Embedding Real Time in Stochastic Process Algebras

J. Markovski⋆ and E.P. de Vink

Technische Universiteit Eindhoven, Formal Methods Group
Den Dolech 2, 5612 AZ, Eindhoven, The Netherlands

Abstract. We present a stochastic process algebra including immediate actions, deadlock and termination, and explicit stochastic delays, in the setting of weak choice between immediate actions and passage of time. The operational semantics is a spent time semantics, avoiding explicit clocks. We discuss the embedding of weak-choice real-time process theories and analyze the behavior of parallel composition in the weak choice framework.
Keywords. Stochastic delay, weak choice, race condition, real-time and stochastic process algebra.

1 Introduction

Traditionally, *process algebras* (PAs) like ACP, CCS and CSP are used for qualitative description and verification of processes. In this setting, process behaviour is reflected by the order of actions. However, untimed description of processes is frequently not sufficiently expressive. (See, e.g., [1].) Thus, several timed extensions of traditional PAs emerged. (A detailed overview can be found in [2].) Also, probabilistic behavior of processes was included in PAs supporting probabilistic analysis. (Cf. [3], for example.) Combined efforts, like [4], considering timing aspects and probability, are reported as well.

Often, real-world processes require stochastic behaviour to be incorporated in their description. Early PAs doing so, employed exponentially distributed stochastic delays. Modeling with exponential distributions greatly simplifies the treatment of parallel composition, because of the memoryless property. Prominent Markovian PAs include EMPA, PEPA and Algebra of IMC [5–7]. The first two associate exponential rates with actions, whereas the latter clearly distinguishes between actions and rates.

Although much success has been reported, an abundance of processes cannot be dealt with exponentially. Consequently, several stochastic PAs with general distributions are proposed like SPADES, IGSMP and NMSPA [8–10]. SPADES introduces clocks to record the residual lifetime of stochastic delays. Each clock initialization is governed by a general distribution. Actions are only enabled after all clocks from a particular set have expired. Semantics for SPADES is given in terms of stochastic automata [11]. IGSMP uses clocks to record spent lifetimes. The clocks have an associated expiration time distribution. When a clock

⋆ Corresponding author, `j.markovski@tue.nl`. Supported by Bsik-project BRICKS AFM 3.2.

expires other clocks are redistributed according to the time that has passed. IGSMP semantics is given using generalized semi-Markov processes extended with actions. An interesting feature is the definition of the alternative composition modeled as a probabilistic choice between differently distributed clocks. NMSPA exploits random variables for the distribution of stochastic delays of actions. Also here, expiration of a stochastic delay induces redistribution of other variables according to the time that has passed. The semantics is given in terms of transition systems. The alternative composition is defined over an arbitrary number of summands in order to achieve maximal progress for internal actions. In NMSPA alternative composition of discrete stochastic delays followed by an internal action represents an inherent probabilistic choice. Other stochastic PAs that we mention here are the extension of LOTOS for performance analysis of distributed systems, the stochastic π-calculus and TIPP [12–14]. More details can be found in the overview papers [15, 16].

The main goal of our paper is to deal with standard real-time in stochastic PAs with an semantics that exploits spent-time and avoids explicit clocks. Our aims is to report on preliminary research on the conservative extension of real-time process algebra where delays are governed by probabilistic distributions. To this end, we consider a stochastic PA with immediate actions, deadlock and termination. We model stochastic delays as timed delays guided by discrete random variables, as we wish to distinguish between actions and stochastic delays, similar to IMC [7]. The alternative composition implements weak choice between immediate actions and passage of time similar to real-time PAs in the style of [1]. Here, we give the semantics in terms of stochastic transition systems. In comparison to other stochastic PA our approach is closest to NMSPA. Unlike NMSPA, we define alternative composition on two processes rather than on arbitrary sums and, in our setting, the alternative composition makes no choice in case both summands can delay together as in the real-time PAs. We propose an appropriate version of stochastic bisimulation for our setting, which is a congruence. α-conversion is introduced to pave the way for a treatment of the parallel operator. However, as we show, no expansion law is available in this set-up. We justify, via an embedding of transition systems, the proposed stochastic process algebra being called an extension of real-time process algebra. In our present work, we consider only discrete stochastic delays, mainly because they almost effortlessly model real-time delays as degenerated discrete random variables. Also, as a technical convenience, they allow two different delays to have the same duration, a property not shared by continuous distributions.

