IML_k}$.

**Theorem 8.3** *$\mathcal{R}_{IML_k}$ is total and deterministic.*

The following theorem establishes the formal correspondence between the FuTS of $\text{IML}_k$ and the semantics definition given in [21].

**Theorem 8.4** *Operational semantics of $\text{IML}_k$ induced by $\mathcal{R}_{IML_k}$ coincides with the one given in [21], i.e. for all $P, Q \in \mathcal{P}_{IML_k}$, $a \in \mathcal{A}$, and unique functions $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$ and $\mathscr{P}' \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0})$ such that $P \overset{a}{\rightarrowtail} \mathscr{P}$ and $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}'$ the following holds: (i) $(\mathscr{P}\,Q) = 1_{\mathbb{B}}$ if and only if $P \overset{a}{\rightarrow} Q$; (ii) $(\mathscr{P}'\,Q) = \mathbf{rt}(P, Q)$.* □

## 9 Related Works and Conclusions

In this paper we have presented a number of stochastic process calculi and we have used a unifying framework as the basis for defining their structural operational semantics. Resorting to the so called *State to Function Labelled Transition Systems*, FuTSs for short, has facilitated the *compact* and *compositional* definition of such semantics. Below, we relate our achievements to similar works in the literature and draw some conclusions.

### 9.1 Related Works

Our approach is somehow reminiscent of that of [18] but it is applied to a more general class of process algebras than probabilistic ones. Work similar to ours, also aiming at systematizing the semantics of stochastic calculi has been presented in [29] that studies a (meta-)syntactic framework, called SGSOS, for defining well-behaved Markovian stochastic transition systems. By analogy to the GSOS congruence format for nondeterministic processes [5], stochastic bisimilarity is guaranteed to be a congruence for systems defined by SGSOS rules. However, in [29] no operator on continuation functions is introduced and the issue of transition multiplicity is dealt by resorting to multi-relation transitions.

The approach taken recently in [9] where the semantics of stochastic calculi is defined by associating to each term a measure that encodes the rates of the transitions from the state of a system to a measurable set of states. The actual semantic definitions heavily rely on general functions defined on measure spaces using operators which are very similar to those we proposed. Their general framework has, however, been applied only to stochastic calculi with binary synchronization. Both [29] and [9] restrict their attention to fully stochastic calculi and cannot be easily (at least not obviously) generalized to handle calculi like IML.

Also [28] aim at providing a uniform account of the semantics of different SPCs with the main objective of automatic analysis of stochastic processes. Their take an axiomatic approach and use axioms as rewriting rules to reduce process terms into the common format used by the $\mu$CRL toolset [**?**].

We would like to conclude by only mentioning a few other models that are related to FuTSs. The *structure* of Continuous Time Markov Decision Processes (CTMDPs), as defined by Hermanns and Johr [24], is similar to the structure of *finite support* FuTS with action-indexed random delays ($\Delta_{\mathcal{A}}$-labeled $\text{FuTS}_{FS}$ over $\mathbb{R}_{\geq 0}$). Indeed, a CTMDP is a tuple $\mathcal{M} = (S, \mathcal{A}, T, s_0)$ with $S$ a (finite) set of *states*, $\mathcal{A}$ a (finite) set of action labels, $s_0 \in S$, and $T \subseteq S \times \mathcal{A} \times \mathbf{FTF}(S, \mathbb{R}_{\geq 0})$ the transition relation. Despite the strong structural similarity, there are important conceptual differences between the two models. In fact, while in $\Delta_{\mathcal{A}}$-labeled $\text{FuTS}_{FS}$ the action to perform is selected among those enabled following the *race condition* principle, in the CTMDPs are based on a *reactive* semantics. Indeed, the action that has to be performed is selected by the environment, while the race condition principle is used to select the next state. However, CTMDPs can be defined as FuTS with the appropriate choice of transition labels and continuation function co-domains.

Several models have been proposed in the literature under the name Continuous Time Probabilistic Automata [31, 30, 10]. All these models can be rendered as FuTS. In particular, for the model proposed in [31] similar considerations as those for CTMDPs apply. Th Continuous Time Probabilistic Automata proposed in [30] have been proposed a language theoretic framework; the element $a_{i,j}(x)$ of the infinitesimal matrix used in [30] coincides, in our approach, with $(\mathscr{P}\ j)$ for $i \overset{\delta^e_x}{\rightarrowtail} \mathscr{P}$. Finally, the variant used in [10] is based on standard automata, where transitions are elements of $S \times S$ and have a rate but no label associated. Thus they are directly related to $\{\delta^e\}$-labelled FuTS over $\mathbb{R}_{\geq 0}$.

### 9.2 Conclusions and Future Works

The key feature of FuTSs is the fact that each transition is a triple of the form $(s, \alpha, \mathscr{P})$. The first and the second components are the source state and the label of the transition, while the third component, $\mathscr{P}$, is the *continuation function*, which associates a value of a suitable type to each state, say $s'$. A non-zero value may represent the *cost* to reach $s'$ from $s$ via transition $\alpha$. The only requirement on the co-domains of the continuation functions is that they must be *commutative semi-rings*, which make FuTSs a very general framework. This provides a high level of flexibility while preserving basic properties on primitive operations like sum and multiplication. Moreover, since the third component of the transition relation can be also a disjoint union of sets of functions with different co-domains, FuTSs can be used to model different 'kinds' of transitions by associating different *co-domains* to continuations. Indeed, the framework has been fruitfully used also as semantic domain for the *compositional* definition of the operational semantics of a calculus with both non-deterministic behavior and stochastic delays.

By defining appropriate operators on continuation functions, we have provided a compositional operational semantics of key fragments of major stochastic process calculi including TIPP, EMPA, PEPA, StoCCS and IML and have shed light on differences and similarities. Specifically, we have made clear the different choices relatively to synchronization and to the rôle assigned to the involved actions. We have seen that some of the calculi have multiparty synchronization but, when determining the global cost of a synchronization, some consider all the involved actions as active, while others consider one action active and all the remaining involved ones passive. Other calculi allow only binary synchronization but nevertheless they also make different choices w.r.t. considering actions active or passive. Finally, we have seen how IML de-couples actions from rates, so that action synchronization can be modeled according to standard multiparty synchronization. The FuTS approach has also been successfully used in other papers to model a stochastic extension of the $\pi$-calculus [15] and of other calculi for Mobile Computing and Service Oriented Computing [6, 13].

FuTSs also elegantly solve the issue of transition multiplicity, that has been the cause of many headaches; the costs of equal derivations among those derivable from the transition relation are simply added via operations on continuation functions. Furthermore, FuTSs make it relatively easy to define *associative* parallel composition operators for calculi with the binary interaction paradigm. Indeed, by appropriately defining the composition of continuation functions, the components to be taken into account when one is interested in guaranteeing associativity of parallel composition can be singled out and appropriately combined. It has also been shown that the modeling of binary synchronization becomes simpler when one distinguishes between active and passive actions; if all actions are considered as active the arithmetics of rates becomes more intricate.

In this paper how FuTSs *generality* can be exploited for providing the stochastic semantics of systems with different types of transitions. By further exploiting this feature, appropriately enriching the set of transition labels and changing the codomain of the continuations, our approach can be used to uniformly describe also timed and probabilistic variants of process algebras. A first step in this direction has been performed in [4] where a general notion of equivalence over FuTSs (there called ULTRAS) is introduced and then instantiated to capture the nondeterministic, probabilistic, and stochastic trace and bisimulation equivalence.

A further line of research we plan to follow is the study of the relationship between our approach and that based on co-algebras (see, e.g. [17, 29]) [7]. It would be interesting to reformulate our approach in the co-algebraic framework, focussing at applying standard results from Category Theory to FuTSs.

# Acknowledgments

This work has been partially funded by EU Collaborative project n. 257414 *Autonomic Service-Component Ensamble* (ASCENS), CNR RSTL project XXL and MIUR-PRIN project *Performability-Aware Computing: Logics, Models, and Languages* ( PaCo). We are very grateful to Marco Bernardo, Pedro D'Argenio, Gethin Norman and Roberto Segala for valuable discussions and suggestions.

[7]The semantics of CTMCs, i.e. FuTS $\mathcal{R}_{CTMC}$, as given in Definition 4.1 on page 9, following the similar reasoning to [17], it can be shown that $\mathcal{R}_{CTMC}$ is (isomorphic to) the $\mathscr{F}$-coalgebra $(\mathcal{P}_{CTMC}, \rho)$, in the category **Set** of sets and functions, where $\mathscr{F}$ is the functor which associates $\mathbf{FTF}(S, \mathbb{R}_{\geq 0})$ to any set $S$, and, recalling that $\mathcal{R}_{CTMC}$ is functional, $\rho : \mathcal{P}_{CTMC} \to \mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$ is nothing more than the transition relation of $\mathcal{R}_{CTMC}$, purged of the (redundant) $\delta^e$ label.

# APPENDIX

## A Proof of Proposition 2.1

**Proposition 2.1**. *The transient behavior of CTMC $C = (S, \mathbf{R})$ with $\mathbf{R}[\bar{s}, \bar{s}] > 0$ for some $\bar{s} \in S$ coincides with that of CTMC $\tilde{C} = (S, \tilde{\mathbf{R}})$, such that*

$$\tilde{\mathbf{R}}[s, s'] =_{\text{def}} \begin{cases} 0 & \text{if } s = s' \\ \mathbf{R}[s, s'] & \text{otherwise} \end{cases}$$

□

**Proof** Suppose $\mathbf{R}[\bar{s}, \bar{s}] > 0$ and let $(\pi\, \bar{s}\, t)$ be the probability that $C$ is in state $\bar{s}$ at time $t$, $\mathbb{P}\{C(t) = \bar{s}\}$. For $h$ small enough, the evolution of $C$ in the period $[t, t+h)$ can be captured using $(\pi\, \bar{s}\, t)$ as shown below, letting $p_{\bar{s}}$ denote the probability that no transition from $\bar{s}$ is taken during the period $[t, t+h)$ and $p_{s,\bar{s}}$ denote the probability that a transition from $s$ to $\bar{s}$ takes place during the period $[t, t+h)$[8]:

$\pi\, \bar{s}\, (t + h)$

= {Probability Theory; Definition of $p_{\bar{s}}$ and $p_{s,\bar{s}}$; $h$ small}

$(\pi\, \bar{s}\, t) \cdot (1 - \sum_{s \in S} \mathbf{R}[\bar{s}, s] \cdot h) + \sum_{s \in S} (\pi\, s\, t) \cdot \mathbf{R}[s, \bar{s}] \cdot h + o(h)$

= {Algebra}

$(\pi\, \bar{s}\, t) - (\pi\, \bar{s}\, t) \cdot \sum_{s \in S \setminus \{\bar{s}\}} \mathbf{R}[\bar{s}, s] \cdot h - (\pi\, \bar{s}\, t) \cdot \mathbf{R}[\bar{s}, \bar{s}] \cdot h +$

$\sum_{s \in S \setminus \{\bar{s}\}} (\pi\, s\, t) \cdot \mathbf{R}[s, \bar{s}] \cdot h + (\pi\, \bar{s}\, t) \cdot \mathbf{R}[\bar{s}, \bar{s}] \cdot h + o(h)$

= {Algebra}

$(\pi\, \bar{s}\, t) - (\pi\, \bar{s}\, t) \cdot \sum_{s \in S \setminus \{\bar{s}\}} \mathbf{R}[\bar{s}, s] \cdot h + \sum_{s \in S \setminus \{\bar{s}\}} (\pi\, s\, t) \cdot \mathbf{R}[s, \bar{s}] \cdot h + o(h)$

Thus the evolution of $C$ in the period $[t, t+h)$ does *not* depend on $\mathbf{R}[\bar{s}, \bar{s}]$. And in fact, letting

$$\mathbf{Q}_{\mathbf{R}}[s, s'] =_{\text{def}} \begin{cases} \mathbf{R}[s, s'], \text{if } s \neq s' \\ \\ -\sum_{s'' \in S \setminus \{s\}} \mathbf{R}[s, s''], \text{if } s = s' \end{cases}$$

we get $\pi\, \bar{s}\, (t + h) = (\pi\, \bar{s}\, t) + \left(\sum_{s \in S} (\pi\, s\, t) \cdot \mathbf{Q}_{\mathbf{R}}[s, \bar{s}]\right) \cdot h + o(h)$ from which we get

$$\frac{d(\pi\, \bar{s}\, t)}{dt} = lim_{h \to 0} \frac{(\pi\, \bar{s}\, (t + h)) - (\pi\, \bar{s}\, t)}{h} = \sum_{s \in S} \mathbf{Q}_{\mathbf{R}}[s, \bar{s}] \cdot (\pi\, s\, t)$$

The vector $((\pi\, s\, t))_{s \in S}$ of the transient probabilities for $C$ is thus characterized as the solution of the equation

$$\left(\frac{d(\pi\, s\, t)}{dt}\right)_{s \in S} = ((\pi\, s\, t))_{s \in S}\, \mathbf{Q}_{\mathbf{R}} \qquad \text{given } ((\pi\, s\, 0))_{s \in S}$$

which clearly coincides with the equation for the transient probabilities of $\tilde{C}$ observing that $\mathbf{Q}_{\mathbf{R}} = \mathbf{Q}_{\tilde{\mathbf{R}}}$

## B SOS definitions

In this section the original SOS definition of the relevant process calculi is given. We point out that in the literature the transition multi-relation has often be defined as the least multi-relation satisfying a set of SOS rules (see, e.g. [26] or [21]). Although this definition is not completely correct, since the *least* multi-relation happens to be a relation, thus not capturing, as a matter of fact, transition multiplicity, in the sequel we stick to the original formulation for conformance with the original proposals.