Related work Surprisingly, there is not much work on embedding real-time into stochastic time PAs. Markovian PAs cannot embed real-time because they employ exponential distributions only. The extension of LOTOS for performance analysis is an extension of timed LOTOS with stochastic timers, but there are strong syntax restrictions and no embedding is given. We remind the reader that the semantics of SPADES [8] is given in terms of stochastic automata [11]. A structural translation from stochastic automata to timed automata with deadlines is given in [17]. The translation is shown to preserve timed traces, so

SPADES can imitate real-time behaviour. There is a translation from IGSMP into pure real-time models termed Interactive Timed Automata (ITA) [9].

The rest of this paper is organized as follows. Section 2 gives the mathematical background for the stochastic delays. Section 3 introduces a basic stochastic PA with alternative composition and stochastic delay prefix. Section 4 provides the transition system and a notion of stochastic bisimulation, for which congruence properties are given. We define in Section 5 a variant of α-conversion to support the operational semantics. Sections 6 and 7 discuss the parallel operator and the embedding of real-time process theories. Section 8 wraps up with concluding remarks. For the complete proofs we refer to the full version of the paper [19].

2 Preliminaries

We denote the set of discrete random variables by $\mathcal{V}$. For $S \subseteq \mathcal{V}$, $y \in \mathbb{R}$ and $\diamond$ either $<, >, =$, we write $S \diamond y$ for $X \diamond y$, $X \in S$. We use X, Y and Z for random variables and $F_X(t)$, $F_Y(t)$ and $F_Z(t)$, for $t \geq 0$, for their distribution functions, unless stated otherwise. For durations of a stochastic delay we have $F_X(t) = 0$ for $t < 0$ and we denote the set of such discrete distribution functions by $\mathcal{F}_d$. The support set of random variable X, denoted by $\mathrm{supp}(X)$ contains the values for which $P(X = t) > 0$. By $\overline{F}_X(t)$ we denote the residual distribution function $1 - F_X(t)$. We extend the notion of support set to a set S of random variables by $\mathrm{supp}(S) = \bigcap_{X \in S} \mathrm{supp}(X)$.

A *stochastic delay* is a time delay which duration is guided by a random variable. It is discrete if the random variable is discrete. The notions of stochastic delay and random variable are used interchangeably depending on the context. We observe simultaneous passage of time for a number of stochastic delays until at least one of their duration passes. This phenomenon is referred to as the *race condition*. In general, simultaneous multiple stochastic delays can be observed as being the shortest; the shortest duration itself can be different and provided by different delays in different observations. Observing several stochastic delays we call a *race*. The stochastic delay or delays that have the shortest duration are called 'winners'. The other ones are called 'losers' of the race.

In general, if one observes a race of a set of random variables $V \subseteq \mathcal{V}$, the resulting delay of the race will be distributed as the minimum $\min(V)$ of these random variables with a distribution function $F_{\min(V)}(t) = 1 - \prod_{X \in V} \overline{F}_X(t)$. The probability that the winners are in the set $W \subseteq V$ is

$$P(W = \min(V)) = \textstyle\sum_{t \in \mathrm{supp}(W)} P(W = t, (V \setminus W) > t).$$

The stochastic delay performed by the winners, is distributed as

$$P(\langle\, X \mid W = \min(V) \,\rangle = t) = \frac{P(W = t, (V \setminus W) > t)}{P(W = \min(V))},$$

for any $X \in W$. We use angle brackets to denote conditional random variables.

Because of associativity and commutativity of the minimum of random variables, it holds that simultaneous observation of all delays amounts to the same as iterated observation of disjoint sets.