[8] Notice that, we do *not* require $s \neq \bar{s}$, as usually found in the literature (see, e.g. [20]).

$$\frac{}{\lambda.P \xrightarrow{\lambda} P} \qquad \frac{P \xrightarrow{\lambda} R}{P+Q \xrightarrow{\lambda} R} \qquad \frac{Q \xrightarrow{\lambda} R}{P+Q \xrightarrow{\lambda} R} \qquad \frac{P \xrightarrow{\lambda} Q, X{:=}P}{X \xrightarrow{\lambda} Q}$$

Figure 15: SOS Rules for the CTMC Calculus.

$$\frac{}{\langle a,\lambda\rangle.P \xrightarrow{a,\lambda} P} \qquad \frac{P \xrightarrow{a,\lambda} R}{P+Q \xrightarrow{a,\lambda} R} \qquad \frac{Q \xrightarrow{a,\lambda} R}{P+Q \xrightarrow{a,\lambda} R} \qquad \frac{P \xrightarrow{a,\lambda} Q, X{:=}P}{X \xrightarrow{a,\lambda} Q}$$

$$\frac{P \xrightarrow{a,\lambda} P', a \notin L}{P \parallel_L Q \xrightarrow{a,\lambda} P' \parallel_L Q} \qquad \frac{Q \xrightarrow{a,\lambda} Q', a \notin L}{P \parallel_L Q \xrightarrow{a,\lambda} P \parallel_L Q'}$$

$$\frac{P \xrightarrow{a,\lambda_P} P', Q \xrightarrow{a,\lambda_Q} Q', a \in L}{P \parallel_L Q \xrightarrow{a,\lambda_P \cdot \lambda_Q} P' \parallel_L Q'}$$

Figure 16: SOS Rules for the $\text{TIPP}_k$.

### B.1 Calculus for finite CTMCs

The SOS of the Calculus for finite CTMCs of Sect. 4 is the multi-LTS $(\mathcal{P}_{CTMC}, \mathbb{R}_{>0}, \rightarrow)$ where $\rightarrow$ is the multi-relation induced by the rules given in Fig. 15.

### B.2 $\text{TIPP}_k$

The SOS of $\text{TIPP}_k$ (see Sect.6.1) is the multi-LTS $(\mathcal{P}_{TIPP_k}, \mathcal{A}_{TIPP_k} \times \mathbb{R}_{>0}, \rightarrow)$ where $\rightarrow$ is the least multi-relation satisfying the rules given in Fig. 16. Notice that in the original definition of TIPP [23], the *rated-action prefix*, *choice* and *parallel composition* operators are denoted by $(a, \lambda).P$, $[]$ and $|[L]|$ respectively.

### B.3 $\text{EMPA}_k$

The SOS of $\text{EMPA}_k$ (see Sect.6.2) is the multi-LTS $(\mathcal{P}_{EMPA_k}, \mathcal{A}_{EMPA_k} \times (\mathbb{R}_{>0} \cup \{*_\omega \mid \omega \in \mathbb{R}_{>0}\}), \rightarrow)$ where $\rightarrow$ is the multi-relation induced by the rules given in Fig. 17. In the figure, $\tilde{\lambda} \in \mathbb{R}_{>0} \cup \{*_\omega \mid \omega \in \mathbb{R}_{>0}\}$, whereas $weight(P, a)$ and $norm(\omega_1, \omega_2, a, P_1, P2)$ are defined as follows, where $\sum\{\!\!\{\}\!\!\} =_{\text{def}} 0$:

$$weight(P, a) =_{\text{def}} \sum\{\!\!\{\omega \in \mathbb{R}_{>0} | \exists P' \in \mathcal{P}_{EMPA_k}.\ P \xrightarrow{a,*_\omega} P'\}\!\!\}$$

$$norm(\omega_1, \omega_2, a, P_1, P2) =_{\text{def}} \frac{\omega_1}{weight(P_1,a)} \cdot \frac{\omega_2}{weight(P_2,a)} \cdot (weight(P_1, a) + weight(P_2, a))$$

In the original definition of EMPA [1], the *rec* operator for constant definition is used, instead of defining equations.

### B.4 $\text{PEPA}_k$

The SOS of $\text{PEPA}_k$ (see Sect.6.3) is the multi-LTS $(\mathcal{P}_{PEPA_k}, \mathcal{A}_{PEPA_k} \times \mathbb{R}_{>0}, \rightarrow)$ where $\rightarrow$ is the least multi-relation satisfying the rules given in Fig. 18. In the figure $r_\alpha(P)$ and $r(\alpha, \lambda_1, \lambda_2, P, Q)$ are used, which are defined as follows:

$$r_\alpha((\beta, \lambda).P) =_{\text{def}} \begin{cases} \lambda, & \text{if } \beta = \alpha \\ 0, & \text{if } \beta \neq \alpha \end{cases}$$

$$\frac{}{\langle a,\lambda\rangle.P \xrightarrow{a,\lambda} P} \qquad \frac{}{\langle a,*_\omega\rangle.P \xrightarrow{a,*\omega} P} \qquad \frac{P \xrightarrow{a,\tilde{\lambda}} R}{P+Q \xrightarrow{a,\tilde{\lambda}} R} \qquad \frac{Q \xrightarrow{a,\tilde{\lambda}} R}{P+Q \xrightarrow{a,\tilde{\lambda}} R}$$

$$\frac{P \xrightarrow{a,\tilde{\lambda}} Q, X:=P}{X \xrightarrow{a,\tilde{\lambda}} Q} \qquad \frac{P \xrightarrow{a,\tilde{\lambda}} P', a\notin L}{P \,\|_L\, Q \xrightarrow{a,\tilde{\lambda}} P' \,\|_L\, Q} \qquad \frac{Q \xrightarrow{a,\tilde{\lambda}} Q', a\notin L}{P \,\|_L\, Q \xrightarrow{a,\tilde{\lambda}} P \,\|_L\, Q'}$$

$$\frac{P \xrightarrow{a,\lambda} P', Q \xrightarrow{a,*\omega} Q', a\in L}{P \,\|_L\, Q \xrightarrow{a,\lambda\cdot\frac{\omega}{weight(Q,a)}} P' \,\|_L\, Q'} \qquad \frac{P \xrightarrow{a,*\omega} P', Q \xrightarrow{a,\lambda} Q', a\in L}{P \,\|_L\, Q \xrightarrow{a,\lambda\cdot\frac{\omega}{weight(P,a)}} P' \|_L Q'}$$

$$\frac{P \xrightarrow{a,*\omega_1} P', Q \xrightarrow{a,*\omega_2} Q', a\in L}{P \,\|_L\, Q \xrightarrow{a,*norm(\omega_1,\omega_2,a,P,Q)} P' \,\|_L\, Q'}$$

Figure 17: SOS Rules for the EMPA$_k$.

$$\frac{}{\langle a,\lambda\rangle.P \xrightarrow{\alpha,\lambda} P} \qquad \frac{P \xrightarrow{\alpha,\lambda} R}{P+Q \xrightarrow{\alpha,\lambda} R} \qquad \frac{Q \xrightarrow{\alpha,\lambda} R}{P+Q \xrightarrow{\alpha,\lambda} R} \qquad \frac{P \xrightarrow{\alpha,\lambda} Q, X:=P}{X \xrightarrow{\alpha,\lambda} Q}$$

$$\frac{P \xrightarrow{\alpha,\lambda} P', \alpha\notin L}{P \,\|_L\, Q \xrightarrow{\alpha,\lambda} P' \,\|_L\, Q} \qquad \frac{Q \xrightarrow{\alpha,\lambda} Q', \alpha\notin L}{P \,\|_L\, Q \xrightarrow{\alpha,\lambda} P \,\|_L\, Q'}$$

$$\frac{P \xrightarrow{\alpha,\lambda_1} P', Q \xrightarrow{\alpha,\lambda_2} Q', \alpha\in L}{P \,\|_L\, Q \xrightarrow{\alpha,r(\alpha,\lambda_1,\lambda_2,P,Q)} P' \,\|_L\, Q'}$$

Figure 18: SOS Rules for PEPA$_k$.

$$r_\alpha(P+Q) \;=_{\text{def}}\; r_\alpha(P) + r_\alpha(Q)$$

$$r_\alpha(P \bowtie_L Q) \;=_{\text{def}}\; \begin{cases} min(r_\alpha(P), r_\alpha(Q)) \text{ if } \alpha \in L \\ r_\alpha(P) + r_\alpha(Q), \text{ if } \alpha \notin L \end{cases}$$

$$r(\alpha,\lambda_1,\lambda_2,P,Q) \;=_{\text{def}}\; \frac{\lambda_1}{r_\alpha(P)} \cdot \frac{\lambda_2}{r_\alpha(Q)} \cdot min(r_\alpha(P), r_\alpha(Q))$$

In the original definition of PEPA [26], the *rated-action prefix* and *parallel composition* operators are denoted by $(a,\lambda).P$ and $\bowtie_L$ respectively.

## B.5 StoCCS

In the *proved operational semantics*, StoCCS is defined by the rules of Fig. 19 where $\theta$ ranges over *derivation proofs*, e.g. represented by terms of the following grammar:

$$\theta ::= (a,\lambda) \;\Big|\; +_1\theta \;\Big|\; +_2\theta \;\Big|\; \|_1\theta \;\Big|\; \|_2\theta \;\Big|\; \langle \|_1\theta, \|_2\theta\rangle$$

Function $r(\alpha,\lambda_1,\lambda_2,P,Q)$, used to compute the rate of a synchronization, is defined as for PEPA when *minimum synchronization rate* approach is used, while is simply defined as $\lambda_1 \cdot \lambda_2$ when *multiplicative rate* approach is used.

$$\frac{}{a^{\lambda}.P \xrightarrow{(a,\lambda)} P} \qquad \frac{P \xrightarrow{\theta} R}{P+Q \xrightarrow{+_1\theta} R} \qquad \frac{Q \xrightarrow{\theta} R}{P+Q \xrightarrow{+_2\theta} R}$$

$$\frac{P \xrightarrow{\theta} P'}{P|Q \xrightarrow{\|_1\theta} P'|Q} \qquad \frac{Q \xrightarrow{\theta} Q'}{P|Q \xrightarrow{\|_2\theta} P|Q'}$$

$$\frac{P \xrightarrow{\theta_1(a,\lambda_1)} P', Q \xrightarrow{\theta_2(\bar{a},\lambda_2)} Q'}{P|Q \xrightarrow{\langle \|_1\theta_1(a,\lambda_1), \|_2\theta_2(\bar{a},\lambda_2)\rangle r(x,\lambda_1,\lambda_2,P,Q)} P'|Q'}$$

Figure 19: Rules for StoCCS.

$$\frac{}{a.P \xrightarrow{a} P} \qquad \frac{P \xrightarrow{a} R}{P+Q \xrightarrow{a} R} \qquad \frac{Q \xrightarrow{a} R}{P+Q \xrightarrow{a} R} \qquad \frac{P \xrightarrow{a} Q, X:=P}{X \xrightarrow{a} Q}$$

$$\frac{P \xrightarrow{a} P', a \notin L}{P \|_L Q \xrightarrow{a} P' \|_L Q} \qquad \frac{Q \xrightarrow{a} Q', a \notin L}{P \|_L Q \xrightarrow{a} P \|_L Q'} \qquad \frac{P \xrightarrow{a} P', Q \xrightarrow{a} Q', a \in L}{P \|_L Q \xrightarrow{a} P' \|_L Q'}$$

$$\frac{}{\lambda.P \overset{\lambda}{\dashrightarrow} P} \qquad \frac{P \overset{\lambda}{\dashrightarrow} R}{P+Q \overset{\lambda}{\dashrightarrow} R} \qquad \frac{Q \overset{\lambda}{\dashrightarrow} R}{P+Q \overset{\lambda}{\dashrightarrow} R} \qquad \frac{P \overset{\lambda}{\dashrightarrow} Q, X:=P}{X \overset{\lambda}{\dashrightarrow} Q}$$

$$\frac{P \overset{\lambda}{\dashrightarrow} P'}{P \|_L Q \overset{\lambda}{\dashrightarrow} P' \|_L Q} \qquad \frac{Q \overset{\lambda}{\dashrightarrow} Q'}{P \|_L Q \overset{\lambda}{\dashrightarrow} P \|_L Q'}$$

Figure 20: SOS Rules for the IML$_k$.

From the LTS induced by the SOS rules an action labelled CTMC can be derived, by removing the proofs from the labels while summing up the rates of identical transitions.

In the original definition of Stochastic CCS [29], the *rated-output-action prefix*, *rated-input-action prefix* and *parallel composition* operators are denoted by $(\bar{a}, \lambda).P$, $(a, \lambda).P$ and $\|$ respectively.

### B.6 IML$_k$

The SOS definition of IML$_k$ (see Sect.8.2) is given in Fig. 20. The *action transition* relation $\rightarrow \subset \mathcal{P}_{IML_k} \times \mathcal{A} \times \mathcal{P}_{IML_k}$ is the least relation and the *Markovian transition* relation $\dashrightarrow \subset \mathcal{P}_{IML_k} \times \mathbb{R}_{>0} \times \mathcal{P}_{IML_k}$ is the least *multi*-relation given by the rules in Fig. 20.

Notice that in [21] parallel composition (and hiding) are not defined by means of an explicit set of SOS rules, but, being derived operators, it is defined by means of expansion laws (and specific laws for hiding). Here we preferred to use explicit SOS rules for uniformity reasons and because we do not address equivalence relations.

In the original definition of IML [21], the *parallel composition* operator is denoted by $\|[L]\|$.

## C Proofs related to Sect. 4

### C.1 Proof of Proposition 4.1

**Proposition 4.1**. For all $P \in \mathcal{P}_{CTMC}$ and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$, if $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}$ can be derived from the rules of Fig. 2, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$. □

**Proof** By derivation induction. Let $n \geq 1$ be the length of the derivation for proving $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}$.

*Base case*: Trivial since the only cases in which $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}$ can be derived with a proof of length 1 are those in which $\mathscr{P} = []_{\mathbb{R}_{\geq 0}}$ or $\mathscr{P} = [P' \mapsto \lambda]$ and $[], [P' \mapsto \lambda] \in \mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$ by definition.

*Inductive step*: The last assert of any proof of length $n > 1$ must be of the form $P + Q \overset{\delta^e}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$ or $X \overset{\delta^e}{\rightarrowtail} \mathscr{P}$. In the first case $\mathscr{P} + \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$, since $\mathscr{P}, \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$ by I.H. and $\mathbf{FTF}(\mathcal{P}_{CTMC}, \mathbb{R}_{\geq 0})$ is closed under + by definition of +. In the second case the assert trivially follows from the I.H.

## C.2 Proof of Theorem 4.2

**Theorem 4.2**. $\mathcal{R}_{CTMC}$ is total and deterministic. □

**Proof** *$\mathcal{R}_{CTMC}$ is total*: By induction on the structure, taking inaction and rate prefix as base cases, for which the assert is trivially proven. For the inductive step we show only the case $P + Q$ which is also very simple because $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}$ and $Q \overset{\delta^e}{\rightarrowtail} \mathscr{Q}$, for some $\mathscr{P}$ and $\mathscr{Q}$ by the I.H., hence $P + Q \overset{\delta^e}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$ by the FuTS semantics of the CTMC Language (Fig. 2).

*$\mathcal{R}_{CTMC}$ is deterministic*: By induction on the length of the derivation. We prove only the inductive step for case $P + Q$ here, the others being simpler.

$P + Q \overset{\delta^e}{\rightarrowtail} \mathscr{R}_1,\ P + Q \overset{\delta^e}{\rightarrowtail} \mathscr{R}_2$

$\Rightarrow$ {Def. of $\rightarrowtail$ (Fig. 2)}

$\mathscr{R}_1 = \mathscr{P}_1 + \mathscr{Q}_1, \mathscr{R}_2 = \mathscr{P}_2 + \mathscr{Q}_2, P \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1, Q \overset{\delta^e}{\rightarrowtail} \mathscr{Q}_1, P \overset{\delta^e}{\rightarrowtail} \mathscr{P}_2, Q \overset{\delta^e}{\rightarrowtail} \mathscr{Q}_2$

$\Rightarrow$ {I.H.}

$\mathscr{P}_1 = \mathscr{P}_2,\ \mathscr{Q}_1 = \mathscr{Q}_2,\ \mathscr{R}_1 = \mathscr{P}_1 + \mathscr{Q}_1, \mathscr{R}_2 = \mathscr{P}_2 + \mathscr{Q}_2$

$\Rightarrow$ {Algebra}

$\mathscr{R}_1 = \mathscr{R}_2$

# D Proofs related to Sect. 6

## D.1 Proof of Proposition 6.1

**Proposition 6.1**. For all $P \in \mathcal{P}_{TIPP_k}$, $\alpha \in \mathcal{L}_{TIPP_k}$ and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules composed only of rules (NIL), (CHO) and (CNS) of Figure 2 plus those of Figure 6, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$. □**Proof** By induction on the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. We prove only the inductive step. The last assert of any derivation of length $n > 1$ must be of the form $P + Q \overset{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$, or $P \parallel_L Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes_{\parallel_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}} Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P) \otimes_{\parallel_L} \mathscr{Q})$, or $P \parallel_L Q \overset{\alpha}{\rightarrowtail} \mathscr{P} \parallel_L \mathscr{Q}$, or $X \overset{\alpha}{\rightarrowtail} \mathscr{P}$. In all cases the assert follows from Proposition 5.1 since $\mathscr{P}, \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$ by I.H. and $(\mathcal{X}_{\mathbb{R}_{\geq 0}} P), (\mathcal{X}_{\mathbb{R}_{\geq 0}} Q) \in \mathbf{FTF}(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$ by definition.

## D.2 Proof of Theorem 6.2

**Theorem 6.2**. $\mathcal{R}_{TIPP_k}$ is total and deterministic. □

**Proof** *$\mathcal{R}_{TIPP_k}$ is total*: By induction on the structure, taking inaction and rated action prefix as base cases, for which the assert is trivially proven. For the inductive step we show only the case $P \parallel_L Q$ which is also very simple because $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ and $Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}$, for some $\mathscr{P}$ and $\mathscr{Q}$ by the I.H., hence, assuming $(n\,\alpha) \notin L$, $P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes_{\|_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{\|_L} \mathscr{Q})$ by the FuTS semantics of TIPP$_k$; the case for $(n\,\alpha) \in L$ is similar.

$\mathcal{R}_{TIPP_k}$ *is deterministic*: By induction of the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. We prove only the inductive step for case $P \;\|_L\; Q$ here, the others being similar or simpler. Let us suppose there are two different derivations of length $n > 1$: $P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} \mathscr{R}_1$ and $P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} \mathscr{R}_2$, with $(n\,\alpha) \notin L$:

$P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} \mathscr{R}_1,\ P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} \mathscr{R}_2$

$\Rightarrow$ {Def. of $\rightarrowtail$}

$\mathscr{R}_1 = (\mathscr{P}_1 \otimes_{\|_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{\|_L} \mathscr{Q}_1), \mathscr{R}_2 = (\mathscr{P}_2 \otimes_{\|_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{\|_L} \mathscr{Q}_2)$

$\Rightarrow$ {I.H.}

$\mathscr{P}_1 = \mathscr{P}_2, \mathscr{Q}_1 = \mathscr{Q}_2,$

$\mathscr{R}_1 = (\mathscr{P}_1 \otimes_{\|_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{\|_L} \mathscr{Q}_1), \mathscr{R}_2 = (\mathscr{P}_2 \otimes_{\|_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{\|_L} \mathscr{Q}_2)$

$\Rightarrow$ {Algebra}

$\mathscr{R}_1 = \mathscr{R}_2$

The case $P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} \mathscr{R}_1$ and $P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} \mathscr{R}_2$, with $(n\,\alpha) \in L$ is similar.

## D.3 Proof of Theorem 6.3

**Theorem 6.3**. For all $P, Q \in \mathcal{P}_{TIPP_k}$, $\alpha \in \mathcal{L}_{TIPP_k}$, and unique $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{TIPP_k}, \mathbb{R}_{\geq 0})$ such that $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ the following holds: $(\mathscr{P}\, Q) = \mathbf{rt}_{(n\,\alpha)}(P, Q)$ □

**Proof** By induction of the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. We prove only the inductive step for case $P_1 \;\|_L\; P_2$, under the assumption $(n\,\alpha) \in L$, the other cases being similar. By definition of the FuTS semantics of TIPP$_k$, the last assert of the derivation is of the form $P_1 \;\|_L\; P_2 \overset{\alpha}{\rightarrowtail} \mathscr{P}_1 \otimes_{\|_L} \mathscr{P}_2$, with $P_1 \overset{\alpha}{\rightarrowtail} \mathscr{P}_1$ and $P_2 \overset{\alpha}{\rightarrowtail} \mathscr{P}_2$. We observe that if $Q$ is not of the form $Q_1 \;\|_L\; Q_2$ then $(\mathscr{P}_1 \otimes_{\|_L} \mathscr{P}_2)\, Q = 0$. On the other hand, we observe that the only transitions from $P_1 \;\|_L\; P_2$ allowed by the SOS semantics of TIPP$_k$ are to terms of the form $Q_1 \;\|_L\; Q_2$, so also $\mathbf{rt}_{(n\,\alpha)}(P_1 \;\|_L\; P_2, Q) = 0$ if $Q$ is not of the form $Q_1 \;\|_L\; Q_2$. Let us assume $Q$ is of the form $Q_1 \;\|_L\; Q_2$.

$(\mathscr{P}_1 \otimes_{\|_L} \mathscr{P}_2)\, Q_1 \;\|_L\; Q_2$

$=$ {Def. $(\mathscr{P}_1 \otimes_{\|_L} \mathscr{P}_2)$}

$(\mathscr{P}_1\, Q_1) \cdot (\mathscr{P}_2\, Q_2)$

$=$ {$P_1 \overset{\alpha}{\rightarrowtail} \mathscr{P}_1$ and $P_2 \overset{\alpha}{\rightarrowtail} \mathscr{P}_2$; I.H.}

$\mathbf{rt}_{(n\,\alpha)}(P_1, Q_1) \cdot \mathbf{rt}_{(n\,\alpha)}(P_2, Q_2)$

$=$ {SOS definition of TIPP$_k$; Def. of $\mathbf{rt}_a$}

$\mathbf{rt}_{(n\,\alpha)}(P_1 \;\|_L\; P_2, Q_1 \;\|_L\; Q_2)$

## D.4 Proof of Proposition 6.4

**Proposition 6.4**. For all $P \in \mathcal{P}_{EMPA_k}$, $\alpha \in \mathcal{L}_{EMPA_k}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules composed only of rules (NIL), (CHO) and (CNS) of Figure 2 plus rules (RAPF1) and (RAPF2) and (PAR1) of Figure 6 and those of Figure 7, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ □

**Proof** By derivation induction. Let $n \geq 1$ be the length of the derivation for proving $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$.

*Base case*: Trivial since the only cases in which $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived with a proof of length 1 are those in which $\mathscr{P} = []_{\mathbb{R}_{\geq 0}}$ or $\mathscr{P} = [P' \mapsto x]$, with $x \in \mathbb{R}_{>0}$, and $[]_{\mathbb{R}_{\geq 0}} \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ and $[P' \mapsto x] \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ by definition.