3 Basic Processes with Discrete Stochastic Time

In this section we introduce $\mathrm{BSP}^{\mathrm{dst}}(\mathcal{A}, \mathcal{V})$, a stochastic PA with immediate actions, termination and deadlock, that implements weak choice between actions and time. We refer to $\mathrm{BSP}^{\mathrm{dst}}$ as Basic Process Theory with Discrete Stochastic Time. The terminology is adopted from [18] and we build on the untimed version $\mathrm{BSP}(\mathcal{A})$. Here, $\mathcal{A}$ is the set of actions and $\mathcal{V}$ is the set of random variables. A new unary operator scheme $\sigma_X.\_$ for $X \in \mathcal{V}$ represents stochastic delays.

The process $\sigma_X.p$ executes a stochastic delay guided by the random variable X and continues behaving as p. Because of the race condition, one cannot observe the execution of a stochastic delay in isolation. Informally, an example of a transition system that corresponds to a race between two discrete stochastic delays is depicted in Fig. 1.

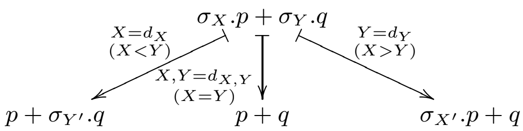

Fig. 1. Race condition

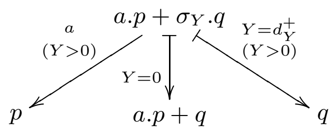

Fig. 2. Weak choice

The relations in the brackets give the condition that enables the transition. Each $\mapsto$ transition represents a stochastic delay. The label shows the winners of the race and their observed duration. The duration is determined by the support set of the winning delay. For clarity, we represent all the transitions by a single transition scheme. For example, the transitions of the stochastic delay guided by X in Fig. 1 are represented by one transition scheme labeled by X and d_X. The observed winning duration d_X takes its values from $\mathrm{supp}(\langle\, X \mid X < Y \,\rangle)$. Thus, the transition scheme replaces $|\, \mathrm{supp}(\langle\, X \mid X < Y \,\rangle) \,|$ different transitions, each executed with its own probability.

When considering the interaction of action transitions and termination versus stochastic delay, we employ weak choice, i.e. a non-deterministic choice between immediate actions, termination and passage of time. The alternative composition depicted in Fig. 2 allows execution of the stochastic delay in the rightmost transition even though the choice is made between an immediate action and passage of time. As a consequence, the losers of the race become dependent on the amount of time that has passed for the winners as in Fig. 1. Thus, the random variables of the remaining stochastic delays do not retain their initial distributions. Another issue we consider is the interaction between immediate actions and zero duration delays. Similar to the timed process theories [1, 2] we take zero duration not to disable immediate actions, as depicted by the middle transition in Fig. 2. Note that the immediate action is enabled only if $F_Y(0) \neq 1$. In order to distinguish between zero and non-zero transitions, we use the notation d_X^+ to denote only positive durations.

In an alternative composition of two stochastic delays, we obtain three transitions. In case the winner is the first summand, one obtains the leftmost transition. The rightmost transition is obtained when the winner is the second summand. The middle transition shows that both delays win the race together with non-zero probability. In this case, the race cannot determine one winner and passage of time does not determine a choice similar as for the real-time setting.

In Fig. 1, the altered probability distributions of X and Y are denoted by X' and Y', respectively. They are termed 'aged' probability distributions of X and Y by the duration d_Y and d_X, respectively. The probability distribution of X' is the aged probability distribution of X by d_Y given by

$$F_{X'}(t) = P(X \leq t \mid X > Y, Y = d_Y) = \frac{F_X(t + d_Y) - F_X(d_Y)}{1 - F_X(d_Y)}.$$

In order to calculate the actual distribution functions in each state, we require the original distribution function and its age. In order to keep track of the ages of the stochastic delays we introduce an environment to the transition system. The basic idea underlying the environments is that they store the actual distribution function of the random variables. The following definition and property of aging justify the use of environments.