*Inductive step*: The last assert of any proof of length $n > 1$ must be of the form $P + Q \overset{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$, or $P \parallel_L Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes_{\parallel_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}} Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P) \otimes_{\parallel_L} \mathscr{Q})$, or $P \parallel_L Q \overset{\delta^e_{a*}}{\rightarrowtail} \mathscr{P} \otimes_{\parallel_L} \mathscr{Q} \cdot \frac{(\oplus \mathscr{P}) + (\oplus \mathscr{Q})}{(\oplus \mathscr{P}) \cdot (\oplus \mathscr{Q})}$, or $P \parallel_L Q \overset{\delta^e_a}{\rightarrowtail} \mathscr{P}_o \otimes_{\parallel_L} \mathscr{Q}_i \cdot \frac{1}{\oplus \mathscr{Q}_i} + \mathscr{P}_i \otimes_{\parallel_L} \mathscr{Q}_o \cdot \frac{1}{\oplus \mathscr{P}_i}$, or, finally $X \overset{\alpha}{\rightarrowtail} \mathscr{P}$. In all cases the assert follows using Proposition 5.1 since $\mathscr{P}, \mathscr{Q}, \mathscr{P}_o, \mathscr{P}_i, \mathscr{Q}_o, \mathscr{Q}_i \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ by I.H. and $(\mathcal{X}_{\mathbb{R}_{\geq 0}} P), (\mathcal{X}_{\mathbb{R}_{\geq 0}} Q) \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ by definition.

## D.5 Proof of Theorem 6.5

**Theorem 6.5**. $\mathcal{R}_{EMPA_k}$ is total and deterministic. □

**Proof** *$\mathcal{R}_{EMPA_k}$ is total*: By induction on the structure. The proof is the same as that of Theorem 6.2 for TIPP$_k$.

*$\mathcal{R}_{EMPA_k}$ is deterministic*: By induction of the length on the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. The proof is essentially the same as that of Theorem 6.2 for TIPP$_k$.

## D.6 Proof of Theorem 6.6

**Theorem 6.6**. For all $P, Q \in \mathcal{P}_{EMPA_k}$, $\delta^e_a, \delta^e_{a*} \in \mathcal{L}_{EMPA_k}$, and unique functions $\mathscr{P}, \mathscr{P}' \in \mathbf{FTF}(\mathcal{P}_{EMPA_k}, \mathbb{R}_{\geq 0})$ such that $P \overset{\delta^e_a}{\rightarrowtail} \mathscr{P}$ and $P \overset{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}'$, the following holds: $(\mathscr{P}\, Q) = \mathbf{rt}_a(P, Q)$, $(\mathscr{P}'\, Q) = \mathbf{wt}_a(P, Q)$, and $(\oplus \mathscr{P}') = weight(P, a)$. □

**Proof** We prove the assert by induction on the length of the derivations for $P \overset{\delta^e_a}{\rightarrowtail} \mathscr{P}$ and for $P \overset{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}'$. We prove only the inductive step for case $P_1 \parallel_L P_2$, under the assumption $a \in L$, the other cases being similar. By definition of the FuTS semantics of EMPA$_k$, the last asserts of the derivations are of the form

$$P_1 \parallel_L P_2 \overset{\delta^e_a}{\rightarrowtail} \mathscr{P}^0_1 \otimes_{\parallel_L} \mathscr{P}^i_2 \cdot \frac{1}{\oplus \mathscr{P}^i_2} + \mathscr{P}^i_1 \otimes_{\parallel_L} \mathscr{P}^o_2 \cdot \frac{1}{\oplus \mathscr{P}^i_1} \text{ and}$$

$$P_1 \parallel_L P_2 \overset{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}_1 \otimes_{\parallel_L} \mathscr{P}_2 \cdot \frac{(\oplus \mathscr{P}_1) + (\oplus \mathscr{P}_2)}{(\oplus \mathscr{P}_1) \cdot (\oplus \mathscr{P}_2)}.$$

We observe that if $Q$ is not of the form $Q_1 \parallel_L Q_2$ then $(\mathscr{P}^o_1 \otimes_{\parallel_L} \mathscr{P}^i_2)\, Q = 0$, as well as $(\mathscr{P}^i_1 \otimes_{\parallel_L} \mathscr{P}^o_2)\, Q$, and $(\mathscr{P}_1 \otimes_{\parallel_L} \mathscr{P}_2)\, Q$. On the other hand, we observe that the only transitions from $P_1 \parallel_L P_2$ allowed by the original SOS semantics of EMPA$_k$ are to terms of the form $Q_1 \parallel_L Q_2$, so also $\mathbf{rt}_a(P_1 \parallel_L P_2, Q) = \mathbf{wt}_a(P_1 \parallel_L P_2, Q) = 0$ if $Q$ is not of the form $Q_1 \parallel_L Q_2$. Let us assume $Q$ is of the form $Q_1 \parallel_L Q_2$.

$$\begin{pmatrix} (\mathscr{P}^o_1 \otimes_{\parallel_L} \mathscr{P}^i_2 \cdot \frac{1}{\oplus \mathscr{P}^i_2} + \mathscr{P}^i_1 \otimes_{\parallel_L} \mathscr{P}^o_2 \cdot \frac{1}{\oplus \mathscr{P}^i_1})\, Q_1 \parallel_L Q_2 \\ (\mathscr{P}_1 \otimes_{\parallel_L} \mathscr{P}_2 \cdot \frac{(\oplus \mathscr{P}_1) + (\oplus \mathscr{P}_2)}{(\oplus \mathscr{P}_1) \cdot (\oplus \mathscr{P}_2)})\, Q_1 \parallel_L Q_2 \\ \oplus\Big((\mathscr{P}_1 \otimes_{\parallel_L} \mathscr{P}_2) \cdot \frac{(\oplus \mathscr{P}_1) + (\oplus \mathscr{P}_2)}{(\oplus \mathscr{P}_1) \cdot (\oplus \mathscr{P}_2)}\Big) \end{pmatrix}$$

$=$ $\{\text{Def. } \otimes_{\parallel_L}, \oplus\}$

$$\begin{pmatrix} (\mathscr{P}_1^o\, Q_1) \cdot (\mathscr{P}_2^i\, Q_2) \cdot \frac{1}{\oplus \mathscr{P}_2^i} + (\mathscr{P}_1^i\, Q_1) \cdot (\mathscr{P}_2^o\, Q_2) \cdot \frac{1}{\oplus \mathscr{P}_1^i} \\ (\mathscr{P}_1\, Q_1) \cdot (\mathscr{P}_2\, Q_2) \cdot \frac{(\oplus \mathscr{P}_1) + (\oplus \mathscr{P}_2)}{(\oplus \mathscr{P}_1) \cdot (\oplus \mathscr{P}_2)} \\ \frac{(\oplus \mathscr{P}_1) + (\oplus \mathscr{P}_2)}{(\oplus \mathscr{P}_1) \cdot (\oplus \mathscr{P}_2)} \cdot \sum_{Q_1' \,\|_L\, Q_2' \in \mathcal{P}_{EMPA_k}} (\mathscr{P}_1\, Q_1') \cdot (\mathscr{P}_2\, Q_2') \end{pmatrix}$$

$$= \quad \{P_h \overset{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}_h^i, P_h \overset{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}_h, h = 1, 2; \text{Unicity corollary (Th. 6.5)}\}$$

$$\begin{pmatrix} (\mathscr{P}_1^o\, Q_1) \cdot (\mathscr{P}_2^i\, Q_2) \cdot \frac{1}{\oplus \mathscr{P}_2^i} + (\mathscr{P}_1^i\, Q_1) \cdot (\mathscr{P}_2^o\, Q_2) \cdot \frac{1}{\oplus \mathscr{P}_1^i} \\ (\mathscr{P}_1^i\, Q_1) \cdot (\mathscr{P}_2^i\, Q_2) \cdot \frac{(\oplus \mathscr{P}_1^i) + (\oplus \mathscr{P}_2^i)}{(\oplus \mathscr{P}_1^i) \cdot (\oplus \mathscr{P}_2^i)} \\ \frac{(\oplus \mathscr{P}_1^i) + (\oplus \mathscr{P}_2^i)}{(\oplus \mathscr{P}_1^i) \cdot (\oplus \mathscr{P}_2^i)} \cdot \sum_{Q_1' \,\|_L\, Q_2' \in \mathcal{P}_{EMPA_k}} (\mathscr{P}_1^i\, Q_1') \cdot (\mathscr{P}_2^i\, Q_2') \end{pmatrix}$$

$$= \quad \{P_h \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_h^o, P_h \overset{\delta^e_{a*}}{\rightarrowtail} \mathscr{P}_h^i, h = 1, 2; \text{I.H.}\}$$

$$\begin{pmatrix} \frac{\mathbf{rt}_a(P_1,Q_1) \cdot \mathbf{wt}_a(P_2,Q_2)}{weight(a,P_2)} + \frac{\mathbf{wt}_a(P_1,Q_1) \cdot \mathbf{rt}_a(P_2,Q_2)}{weight(a,P_1)} \\ \mathbf{wt}_a(P_1, Q_1) \cdot \mathbf{wt}_a(P_2, Q_2) \cdot \frac{weight(a,P_1) + weight(a,P_2)}{weight(a,P_1) \cdot weight(a,P_2)} \\ \frac{weight(a,P_1) + weight(a,P_2)}{weight(a,P_1) \cdot weight(a,P_2)} \cdot \sum_{Q_1' \,\|_L\, Q_2' \in \mathcal{P}_{EMPA_k}} \mathbf{wt}_a(P_1, Q_1') \cdot \mathbf{wt}_a(P_2\, Q_2') \end{pmatrix}$$

$$= \quad \{\text{Def. } \mathbf{rt}_a; \text{Def. } \mathbf{wt}_a; \text{Def. } weight; \text{SOS definition of EMPA}\}$$

$$\begin{pmatrix} \mathbf{rt}_a(P_1 \;\|_L\; P_2, Q_1 \;\|_L\; Q_2) \\ \mathbf{wt}_a(P_1 \;\|_L\; P_2, Q_1 \;\|_L\; Q_2) \\ weight(a, P_1 \;\|_L\; P_2) \end{pmatrix}$$

## D.7 Proof of Proposition 6.7

**Proposition 6.7**. For all $P \in \mathcal{P}_{PEPA_k}$, $\alpha \in \mathcal{L}_{PEPA_k}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules composed only of rules (CHO) and (CNS) of Figure 2, (RAPF1), (RAPF2) and (PAR1) of Figure 6 and rule (PAR2$_\text{P}$) Figure 8, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$. □

**Proof** By induction on the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. We prove only the inductive step. The last assert of any derivation of length $n > 1$ must be of the form $P + Q \overset{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$, or $P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes_{\|_L} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{\|_L} \mathscr{Q})$, or $P \;\|_L\; Q \overset{\alpha}{\rightarrowtail} \mathscr{P} \;\|_L\; \mathscr{Q}$, or $X \overset{\alpha}{\rightarrowtail} \mathscr{P}$. In all cases the assert follows from Proposition 5.1 since $\mathscr{P}, \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$ by I.H. and $(\mathcal{X}\, \alpha\, P), (\mathcal{X}\, \alpha\, Q) \in \mathbf{FTF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$ by definition.

## D.8 Proof of Theorem 6.8

**Theorem 6.8**. $\mathcal{R}_{PEPA_k}$ is total and deterministic. □

**Proof** $\mathcal{R}_{PEPA_k}$ *is total*: By induction on the structure. The proof is the same as that of Theorem 6.2 for TIPP$_k$.
$\mathcal{R}_{PEPA_k}$ *is deterministic*: By induction of the length on the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. The proof is essentially the same as that of Theorem 6.2 for TIPP$_k$.

## D.9 Proof of Theorem 6.9

**Theorem 6.9**. For all $P, Q \in \mathcal{P}_{PEPA_k}$, $\alpha \in \mathcal{L}_{PEPA_k}$, and unique $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{PEPA_k}, \mathbb{R}_{\geq 0})$ such that $P \stackrel{\alpha}{\rightarrowtail} \mathscr{P}$ the following holds: $(\mathscr{P}\, Q) = \mathbf{rt}_{(n\,\alpha)}(P, Q)$ □

**Proof** By induction of the length of the derivation for $P \stackrel{\alpha}{\rightarrowtail} \mathscr{P}$. The proof is essentially the same as that of Theorem 6.2 for TIPP$_k$.