Definition 1. *A distribution function F can be 'aged' by a time duration $d \geq 0$ if $F(d) < 1$. The resulting distribution $F|d$ is $(F|d)(t) = \frac{F(t+d)-F(d)}{1-F(d)}$.*

If the conditions of Definition 1 are fulfilled, then $F|d$ is again a distribution function. We have that iterative application of the aging function is the same as aging the function once by the sum of the time durations (for proof see [19]), i.e.

$$(\ldots(F|d_1)\ldots)|d_n = F|\left(\textstyle\sum_{i=1}^{n} d_i\right).$$

Using this property one easily calculates the age of the losers after each stochastic delay transition by adding the duration for the winners to the existing ages.

The environment is implemented using two injective functions: $\Phi\colon \mathcal{V} \to \mathcal{F}_d$ for the distribution functions and $\Delta\colon \mathcal{V} \to \mathbb{R}_0^+ \cup \{\bot\}$ for the age of the stochastic delays. We add the special symbol $\bot$ to denote that no time has passed for the stochastic delay, i.e. the delay has not participated in a race yet. Note that this is not the same as saying that the delay is of age zero. Having age zero means that the variable has lost a race with a zero duration and, ultimately, that disabled its possible zero duration transitions. Thus, we have to extend the domain of $|$ to $|\colon \mathcal{F}_d \times (\mathbb{R}_0^+ \cup \{\bot\}) \to \mathcal{F}_d$. We put $F|\bot = F$, $x + \bot = x$, for $x \in \mathbb{R}_0^+$, and we write $\mathbb{R}_\bot^+$ instead of $\mathbb{R}_0^+ \cup \{\bot\}$. We consider a well-defined environment to be a pair of two injective functions $(\Phi, \Delta) \in \mathcal{F}_d^{\mathcal{V}} \times {\mathbb{R}_\bot^+}^{\mathcal{V}}$ such that for all $X \in \mathcal{V}$ the probability distribution function $\Phi(X) \mid \Delta(X)$ is defined. The set of well-defined environments is denoted by Env. Next, we introduce the signature of $\mathrm{BSP}^{\mathrm{dst}}$ and describe its constants and operators.

Definition 2. *The signature of* $\mathrm{BSP}^{\mathrm{dst}}$ *contains the two constants δ and ϵ, the two unary operator schemes $a.\_$, for $a \in A$ and $\sigma_X.\_$, for $X \in \mathcal{V}$ and the binary operator $\_ + \_$. The syntax of* $\mathrm{BSP}^{\mathrm{dst}}$ *is given by*

$$P ::= \delta \mid \epsilon \mid a.P \mid \sigma_X.P \mid P + P,$$

with $a \in A$ and $X \in \mathcal{V}$. The set of closed terms over the signature of $\mathrm{BSP}^{\mathrm{dst}}$ *is denoted by* $\mathcal{C}(\mathrm{BSP}^{\mathrm{dst}})$ *and it is ranged over by p, q and r.*

We adopt the signature from $\mathrm{BSP}(\mathcal{A})$ [18] where immediate constants and actions are denoted by $\tilde{\tilde{\delta}}$, $\tilde{\tilde{\epsilon}}$ and $\tilde{\tilde{a}}$. However, here, we do not use the $\approx$-notation. The constant δ represents an immediate deadlock which does not allow passage of time. Immediate termination ϵ terminates without allowing any time to pass. The unary operator scheme $a.p$, for $a \in A$, comprises processes that execute the action a without consuming any time and continue behaving as p. The unary operator scheme $\sigma_X.p$ provides processes that execute a stochastic delay guided by the random variable X and afterwards continue behaving as p. The alternative composition behaves differently depending on three different contexts. It makes a non-deterministic choice between actions, a weak choice between actions, successful termination and stochastic delays, and imposes a race condition on stochastic delays.

4 Structural Operational Semantics

First, we define a *stochastic transition system* (STS) that deals with aging of distributions as informally discussed in the example of Fig. 1. The transitions of the STS are performed in an environment that keeps track of the up-to-date distribution functions of the racing stochastic delays. It contains the distribution functions for the random variables and the age of the delays.