## D.10 Proof of Proposition 7.1

**Proposition 7.1**. For all $P \in \mathcal{P}_{StoCCS_{AA}}$, $\alpha \in \mathcal{L}_{StoCCS_{AA}}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0})$ if $P \stackrel{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules consisting of rules (NIL) and (CHO) of Figure 2, plus rules in Figure 9, and using one out of (PAR$_{\text{AA-M}}$) and (PAR$_{\text{AA-}*}$), then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{StoCCS_{AA}}, \mathbb{R}_{\geq 0})$. □

**Proof** By induction on the length of the derivation for $P \stackrel{\alpha}{\rightarrowtail} \mathscr{P}$. We prove only the inductive step. The last assert of any derivation of length $n > 1$ must be of the form

- $P + Q \stackrel{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$,
- or $P \mid Q \stackrel{\alpha}{\rightarrowtail} (\mathscr{P} \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{|} \mathscr{Q})$,
- or $P \mid Q \stackrel{\delta^e_{\hat{a}}}{\rightarrowtail} (\mathscr{P} \mid (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \mid \mathscr{Q}) + \mathscr{P}_i \otimes_{|} \mathscr{Q}_o + \mathscr{P}_o \otimes_{|} \mathscr{Q}_i$,
- or $P \mid Q \stackrel{\delta^e_{\hat{a}}}{\rightarrowtail} (\mathscr{P} \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{|} \mathscr{Q}) + \mathscr{P}_i \mid \mathscr{Q}_o + \mathscr{P}_o \mid \mathscr{Q}_i$ (rule (PAR$_{\text{AA-}*}$) is used),
- or $P \mid Q \stackrel{\delta^e_{\hat{a}}}{\rightarrowtail} (\mathscr{P} \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P) \otimes_{|} \mathscr{Q}) + \mathscr{P}_i \mid \mathscr{Q}_o \cdot \frac{\textsc{min}\{\oplus\mathscr{P}_i, \oplus\mathscr{Q}_o\}}{\oplus\mathscr{P}_i \cdot \oplus\mathscr{Q}_o} + \mathscr{P}_o \mid \mathscr{Q}_i \cdot \frac{\textsc{min}\{\oplus\mathscr{P}_o, \oplus\mathscr{Q}_i\}}{\oplus\mathscr{P}_o \cdot \oplus\mathscr{Q}_i}$ (rule (PAR$_{\text{AA-M}}$) is used).

In all cases the assert follows using Proposition 5.1 since $\mathscr{P}, \mathscr{P}_i, \mathscr{P}_0, \mathscr{Q}, \mathscr{Q}_i, \mathscr{Q}_o \in \mathbf{FTF}(\mathcal{P}_{StoCCS}, \mathbb{R}_{\geq 0})$ by I.H. and $(\mathcal{X}_{\mathbb{R}_{\geq 0}}\, P), (\mathcal{X}_{\mathbb{R}_{\geq 0}}\, Q) \in \mathbf{FTF}(\mathcal{P}_{StoCCS_I}, \mathbb{R}_{\geq 0})$ by definition.

## D.11 Proof of Theorem 7.2

**Theorem 7.2**. Both $\mathcal{R}^*_{StoCCS_{AA}}$ and $\mathcal{R}^M_{StoCCS_{AA}}$ are total and deterministic.

**Proof** $\mathcal{R}^*_{StoCCS_{AA}}$ and $\mathcal{R}^M_{StoCCS_{AA}}$ *are total*: By induction on the structure. The proof is the same as that of Theorem 6.2 for TIPP$_k$.

$\mathcal{R}^*_{StoCCS_{AA}}$ and $\mathcal{R}^M_{StoCCS_{AA}}$ *are deterministic*: By induction of the length of the derivation for $P \stackrel{\alpha}{\rightarrowtail} \mathscr{P}$. The proof is essentially the same as that of Theorem 6.2 for TIPP$_k$.

## D.12 Proof of Theorem 7.3

**Theorem 7.3**. Both in the minimum rate and in the multiplicative approach, for all $P, Q \in \mathcal{P}_{StoCCS_I}$, $\alpha \in \mathcal{L}_{StoCCS_I}$, and unique $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{StoCCS_I}, \mathbb{R}_{\geq 0})$ such that $P \stackrel{\alpha}{\rightarrowtail} \mathscr{P}$ the following holds:

$$(\mathscr{P}\, Q) = \begin{cases} \mathbf{rt}_a(P, Q), \text{ if } \alpha = \delta^e_a \text{ with } a \in \mathcal{A} \cup \bar{\mathcal{A}} \\ \\ \mathbf{rt}_{\langle a|\bar{a}\rangle}(P, Q) + \mathbf{rt}_{\langle \bar{a}|a\rangle}(P, Q), \text{ if } \alpha = \delta^e_{\hat{a}} \text{ with } a \in \mathcal{A} \end{cases}$$

□

**Proof**

**Minimum rate approach:**

We prove only the inductive step for case $P_1 \mid P_2$, under the assumption $\alpha = \delta^e_{\hat{a}}$ with $a \in \mathcal{A}$, the other cases being similar. By definition of the FuTS semantics of StoCCS$_I$, the last assert of the derivation is of the form $P_1 \mid P_2 \stackrel{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{R}$, where

$$\mathscr{R} = (\mathscr{P}_1 \mid (\mathcal{X}_a\, P_2)) + ((\mathcal{X}_a\, P_1) \mid \mathscr{P}_2) + \mathscr{P}_{1i} \mid \mathscr{P}_{2o} \cdot \frac{\text{MIN}\{\oplus \mathscr{P}_{1i}, \oplus \mathscr{P}_{2o}\}}{\oplus \mathscr{P}_{1i} \cdot \oplus \mathscr{P}_{2o}} + \mathscr{P}_{1o} \mid \mathscr{P}_{2i} \cdot \frac{\text{MIN}\{\oplus \mathscr{P}_{1o}, \oplus \mathscr{P}_{1i}\}}{\oplus \mathscr{P}_{1o} \cdot \oplus \mathscr{P}_{2i}}$$

with $P_1 \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{P}_1$, $P_1 \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_{1i}$, $P_1 \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}_{1o}$, $P_2 \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{P}_2$, $P_2 \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_{2i}$, $P_2 \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}_{2o}$. We observe that if $Q$ is not of the form $Q_1 \mid Q_2$ then $\mathscr{R}\, Q = 0$. On the other hand, we observe that the only transitions from $P_1 \mid P_2$ allowed by the SOS semantics of StoCCS$_I$ are to terms of the form $Q_1 \mid Q_2$, so also $\mathbf{rt}_\alpha(P_1 \mid P_2, Q) = 0$ if $Q$ is not of the form $Q_1 \mid Q_2$. Let us assume $Q$ is of the form $Q_1 \mid Q_2$. There are several cases to be considered depending to the fact that $\delta^e_{\hat{a}}$ denotes synchronisations

1. *between* $P_1$ and $P_2$ where
   (a) $P_1$ performs the input $a$ and $P_2$ performs the output $\bar{a}$, or
   (b) $P_1$ performs the output $\bar{a}$ and $P_2$ performs the input $a$, or
2. *within*
   (a) $P_1$ alone, or
   (b) $P_2$ alone

   and are inherited by $P_1 \mid P_2$.

We consider only the case of synchronisations between $P_1$ and $P_2$ where $P_1$ performs the input $a$ and $P_2$ performs the output $\bar{a}$, the other cases being similar or simpler. In this case, function $\mathscr{R}$ reduces to $\mathscr{P}_{1i} \mid \mathscr{P}_{2o} \cdot \frac{\text{MIN}\{\oplus \mathscr{P}_{1i}, \oplus \mathscr{P}_{2o}\}}{\oplus \mathscr{P}_{1i} \cdot \oplus \mathscr{P}_{2o}}$ since this is the sub-term of $\mathscr{R}$ dealing with such synchronisations between $P_1$ and $P_2$. We get the following derivation noting that, under our assumptions, $\oplus \mathscr{P}_{1i} \cdot \oplus \mathscr{P}_{2o} \neq 0$:

$$
\begin{array}{ll}
& (\mathscr{P}_{1i} \mid \mathscr{P}_{2o} \cdot \frac{\text{MIN}\{\oplus \mathscr{P}_{1i}, \oplus \mathscr{P}_{2o}\}}{\oplus \mathscr{P}_{1i} \cdot \oplus \mathscr{P}_{2o}})(\, Q_1 \mid Q_2) \\
= & \{\text{Def. } \mathscr{R}_1 \mid \mathscr{R}_2 \cdot \frac{\text{MIN}\{\oplus \mathscr{R}_1, \oplus \mathscr{R}_2\}}{\oplus \mathscr{R}_1 \cdot \oplus \mathscr{R}_2} \text{ in } \mathbf{FTF}(\mathcal{P}_{StoCCS}, \mathbb{R}_{\geq 0})\} \\
& (\mathscr{P}_{1i}\, Q_1) \cdot (\mathscr{P}_{2o}\, Q_2) \cdot \frac{\text{MIN}\{\oplus \mathscr{P}_{1i}, \oplus \mathscr{P}_{2o}\}}{\oplus \mathscr{P}_{1i} \cdot \oplus \mathscr{P}_{2o}} \\
= & \{\, P_1 \overset{\delta^e_{a}}{\rightarrowtail} \mathscr{P}_{1i},\, P_2 \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}_{2o}; \text{I.H.}\} \\
& \mathbf{rt}_a(P_1, Q_1) \cdot \mathbf{rt}_{\bar{a}}(P_2, Q_2) \cdot \frac{\text{MIN}\{\oplus \mathscr{P}_{1i}, \oplus \mathscr{P}_{2o}\}}{\oplus \mathscr{P}_{1i} \cdot \oplus \mathscr{P}_{2o}} \\
= & \{\text{SOS definition of StoCCS}_I\text{; Def. of } \mathbf{rt}_{\langle a|\bar{a}\rangle}\ \} \\
& \mathbf{rt}_{\langle a|\bar{a}\rangle}(P_1 \mid P_2, Q_1 \mid Q_2)
\end{array}
$$

Moreover, the assumption that $P_1$ performs the input $a$ and $P_2$ performs the output $\bar{a}$ implies $\mathbf{rt}_{\langle \bar{a}|a\rangle}(P_1 \mid P_2, Q_1 \mid Q_2) = 0$ and this completes the proof.

**Multiplicative rate approach**
The proof proceeds as in the previous case, where rule (PAR$_{\text{AA-M}}$) is replaced with (PAR$_{\text{AA-}*}$), and $r(\alpha, \lambda_1, \lambda_2, P, Q)$ is $\lambda_1 \cdot \lambda_2$.

## D.13 Proof of Proposition 7.4

**Proposition 7.4**. For all $P \in \mathcal{P}_{StoCCS_{AP}}$, $\alpha \in \mathcal{L}_{StoCCS_{AP}}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{StoCCS_{AP}}, \mathbb{R}_{\geq 0})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set of rules consisting of rules (NIL) and (CHO) of Figure 2, rules (OUT1), (OUT2) and (PAR1) in Figure 9, rules (IN1$_\text{P}$), (IN2$_\text{P}$) in Figure 12 and rule (PAR2$_{\text{AP}}$) of Figure 13, then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{StoCCS_{AP}}, \mathbb{R}_{\geq 0})$. □

**Proof** By induction on the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. We prove only the inductive step. The last assert of any derivation of length $n > 1$ must be of the form $P + Q \overset{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$, or $P \mid Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \mid (\mathcal{X}\, \alpha\, Q)) + ((\mathcal{X}\, \alpha\, P) \mid \mathscr{Q})$, or $P \mid Q \overset{\delta^e_{\hat{a}}}{\rightarrowtail} (\mathscr{P} \mid (\mathcal{X}_a\, Q)) + ((\mathcal{X}_a\, P) \mid \mathscr{Q}) + \frac{\mathscr{P}_i | \mathscr{Q}_o}{\oplus \mathscr{P}_i} + \frac{\mathscr{P}_o | \mathscr{Q}_i}{\oplus \mathscr{Q}_i}$, or

$X \overset{\alpha}{\rightarrowtail} \mathscr{P}$. In all cases the assert follows using Proposition 5.1 since $\mathscr{P}, \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{StoCCS}, \mathbb{R}_{\geq 0})$ by I.H. and $(\mathcal{X}_a\, P), (\mathcal{X}_a\, Q) \in \mathbf{FTF}(\mathcal{P}_{StoCCS_{II}}, \mathbb{R}_{\geq 0})$ by definition.

## D.14 Proof of Theorem 7.5

**Theorem 7.5**. $\mathcal{R}_{StoCCS_{AP}}$ is total and deterministic. □

**Proof** $\mathcal{R}_{StCCS_k}$ *is total*: By induction on the structure. The proof is the same as that of Theorem 6.2 for $\text{TIPP}_k$.

$\mathcal{R}_{StoCCS_{II}}$ *is deterministic*: By induction of the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. The proof is essentially the same as that of Theorem 6.2 for $\text{TIPP}_k$.

## D.15 Proof Theorem 7.6

**Theorem 7.6**. For all $P, Q, R \in \mathcal{P}_{StoCCS_{AP}}$, $(P \mid Q) \mid R \sim P \mid (Q \mid R)$ □

**Proof** The statement follows by proving that

$$
\begin{aligned}
\mathcal{E} \;=\; & \{\langle P|(Q|R), (P|Q)|R\rangle | P, Q, R \in \mathcal{P}_{StoCCS_{AP}}\} \\
& \cup\{\langle (P|Q)|R, P|(Q|R)\rangle | P, Q, R \in \mathcal{P}_{StoCCS_{AP}}\} \\
& \cup\{\langle P|(Q|R), P|(Q|R)\rangle | P, Q, R \in \mathcal{P}_{StoCCS_{AP}}\} \\
& \cup\{\langle (P|Q)|R, (P|Q)|R\rangle | P, Q, R \in \mathcal{P}_{StoCCS_{AP}}\} \\
& \cup\{\langle P, P\rangle | P \in \mathcal{P}_{StoCCS_{AP}}\}
\end{aligned}
$$

is a bisimulation in $\mathcal{R}_{StoCCS_{AP}}$.

Indeed, if we prove that $\mathcal{E}$ is a bisimulation, we have that $\mathcal{E} \subseteq \sim$. Moreover, for each $P, Q, R \in \mathcal{P}_{StoCCS_{AP}}$, by definition of $\mathcal{E}$, we have that $\langle (P \mid Q) \mid R, P \mid (Q \mid R)\rangle \in \mathcal{E}$. Therefore, for each $P, Q, R \in \mathcal{P}_{StoCCS_{AP}}$, $(P \mid Q) \mid R \sim P \mid (Q \mid R)$.

To prove that $\mathcal{E}$ is a bisimulation, first we have to prove that $\mathcal{E}$ is an equivalence relation. It is easy to prove that:

- for each $P$, $\langle P, P\rangle \in \mathcal{E}$;
- $\langle P, Q\rangle \in \mathcal{E} \Rightarrow \langle Q, P\rangle \in \mathcal{E}$
- $\langle P, Q\rangle \in \mathcal{E} \wedge \langle Q, R\rangle \in \mathcal{E} \Rightarrow \langle P, R\rangle \in \mathcal{E}$

We have to prove that if $\langle P, Q\rangle \in \mathcal{E}$ then for each equivalence class $C$ of $\mathcal{E}$:

$$P \overset{\alpha}{\rightarrowtail} \mathscr{P} \Longrightarrow Q \overset{\alpha}{\rightarrowtail} \mathscr{Q} \wedge \bigoplus \mathscr{P}\, C = \bigoplus \mathscr{Q}\, C$$

We can distinguish three cases:

- $P = Q$

  $\Rightarrow$ $\{\mathcal{R}_{StoCCS_{AP}}$ is fully stochastic$\}$

  $P \overset{\alpha}{\rightarrowtail} \mathscr{P} \wedge Q \overset{\alpha}{\rightarrowtail} \mathscr{Q} \Longrightarrow \mathscr{P} = \mathscr{Q}$

  $\Rightarrow$ {Def. $(\bigoplus \mathscr{P} C)$}

  $\bigoplus \mathscr{P}\, C = \bigoplus \mathscr{Q}\, C$

- $P = (P_1|P_2)|P_3$ and $Q = P_1|(P_2|P_3)$:

  - $\alpha \neq \delta_{\bar{a}}^{e}$

    $\Rightarrow$ {Def. $\rightarrowtail$ where $P_i \overset{\alpha}{\rightarrowtail} \mathscr{P}_i$}

    $\mathscr{P} = ((\mathscr{P}_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2)) + ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}_2)) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3) + (\mathcal{X}_{\mathbb{R}_{\geq 0}} (P_1|P_2)) \otimes_{|} \mathscr{P}_3$

    $\Rightarrow$ {Def. $\rightarrowtail$ }

$$Q \overset{\alpha}{\rightarrowtail} \mathscr{P}_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}}(P_1 \mid P_2)) + (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} (\mathscr{P}_2 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)) + (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) \otimes_{|} \mathscr{P}_3)$$

$\Rightarrow$ $\{C = \{X|(Y|Z), (X|Y)|Z\}\}$

$$\bigoplus \mathscr{P}\ C = \begin{cases} \mathscr{P}_1(X) & Y = P_2 \wedge Z = P_3 \\ \mathscr{P}_2(Y) & X = P_1 \wedge Z = P_3 \\ \mathscr{P}_3(Z) & X = P_1 \wedge Y = P_2 \end{cases}$$

$$\bigoplus \mathscr{Q}\ C = \begin{cases} \mathscr{P}_1(X) & Y = P_2 \wedge Z = P_3 \\ \mathscr{P}_2(Y) & X = P_1 \wedge Z = P_3 \\ \mathscr{P}_3(Z) & X = P_1 \wedge Y = P_2 \end{cases}$$

$\Rightarrow$

$$\bigoplus \mathscr{P}\ C = \bigoplus \mathscr{Q}\ C$$

- $\alpha = \delta^e_{\hat{a}}$

$\Rightarrow$ $\{\text{Def.} \rightarrowtail\}$

$$\begin{aligned}
\mathscr{P} \quad = \quad & \frac{\frac{\mathscr{P}_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) \cdot \oplus \mathscr{P}^i_1}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2} \otimes_{|} P_3 \cdot (\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{\frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}_2 \cdot \oplus \mathscr{P}^i_2}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2} \otimes_{|} P_3 \cdot (\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \\
& \frac{\frac{\mathscr{P}^i_1 \otimes_{|} \mathscr{P}^o}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2} \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3) \cdot (\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{\frac{\mathscr{P}^o_1 \otimes_{|} \mathscr{P}^i}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2} \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3) \cdot (\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^o_3} + \\
& \frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}}(P_1|P_2)) \otimes_{|} \mathscr{P}_3 \cdot \oplus \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^i_3} + \frac{(\mathscr{P}^i_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) + (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}^i_2) \otimes_{|} \mathscr{P}^o_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \\
& \frac{(\mathscr{P}^o_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) + (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}^o_2) \otimes_{|} \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} \\
= \quad & \frac{((\mathscr{P}_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2)) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)) \cdot \oplus \mathscr{P}^i_1}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{(((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}_2) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)) \cdot \oplus \mathscr{P}^i_2}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \\
& \frac{(\mathscr{P}^i_1 \otimes_{|} \mathscr{P}^o) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{(\mathscr{P}^o_1 \otimes_{|} \mathscr{P}^i) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^o_3} + \frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}}(P_1|P_2)) \otimes_{|} \mathscr{P}_3 \cdot \oplus \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^i_3} + \\
& \frac{(\mathscr{P}^i_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) + (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}^i_2) \otimes_{|} \mathscr{P}^o_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \frac{(\mathscr{P}^o_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) + (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}^o_2) \otimes_{|} \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} \\
= \quad & \frac{((\mathscr{P}_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2)) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)) \cdot \oplus \mathscr{P}^i_1}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{(((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}_2) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)) \cdot \oplus \mathscr{P}^i_2}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \\
& \frac{(\mathscr{P}^i_1 \otimes_{|} \mathscr{P}^o) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{(\mathscr{P}^o_1 \otimes_{|} \mathscr{P}^i) \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^o_3} + \frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}}(P_1|P_2)) \otimes_{|} \mathscr{P}_3 \cdot \oplus \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^i_3} + \\
& \frac{(\mathscr{P}^i_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2)) \otimes_{|} \mathscr{P}^o_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \frac{((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}^i_2) \otimes_{|} \mathscr{P}^o_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \frac{(\mathscr{P}^o_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2)) \otimes_{|} \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \frac{((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} \mathscr{P}^o_2) \otimes_{|} \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3}
\end{aligned}$$

where:

$$P_j \overset{\delta^e_{\hat{a}}}{\rightarrowtail} \mathscr{P}_j, P_j \overset{\delta^e_a}{\rightarrowtail} \mathscr{P}^i_j, P_j \overset{\delta^e_{\bar{a}}}{\rightarrowtail} \mathscr{P}^o_j$$

$\Rightarrow$ $\{\text{Def.} \rightarrowtail\}$

$$\begin{aligned}
\mathscr{Q} \quad = \quad & \frac{\mathscr{P}_1 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}}(P_2|P_3)) \cdot \oplus \mathscr{P}^i_1}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} (\mathscr{P}_2 \otimes_{|} (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3)) \cdot \oplus \mathscr{P}^i_2}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \\
& \frac{\mathscr{P}^i_1 | (\mathscr{P}^o | (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3))}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}_3} + \frac{\mathscr{P}^o_1 | (\mathscr{P}^i | (\mathcal{X}_{\mathbb{R}_{\geq 0}} P_3))}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^o_3} + \frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) \otimes_{|} \mathscr{P}_3) \cdot \oplus \mathscr{P}^i_3}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 + \oplus \mathscr{P}^i_3} + \\
& \frac{\mathscr{P}^i_1 | ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) \otimes_{|} \mathscr{P}^o_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} (\mathscr{P}^i_2 \otimes_{|} \mathscr{P}^o_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \frac{\mathscr{P}^o_1 \otimes_{|} ((\mathcal{X}_{\mathbb{R}_{\geq 0}} P_2) \otimes_{|} \mathscr{P}^i_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3} + \frac{(\mathcal{X}_{\mathbb{R}_{\geq 0}} P_1) \otimes_{|} (\mathscr{P}^o_2 \otimes_{|} \mathscr{P}^i_3)}{\oplus \mathscr{P}^i_1 + \oplus \mathscr{P}^i_2 \oplus \mathscr{P}^i_3}
\end{aligned}$$

$\Rightarrow$ $\{C = \{X|(Y|Z), (X|Y)|Z\}\}$

$$\bigoplus \mathscr{P}\ C = \bigoplus \mathscr{Q}\ C$$

- $P = X|(Y|Z)$ and $Q = (X|Y)|Z$: follows like the previous one.

### D.16 Proof Theorem 7.7

**Theorem 7.7**. For all $P, Q, R \in \mathcal{P}_{StoCCS_{AA}}$, $(P \mid Q) \mid R \sim P \mid (Q \mid R)$ □

**Proof** The proof proceeds like in the case of Theorem 7.6.

## E Proofs related to Sect. 8.2

### E.1 Proof of Proposition 8.1

**Proposition 8.1**. For all $P \in \mathcal{P}_{IML_k}$, $\alpha \in \mathcal{L}_{IML_k}$, and $\mathscr{P} \in \mathbf{TF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0} \uplus \mathbb{B})$, if $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set consisting of rules (CHO) and (CNS) of Figure 2 and of the rules of Figure 14, then the following holds: $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0} \uplus \mathbb{B})$. □

**Proof** By induction on the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. The proof is similar to that of of Proposition 6.1 for TIPP$_k$.

### E.2 Proof of Proposition 8.2

**Proposition 8.2**. For all $P \in \mathcal{P}_{IML_k}$, $\alpha \in \mathcal{L}_{IML_k}$ and $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0} \uplus \mathbb{B})$ such that $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived using the set consisting of rules (CHO) and (CNS) of Figure 2 and of the rules of Figure 14, then the following holds: (i) if $\alpha \in \mathcal{A}$ then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$; (ii) if $\alpha = \delta^e$ then $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0})$. □.

**Proof** By induction on the length of the derivation for $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$. Let $n \geq 1$ be the length of the derivation for proving $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$.

*Base case*: Trivial since the only cases in which $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$ can be derived with a proof of length 1 are those for inaction, rate-prefix and action-prefix. In all these cases the assert easily follows from the relevant semantics rules (see Figure 14).

*Inductive step*: The last assert of any proof of length $n > 1$ must be of the form $P + Q \overset{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$, or $P \parallel_L Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\, \alpha\, Q)) + ((\mathcal{X}\, \alpha\, P) \otimes^{\alpha}_{\parallel_L} \mathscr{Q})$, or $X \overset{\alpha}{\rightarrowtail} \mathscr{P}$, or $P \parallel_L Q \overset{\alpha}{\rightarrowtail} \mathscr{P} \otimes^{\alpha}_{\parallel_L} \mathscr{Q}$. We show the proof only for the first two cases, the other being simpler.

*Case*: $P + Q \overset{\alpha}{\rightarrowtail} \mathscr{P} + \mathscr{Q}$

(i) if $\alpha \in \mathcal{A}$, from the definition of the FuTS semantics of IML$_k$ we get $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$, $Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}$ and by the I.H., $\mathscr{P}, \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$ thus also $\mathscr{P} + \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$ by definition of $+_{\mathbb{B}}$ and, consequently, of $+$ over $\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$; (ii) if $\alpha = \delta^e$ the proof is identical as for case (i), but using $\mathbb{R}_{\geq 0}$ instead of $\mathbb{B}$.