Definition 3. *STS is a structure $STS = (\mathcal{S}, (\Phi, \Delta), \rightarrow, \mapsto, \downarrow)$ where*

- *$\mathcal{S}$ is a set of states labeled by closed $\mathrm{BSP}^{\mathrm{dst}}$-terms;*
- *$(\Phi, \Delta) \in \mathrm{Env}$ is a well-defined environment;*
- *$\rightarrow \subseteq \mathcal{S} \times \mathrm{Env} \times \mathcal{A} \times \mathcal{S} \times \mathrm{Env}$ is a labeled transition relation;*
- *$\mapsto \subseteq \mathcal{S} \times \mathrm{Env} \times 2^{\mathcal{V}} \times \mathbb{R}_0^+ \times \mathcal{S} \times \mathrm{Env}$ is a stochastic delay (probabilistic) transition relation;*
- *$\downarrow \subseteq \mathcal{S}$ is an immediate termination predicate.*

For $\rightarrow$ and $\mapsto$ we will use infix notation. By $\langle p, (\Phi, \Delta)\rangle \xrightarrow{a} \langle p', (\Phi, \Delta')\rangle$ we denote that a process term p in the environment (Φ, Δ) does an action transition with the label a to the term p' and changes the environment to (Φ, Δ'). By $\langle p, (\Phi, \Delta)\rangle \overset{S}{\mapsto}_{d_S} \langle p', (\Phi, \Delta')\rangle$ we denote that a term p in the environment (Φ, Δ) exhibits a passage of time of duration d_S, transforms to p' and changes the environment to (Φ, Δ'). The observed time is a result of a race won by the set of stochastic delays that are guided by the set of random variables S. The possible durations of the winners are determined by $d_S \in \mathrm{supp}(\langle\ X \mid S = \min(\mathrm{rd}(p))\ \rangle)$, where $\mathrm{rd}(p)$ (we define this function later) is the set of racing delays of p and $X \in S$. In case there is a separation between zero duration and non-zero duration, we denote the non-zero durations by $d_S^+ > 0$. The random variables $X \in \mathcal{V}$ obtain their probability distributions as $F_X = \Phi(X) \mid \Delta(X)$. The race changes the age binding function Δ by setting age $\perp$ to every winning stochastic delay and increasing the ages of the losing delays by d_S.

Since all transitions only change the 'age parameter' Δ that assigns the ages, we suppress Φ and use the shorthand Δ for the environment (Φ, Δ). The STS represents a scheme because we leave implicit the conditions that enable the transitions and we parameterize multiple delay transitions by their support set. Also, we write X for $\{X\}$ and d_X for $d_{\{X\}}$ in the transition labels. We introduce the the set of all age parameters as $\mathrm{Del} = {\mathbb{R}^{+}_{\perp}}^{\mathcal{V}}$. In case we wish to give a transition system for a specific term $p \in \mathcal{C}(\mathrm{BSP^{dst}})$ we write $STS(p, (\Phi, \Delta_0))$, where (Φ, Δ_0) is the initial environment. We denote the set of STS's as $\mathcal{STS}$.

The sets of winning stochastic delays are given as labels of the probabilistic transitions. However, not all stochastic delays participate in a race at the same time. So, we have to identify only the racing stochastic delays, i.e. the ones that participate in the race. A function named $\mathrm{rd} \colon \mathcal{C}(\mathrm{BSP^{dst}}) \to 2^{\mathcal{V}}$ extracts the random variables that guide the racing delays of a process term. They are identified as all stochastic delays that are directly connected by alternative composition.

$$\mathrm{rd}(\epsilon) = \emptyset \quad \mathrm{rd}(a.p) = \emptyset \quad \mathrm{rd}(\delta) = \emptyset \quad \mathrm{rd}(\sigma_X.p) = \{X\} \quad \mathrm{rd}(p+q) = \mathrm{rd}(p) \cup \mathrm{rd}(q)$$

In order to provide a concise presentation of the operational semantics, we define two functions res and age which alter the age parameter Δ of the environment. The function res resets the images of the winners to $\perp$, whereas age ages the losers by the duration observed for the winners.