*Case*: $P \parallel_L Q \overset{\alpha}{\rightarrowtail} (\mathscr{P} \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\, \alpha\, Q)) + ((\mathcal{X}\, \alpha\, P) \otimes^{\alpha}_{\parallel_L} \mathscr{Q})$

(i) Suppose $\alpha \in \mathcal{A}$. From the definition of the FuTS semantics of IML$_k$ $P \overset{\alpha}{\rightarrowtail} \mathscr{P}$, $Q \overset{\alpha}{\rightarrowtail} \mathscr{Q}$, and, by I.H., $\mathscr{P}, \mathscr{Q} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$. Moreover, also $(\mathcal{X}\, \alpha\, Q), (\mathcal{X}\, \alpha\, P) \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$ by definition of $\mathcal{X}$, since $\alpha \in \mathcal{A}$. Using the closure properties of $\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$, and in particular Proposition 5.1, we get the assert; (ii) if $\alpha = \delta^e$ the proof is identical as for case (i), but using $\mathbb{R}_{\geq 0}$ instead of $\mathbb{B}$.

### E.3 Proof of Theorem 8.3

**Theorem 8.3**. $\mathcal{R}_{IML_k}$ is total and deterministic.

**Proof** By induction on the structure. The proof is the similar to that of Theorem 6.2 for TIPP$_k$.

## E.4 Proof of Theorem 8.4

**Theorem** 8.4. For all $P, Q \in \mathcal{P}_{IML_k}$, $a \in \mathcal{A}$, and unique functions $\mathscr{P} \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$ and $\mathscr{P}' \in \mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0})$ such that $P \overset{a}{\rightarrowtail} \mathscr{P}$ and $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}'$ the following holds: (i) $(\mathscr{P}\, Q) = 1_{\mathbb{B}}$ if and only if $P \overset{a}{\to} Q$; (ii) $(\mathscr{P}'\, Q) = \mathbf{rt}(P, Q)$. □

**Proof** *Proof of part (i)*. For the sake of conciseness, we prove both the direct (⇒) and the reverse (⇐) implication together. For the direct implication we proceed by induction on the length of the derivation for FuTS semantics ($P \overset{a}{\rightarrowtail} \mathscr{P}$), while we use induction on the length of the derivation for the SOS ($P \overset{a}{\to} Q$) for the reverse implication. Let $n \geq 1$ be the length of the derivation for proving $P \overset{a}{\rightarrowtail} \mathscr{P}$ ($P \overset{a}{\to} Q$, respectively).

*Base case*: Trivial since the only case in which $P \overset{a}{\rightarrowtail} \mathscr{P}$ can be derived with a proof of length 1 and $(\mathscr{P}\, Q) = 1_{\mathbb{B}}$ is $a.Q \overset{a}{\rightarrowtail} \mathscr{P}$ with $\mathscr{P} = [Q \mapsto 1_{\mathbb{B}}]$. But $a.Q \overset{a}{\to} Q$ in the SOS definition of IML$_k$. On the other hand, the only case in which $P \overset{a}{\to} Q$ can be derived with a proof of length 1 is when $P = a.Q$, in which case $P \overset{a}{\rightarrowtail} [Q \mapsto 1_{\mathbb{B}}]$.

*Inductive step*: The last assert of any proof of length $n > 1$ must be of the form $P_1 + P_2 \overset{a}{\rightarrowtail} \mathscr{P}_1 + \mathscr{P}_2$, or $X \overset{a}{\rightarrowtail} \mathscr{P}_1$, or $P_1 \parallel_L P_2 \overset{a}{\rightarrowtail} (\mathscr{P}_1 \otimes^{\alpha}_{\parallel_L} (X\, a\, P_2)) + ((X\, a\, P_1) \otimes^{\alpha}_{\parallel_L} \mathscr{P}_2)$, or $P_1 \parallel_L P_2 \overset{a}{\rightarrowtail} \mathscr{P}_1 \otimes^{\alpha}_{\parallel_L} \mathscr{P}_2$ and $P_1 + P_2 \overset{a}{\to} Q$, or $X \overset{a}{\to} Q$, or $P_1 \parallel_L P_2 \overset{a}{\to} Q$ (with $a \notin L$), or $P_1 \parallel_L P_2 \overset{a}{\to} Q$ (with $a \in L$), respectively.

*Case*: $\begin{cases} P_1 + P_2 \overset{a}{\rightarrowtail} \mathscr{P}_1 + \mathscr{P}_2, \text{ for } \Rightarrow \\ P_1 + P_2 \overset{a}{\to} Q, \text{ for } \Leftarrow \end{cases}$

$(\mathscr{P}_1 + \mathscr{P}_2)\, Q = 1_{\mathbb{B}}$

⇒ ⇐ {Def. FuTS semantics of IML$_k$; Def. $(\mathscr{P}_1 + \mathscr{P}_2)$}

$P_1 \overset{a}{\rightarrowtail} \mathscr{P}_1, P_2 \overset{a}{\rightarrowtail} \mathscr{P}_2, (\mathscr{P}_1\, Q) = 1_{\mathbb{B}}$ or $(\mathscr{P}_2\, Q) = 1_{\mathbb{B}}$

⇒ {I.H.}
⇐ {I.H.; Unicity of $\mathscr{P}_1$, $\mathscr{P}_2$}

$P_1 \overset{a}{\to} Q$ or $P_2 \overset{a}{\to} Q$

⇒ ⇐ {Def. SOS of IML$_k$}

$P_1 + P_2 \overset{a}{\to} Q$

*Case*: $\begin{cases} X \overset{a}{\rightarrowtail} \mathscr{P}_1, X := P_1, \text{ for } \Rightarrow \\ X \overset{a}{\to} Q, X := P_1, \text{ for } \Leftarrow \end{cases}$

$(\mathscr{P}_1\, Q) = 1_{\mathbb{B}}$

⇒ {Def. FuTS semantics of IML$_k$}
⇐ {Logics}

$P_1 \overset{a}{\rightarrowtail} \mathscr{P}_1, (\mathscr{P}_1\, Q) = 1_{\mathbb{B}}$

⇒ {I.H.}
⇐ {I.H.; Unicity of $\mathscr{P}_1$}

$P_1 \overset{a}{\to} Q$

$\substack{\Rightarrow \\ \Leftarrow}$ {Def. SOS of $\text{IML}_k$}

$X \xrightarrow{a} Q$

$$Case: \begin{cases} P_1 \ \|_L \ P_2 \overset{a}{\rightarrowtail} (\mathscr{P}_1 \otimes^{\alpha}_{\|_L} (\mathcal{X}\, a\, P_2)) + ((\mathcal{X}\, a\, P_1) \otimes^{\alpha}_{\|_L} \mathscr{P}_2), \text{ for } \Rightarrow \\ P_1 \ \|_L \ P_2 \xrightarrow{a} Q, a \notin L, \text{ for } \Leftarrow \end{cases}$$

$(\mathscr{P}_1 \otimes^{\alpha}_{\|_L} (\mathcal{X}\, a\, P_2)) + ((\mathcal{X}\, a\, P_1) \otimes^{\alpha}_{\|_L} \mathscr{P}_2)\, Q = 1_{\mathbb{B}}$

$\Rightarrow$ {Def. FuTS semantics of $\text{IML}_k$; Def. $+$ and $\otimes^{\alpha}_{\|_L}$ over $\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$; Def. $\mathcal{X}$}

$\Leftarrow$ {Def. $+$ and $\otimes^{\alpha}_{\|_L}$ over $\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$; Def. $\mathcal{X}$}

$P_1 \overset{a}{\rightarrowtail} \mathscr{P}_1, P_2 \overset{a}{\rightarrowtail} \mathscr{P}_2, a \notin L,$

$Q = Q_1 \ \|_L \ P_2$ for some $Q_1$ such that $(\mathscr{P}_1\, Q_1) = 1_{\mathbb{B}}$, or

$Q = P_1 \ \|_L \ Q_2$ for some $Q_2$ such that $(\mathscr{P}_2\, Q_2) = 1_{\mathbb{B}}$

$\Rightarrow$ {I.H.}

$\Leftarrow$ {I.H.; Unicity of $\mathscr{P}_1$, $\mathscr{P}_2$}

$Q = Q_1 \ \|_L \ P_2$ for some $Q_1$ such that $P_1 \xrightarrow{a} Q_1, a \notin L$ or

$Q = P_1 \ \|_L \ Q_2$ for some $Q_2$ such that $P_2 \xrightarrow{a} Q_2, a \notin L$

$\substack{\Rightarrow \\ \Leftarrow}$ {Def. SOS of $\text{IML}_k$}

$P_1 \ \|_L \ P_2 \xrightarrow{a} Q, a \notin L$

$$Case: \begin{cases} P_1 \ \|_L \ P_2 \overset{a}{\rightarrowtail} \mathscr{P}_1 \ \|_L \ \mathscr{P}_2, \text{ for } \Rightarrow \\ P_1 \ \|_L \ P_2 \xrightarrow{a} Q, a \in L, \text{ for } \Leftarrow \end{cases}$$

$(\mathscr{P}_1 \otimes^{\alpha}_{\|_L} \mathscr{P}_2)\, Q = 1_{\mathbb{B}}$

$\Rightarrow$ {Def. FuTS semantics of $\text{IML}_k$; Def. $\|_L$ over $\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$}

$\Leftarrow$ {Def. $\|_L$ over $\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{B})$}

$P_1 \overset{a}{\rightarrowtail} \mathscr{P}_1, P_2 \overset{a}{\rightarrowtail} \mathscr{P}_2, a \in L,$

$Q = Q_1 \ \|_L \ Q_2$ for some $Q_1, Q_2$ such that$(\mathscr{P}_1\, Q_1) = 1_{\mathbb{B}}, (\mathscr{P}_2\, Q_2) = 1_{\mathbb{B}}$

$\Rightarrow$ {I.H.}

$\Leftarrow$ {I.H.; Unicity of $\mathscr{P}_1$, $\mathscr{P}_2$}

$Q = Q_1 \ \|_L \ Q_2$ for some $Q_1, Q_2$ such that$P_1 \xrightarrow{a} Q_1, P_2 \xrightarrow{a} Q_2, a \in L$

$\substack{\Rightarrow \\ \Leftarrow}$ {Def. SOS of $\text{IML}_k$}

$P_1 \ \|_L \ P_2 \xrightarrow{a} Q, a \in L$

*Proof of part (ii).* We proceed by induction on the length of the derivation for $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}'$. Let $n \geq 1$ be the length of the derivation for proving $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}'$.

*Base case*: Trivial since the only case in which $P \overset{\delta^e}{\rightarrowtail} \mathscr{P}'$ can be derived with a proof of length 1 and $(\mathscr{P}'\, Q) \neq 0$ is $\lambda.Q \overset{\delta^e}{\rightarrowtail} [Q \mapsto \lambda]$ and $\mathbf{rt}(\lambda.Q, Q) = \lambda$ by the SOS definition of $\mathrm{IML}_k$ and the definition of $\mathbf{rt}$. In all other cases, $(\mathscr{P}'\, Q) = 0$ and there are no transitions $P \overset{\lambda}{\dashrightarrow} Q$, hence $\mathbf{rt}(P, Q) = 0$ by definition.

*Inductive step*: The last assert of any proof of length $n > 1$ must be of the form $P_1 + P_2 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1 + \mathscr{P}_2$, or $X \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1$, or $P_1 \parallel_L P_2 \overset{\delta^e}{\rightarrowtail} (\mathscr{P}_1 \parallel_L (\mathcal{X}\, \delta^e\, P_2)) + ((\mathcal{X}\, \delta^e\, P_1) \parallel_L \mathscr{P}_2)$.

*Case*: $P_1 + P_2 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1 + \mathscr{P}_2$

$(\mathscr{P}_1 + \mathscr{P}_2)\, Q$

$=$ {Def. $(\mathscr{P}_1 + \mathscr{P}_2)$}

$(\mathscr{P}_1\, Q) + (\mathscr{P}_2\, Q)$

$=$ {$P_1 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1, P_2 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_2$; I.H.}

$\mathbf{rt}(P_1, Q) + \mathbf{rt}(P_2, Q)$

$=$ {SOS definition of $\mathrm{IML}_k$; Def. of $\mathbf{rt}$}

$\mathbf{rt}(P_1 + P_2, Q)$

*Case*: $X \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1, X := P_1$

$(\mathscr{P}_1\, Q)$

$=$ {$P_1 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1$; I.H.}

$\mathbf{rt}(P_1, Q)$

$=$ {SOS definition of $\mathrm{IML}_k$; Def. of $\mathbf{rt}$}

$\mathbf{rt}(X, Q)$

*Case*: $P_1 \parallel_L P_2 \overset{\delta^e}{\rightarrowtail} (\mathscr{P}_1 \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\, \delta^e\, P_2)) + ((\mathcal{X}\, \delta^e\, P_1) \otimes^{\alpha}_{\parallel_L} \mathscr{P}_2)$

We observe that, from the FuTS semantics of $\mathrm{IML}_k$, if $Q$ is neither of the form $Q_1 \parallel_L P_2$, nor of the form $P_1 \parallel_L Q_2$ then $((\mathscr{P}_1 \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\, \delta^e\, P_2)) + ((\mathcal{X}\, \delta^e\, P_1) \otimes^{\alpha}_{\parallel_L} \mathscr{P}_2))\, Q = 0$. On the other hand, we observe that the only $\dashrightarrow$ transitions allowed by the SOS definition of $\mathrm{IML}_k$ are to terms of the form $Q_1 \parallel_L P_2$ or $P_1 \parallel_L Q_2$, so, also $\mathbf{rt}(P_1 \parallel_L P_2, Q) = 0$ if $Q$ is neither of the form $Q_1 \parallel_L P_2$, nor of the form $P_1 \parallel_L Q_2$. Let us assume, w.l.g., $Q$ be of the form $Q_1 \parallel_L P_2$; we have to consider two cases: (a) $Q_1 \neq P_1$, and (b) $Q_1 = P_1$.

*Case a)*:

$((\mathscr{P}_1 \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\, \delta^e\, P_2)) + ((\mathcal{X}\, \delta^e\, P_1) \otimes^{\alpha}_{\parallel_L} \mathscr{P}_2))\, Q_1 \parallel_L P_2$

$=$ {Def. $+$ over $\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0})$}

$((\mathscr{P}_1 \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\, \delta^e\, P_2))\, Q_1 \parallel_L P_2) + (((\mathcal{X}\, \delta^e\, P_1) \otimes^{\alpha}_{\parallel_L} \mathscr{P}_2)\, Q_1 \parallel_L P_2)$

$=$ {Def. $\otimes^{\alpha}_{\parallel_L}$ ; Def. $\mathcal{X}$}

$(\mathscr{P}_1\, Q_1)$

$=\quad \{P_1 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1;\text{ I.H.}\}$

$\mathbf{rt}(P_1, Q_1)$

$=\quad \{\text{SOS definition of IML}_k;\text{ Def. of }\mathbf{rt}\}$

$\mathbf{rt}(P_1 \parallel_L P_2, Q_1 \parallel_L P_2)$

*Case b)*:

$((\mathscr{P}_1 \otimes^{\alpha}_{\parallel_L} (\mathcal{X}\,\delta^e\, P_2)) + ((\mathcal{X}\,\delta^e\, P_1) \otimes^{\alpha}_{\parallel_L} \mathscr{P}_2))\, P_1 \parallel_L P_2$

$=\quad \{\text{Def. + over }\mathbf{FTF}(\mathcal{P}_{IML_k}, \mathbb{R}_{\geq 0})\}$

$((\mathscr{P}_1 \parallel_L (\mathcal{X}\,\delta^e\, P_2))\, P_1 \parallel_L P_2) + (((\mathcal{X}\,\delta^e\, P_1) \parallel_L \mathscr{P}_2)\, P_1 \parallel_L P_2)$

$=\quad \{\text{Def. }\otimes^{\alpha}_{\parallel_L};\text{ Def. }\mathcal{X}\}$

$(\mathscr{P}_1\, P_1) + (\mathscr{P}_2\, P_2)$

$=\quad \{P_1 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_1;\ P_2 \overset{\delta^e}{\rightarrowtail} \mathscr{P}_2\text{ I.H.}\}$

$\mathbf{rt}(P_1, \mathscr{P}_1) + \mathbf{rt}(P_2, \mathscr{P}_2)$

$=\quad \{\text{SOS definition of IML}_k;\text{ Def. of }\mathbf{rt}\}$

$\mathbf{rt}(P_1 \parallel_L P_2, P_1 \parallel_L P_2)$

The proof for the case in which $Q$ is of the form $P_1 \parallel_L Q_2$ is similar.

# References


[1] A. Aldini, M. Bernardo, and F. Corradini. *A Process Algebraic Approach to Software Architecture Design*. Springer, 2010. ISBN 978-1-84800-222-7.

[2] M. Bernardo. On the Expressiveness of Markovian Process Calculi with Durational and Durationless Actions. In A. Montanari, M. Napoli, and D. Parente, editors, *1st Int. Symp. on Games, Automata, Logics and Formal Verification (GANDALF 2010)*, volume 25 of *EPTCS*, pages 199–213. Minori, 2010.

[3] M. Bernardo and R. Gorrieri. A tutorial on EMPA: A theory of concurrent processes with nondeterminism, priorities, probabilities and time. *Theoretical Computer Science. Elsevier.*, 202(1-2):1–54, 1998.

[4] Marco Bernardo, Rocco De Nicola, and Michele Loreti. Uniform labeled transition systems for nondeterministic, probabilistic, and stochastic processes. In Martin Wirsing, Martin Hofmann, and Axel Rauschmayer, editors, *TGC*, volume 6084 of *Lecture Notes in Computer Science*, pages 35–56. Springer, 2010.

[5] Bard Bloom, Sorin Istrail, and Albert R. Meyer. Bisimulation can't be traced. *J. ACM*, 42(1):232–268, 1995.

[6] M. Bravetti, D. Latella, M. Loreti, M. Massink, and G. Zavattaro. Combining Timed Coordination Primitives and Probabilistic Tuple Spaces. In C. Kaklamanis and F. Nielson, editors, *Trustworthy Global Computing, 4th International Symposium, TGC 2008, Barcelona, Spain, November 3-4, 2008, Revised Selected Papers*, volume 5474 of *LNCS*, pages 52–68. Springer, 2009.

[7] E. Brinksma and H. Hermanns. Process Algebra and Markov Chains. In Brinksma et al. [8], pages 181–231.

[8] E. Brinksma, H. Hermanns, and J. Katoen, editors. *Lectures on Formal Methods and Performance Analysis*, volume 2090 of *LNCS*. Springer-Verlag, 2001.

[9] L. Cardelli and R. Mardare. The Measurable Space of Stochastic Processes. In *Seventh International Conference on the Quantitative Evaluation of Systems (QEST 2010)*, pages 171–180. IEEE Computer Society Press, 2010. ISBN 978-0-7695-4188-4.

[10] Dang Van Hung and Zhou Chaochen. Probabilistic Duration Calculus for Continuous Time. *Formal Aspects of Computing. The International Journal of Formal Methods. Springer-Verlag*, 11:21–44, 1999.

[11] R. De Nicola, J.-P. Katoen, D. Latella, M. Loreti, and M. Massink. KLAIM and its Stochastic Semantics. Technical Report 6, Dipartimento di Sistemi e Informatica, Università di Firenze, 2006. (available at `http://rap.dsi.unifi.it/~loreti/papers/TR062006.pdf` ).

[12] R. De Nicola, J.-P. Katoen, D. Latella, M. Loreti, and M. Massink. Model Checking Mobile Stochastic Logic. *Theoretical Computer Science. Elsevier.*, 382(1):42–70, 2007. http://dx.doi.org/10.1016/j.tcs.2007.05.008.

[13] R. De Nicola, D. Latella, M. Loreti, and M. Massink. MarCaSPiS: a Markovian Extension of a Calculus for Services. In M. Hennessy and B. Klin, editors, *Proceedings of the 5th Workshop on Structural Operational Semantics (SOS 2008)*, volume 229 of *Electronic Notes in Theoretical Computer Science*, pages 11–26. Elsevier Science Publishers B.V., 2009. ISSN 1571-0661.

[14] R. De Nicola, D. Latella, M. Loreti, and M. Massink. On a Uniform Framework for the Definition of Stochastic Process Languages. In M. Alpuente, B. Cook, and C. Joubert, editors, *Formal Methods for Industrial Critical Systems - FMICS 2009*, volume 5825 of *LNCS*, pages 9–25. Springer-Verlag, 2009.

[15] R. De Nicola, D. Latella, M. Loreti, and M. Massink. Rate-based Transition Systems for Stochastic Process Calculi. In Albers S., A. Marchetti-Spaccamela, Y. Matias, S. Nikoletseas, and W. Thomas, editors, *Automata, Languages and Programming. Part II*, volume 5556 of *LNCS*, pages 435–446. Springer-Verlag, 2009. ISBN 978-3-642-02929-5.

[16] R. De Nicola, D. Latella, and M. Massink. Formal modeling and quantitative analysis of KLAIM-based mobile systems. In H. Haddad, L. Liebrock, A. Omicini, R. Wainwright, M. Palakal, M. Wilds, and H. Clausen, editors, *APPLIED COMPUTING 2005. Proceedings of the 20th Annual ACM Symposium on Applied Computing*, pages 428–435. Association for Computing Machinery - ACM, 2005. ISBN 1-58113-964-0.

[17] E. De Vink and J. Rutten. Bisimulation for probabilistic transition systems: a coalgebraic approach. *Theoretical Computer Science. Elsevier.*, (221):271–293, 1999.

[18] Y. Deng, R. van Glabbeek, M. Hennessy, C. Morgan, , and C. Zhang. Characterising testing preorders for finite probabilistic processes. In *IEEE Symposium on Logic in Computer Science*, pages 313–325. IEEE, Computer Society Press, 2007.

[19] N. Gotz, U. Herzog, and M. Rettelbach. Multiprocessor and distributed systems design: The integration of functional specification and performance analysis using stochastic process algebras. In L. Donatiello and R. Nelson, editors, *Performance Evaluation of Computer and Communication Systems. - Joint Tutorial Papers of Performance '93 and Sigmetrics '93*, volume 729 of *LNCS*. Springer-Verlag, 1993.

[20] B. Haverkort. Markovian Models for Performance and Dependability Evaluation. In Brinksma et al. [8], pages 38–83.

[21] H. Hermanns. *Interactive Markov Chains*. Springer-Verlag, Berlin/New York, 2002. LNCS 2428.

[22] H. Hermanns, U. Herzog, and J.-P. Katoen. Process algebra for performance evaluation. *Theoretical Computer Science. Elsevier.*, 274(1-2):43–87, 2002.

[23] H. Hermanns, U. Herzog, and V. Mertsiotakis. Stochastic process algebras - between LOTOS and Markov chains. *Computer Networks and ISDN Systems. North-Holland*, 30:901–924, 1998.

[24] H. Hermanns and S. Johr. Uniformity by Construction in the Analysis of Nondeterministic Stochastic Systems. In *2007 International Conference on Dependable Systems & Networks*, pages 718–728. IEEE Computer Society Press, 2007.

[25] J. Hillston. The nature of synchronisation. In U. Herzog and M. Rettelbach, editors, *Proceedings of the 2nd workshop on process algebras and performance modeling*, pages 51–70, July 1994. Erlangen, Germany - ISSN 0344-3515.

[26] J. Hillston. A compositional approach to performance modelling, 1996. Distinguished Dissertation in Computer Science. Cambridge University Press.

[27] C.A.R. Hoare. *Communicating Sequential Processes*. Series in Computer Science. Prentice Hall, 1985.

[28] H. Hojjat, M. Reza, Mousavi, and M. Sirjani. A framework for performance evaluation and functional verification in stochastic process algebras. In R. Wainwright and H. Haddad, editors, *APPLIED COMPUTING 2008. Proceedings of the 23th Annual ACM Symposium on Applied Computing*, pages 339–346. Association for Computing Machinery - ACM, 2008.

[29] B. Klin and V. Sassone. Structural Operational Semantics for Stochastic Process Calculi. In R. Amadio, editor, *FoSSaCS 2008*, volume 4962 of *LNCS*, pages 428–442. Springer-Verlag, 2008.

[30] R. Knast. Continuous-Time Probabilistic Automata. *Information and Control*, 15:335–352, 1969.

[31] Lijun Zhang, H. Hermanns, F. Eisenbrand, and D. Jansen. Flow Faster: Efficient Decision Algorithms For Probabilistic Simulations. *Logical Methods in Computer Science*, 4(6):1–43, 2008.

[32] R. Milner. *Communication and Concurrency*. Series in Computer Science. Prentice Hall, 1989.

[33] D. Prandi and P. Quaglia. Stochastic COWS. In B. Krämer and K. Lin, editors, *Service Oriented Computing - ICSOC 2007*, volume 4749 of *LNCS*, pages 245–256. Springer-Verlag, 2007.

[34] C. Priami. Stochastic $\pi$-Calculus. *The Computer Journal. Oxford University Press.*, 38(7):578–589, 1995.