Definition 4. *For an environment Δ, a set of winners $W \subseteq \mathcal{V}$ and a set of losers $L \subseteq \mathcal{V}$ of a race of duration d, the functions* $\mathrm{res} \colon \mathrm{Del} \times 2^{\mathcal{V}} \to \mathrm{Del}$ *and* $\mathrm{age} \colon \mathrm{Del} \times 2^{\mathcal{V}} \times \mathbb{R}^{+}_{0} \to \mathrm{Del}$ *are defined as*

$$\mathrm{res}(\Delta, W) = \begin{cases} \Delta(X) & \text{if } X \notin W \\ \perp & \text{if } X \in W \end{cases} \qquad \mathrm{age}(\Delta, L, d) = \begin{cases} \Delta(X) & \text{if } X \notin L \\ \Delta(X) + d & \text{if } X \in L. \end{cases}$$

Next, we give the structural operational semantics for $\mathrm{BSP^{dst}}$.

$$1\ \langle \epsilon, \Delta \rangle \downarrow \quad 2\ \frac{\langle p, \Delta \rangle \downarrow}{\langle p+q, \Delta \rangle \downarrow} \quad 4\ \langle a.p, \Delta \rangle \xrightarrow{a} \langle p, \Delta \rangle \quad 5\ \langle \sigma_X.p, \Delta \rangle \overset{X}{\mapsto}_{d_X} \langle p, \mathrm{res}(\Delta, \{X\}) \rangle$$

$$6\ \frac{\langle p, \Delta \rangle \xrightarrow{a} \langle p', \Delta' \rangle,\ \langle q, \Delta \rangle \not\mapsto}{\langle p+q, \Delta \rangle \xrightarrow{a} \langle p', \Delta' \rangle} \qquad 7\ \frac{\langle p, \Delta \rangle \xrightarrow{a} \langle p', \Delta' \rangle,\ \langle q, \Delta \rangle \overset{T}{\mapsto}_{d_T^+} \langle q', \Delta'' \rangle}{\langle p+q, \Delta \rangle \xrightarrow{a} \langle p', \mathrm{res}(\Delta', \mathrm{rd}(q)) \rangle}$$

$$10\ \frac{\langle p, \Delta \rangle \overset{S}{\mapsto}_0 \langle p', \Delta' \rangle,\ \langle q, \Delta \rangle \not\mapsto}{\langle p+q, \Delta \rangle \overset{S}{\mapsto}_0 \langle p'+q, \Delta' \rangle} \qquad 11\ \frac{\langle p, \Delta \rangle \overset{S}{\mapsto}_{d_S^+} \langle p', \Delta' \rangle,\ \langle q, \Delta \rangle \not\mapsto}{\langle p+q, \Delta \rangle \overset{S}{\mapsto}_{d_S^+} \langle p', \Delta' \rangle}$$

$$14\ \frac{\langle p, \Delta \rangle \overset{S}{\mapsto}_{d_S} \langle p', \Delta' \rangle,\ \langle q, \Delta \rangle \overset{T}{\mapsto}_{d_T} \langle q', \Delta'' \rangle,\ d_S < d_T}{\langle p+q, \Delta \rangle \overset{S}{\mapsto}_{d_S} \langle p'+q, \Delta''' \rangle}, \quad \text{where } \Delta''' = \mathrm{age}(\Delta', \mathrm{rd}(q), d_S)$$

$$16\ \frac{\langle p, \Delta \rangle \overset{S}{\mapsto}_{d_S} \langle p', \Delta' \rangle,\ \langle q, \Delta \rangle \overset{T}{\mapsto}_{d_T} \langle q', \Delta'' \rangle,\ d_S = d_T}{\langle p+q, \Delta \rangle \overset{S \cup T}{\mapsto}_{d_{S \cup T}} \langle p'+q', \Delta''' \rangle}, \quad \text{where } \Delta''' = \mathrm{res}(\mathrm{age}(\Delta, \mathrm{rd}(p+q), d_{S \cup T}), S \cup T)$$

Rules 1, 2, and 4 are the standard rules for termination and action prefix. Rule 5 states that stochastic delays $\sigma_X.p$ allow passage of time sampling from $\Phi(X)|\Delta(X)$. The non-deterministic choice made by action transitions from the first summand is shown by Rule 6 when the second summand cannot do a stochastic delay and by Rule 7 when it can do a stochastic delay with non-zero duration. Rule 10 states that zero delay of p does not enforce a choice, still allowing action transitions from q. In case p does perform a non-zero delay as in Rule 11 weak choice is enabled between action transitions and passage of time, where passage of time disables the action transitions of q. Rule 14 describes the race in case when the first summand wins the race. The winners given by the set S perform a stochastic delay transition with duration d_S. The racing delays of the losing summand $(\mathrm{rd}(q))$ are aged by d_S using the function *age* and the environment of the winner Δ' (in which the losers of the first summand are already aged). Note that since the second summand can perform a stochastic delay $d_T > d_S$, the aging of its racing delays is allowed. Rule 16 states that if both summands have stochastic delays that can win with the same duration, the joint race enabled by the alternative composition can be won by the union of the winners of the both summands. The new environment is obtained by aging all racing delays of both summands in the original environment and resetting the winners. (Because of lack of space we omit the symmetric rules 3, 8, 9, 12, 13 and 15, analogous to 2, 6, 7, 10, 11 and 14.)

Next, we define when two STS's are bisimilar. Intuitively, two STS should be bisimilar if related states (1) do the same action transitions, (2) have the same termination options and (3) go to another class of states with the same accumulative probability of performing a stochastic delay with the same duration. The following definition defines the accumulative probability of (3).

Definition 5. *Let R be an equivalence relation on $\mathcal{S} \times \mathrm{Env}$, $C \in \big(\mathcal{S} \times \mathrm{Env}\big)/R$ an arbitrary class and $(\Phi, \Delta) \in \mathrm{Env}$, where $\mathcal{S} \subseteq \mathcal{C}(\mathrm{BSP}^{\mathrm{dst}})$. By $\mathrm{ws}(p, \Delta, C, d)$ we define the set of sets of winning stochastic delays that p can do in time d and afterwards transform into a process that belongs to the class C, i.e.*

$$\mathrm{ws}(p, \Delta, C, d) = \bigcup_{\langle p', \Delta' \rangle \in C} \{\, S \subseteq \mathrm{rd}(p) \mid \langle p, \Delta \rangle \overset{S}{\mapsto}_d \langle p', \Delta' \rangle \,\}.$$

The accumulative probability of doing a transition from a term to an equivalence class in time d is given as

$$\mathrm{ap}(p, \Delta, C, d) = \begin{cases} 0 & \mathrm{ws}(p, \Delta, C, d) = \emptyset \\ \sum_{S \in \mathrm{ws}(p, \Delta, C, d)} P(S = \min(\mathrm{rd}(p)), S = d) & \mathrm{ws}(p, \Delta, C, d) \neq \emptyset. \end{cases}$$

Next, we define strong bisimulation on STS's.

Definition 6. *A strong bisimulation on $STS = (\mathcal{S}, (\Phi, \Delta), \to, \mapsto, \downarrow)$ is an equivalence relation R on $\mathcal{S} \times \mathrm{Env}$ such that the following conditions hold:*

1. *if $\langle p, \Delta \rangle \xrightarrow{a} \langle p', \Delta' \rangle$, then $\langle q, \Delta \rangle \xrightarrow{a} \langle q', \Delta' \rangle$, such that $(\langle p', \Delta' \rangle, \langle q', \Delta' \rangle) \in R$,*
2. *for all $d \geq 0$, it holds that $\mathrm{ap}(p, \Delta, C, d) = \mathrm{ap}(q, \Delta, C, d)$,*

3. if $\langle p, \Delta\rangle \downarrow$ then $\langle q, \Delta\rangle \downarrow$,

for all $p, p', q, q' \in \mathcal{S}$ and all $C \in (\mathcal{S} \times \mathrm{Env})_{/R}$ such that $(\langle p, \Delta\rangle, \langle q, \Delta\rangle) \in R$.

Note that the second transfer condition implies that after doing a stochastic delay, both terms must result again in bisimilar terms. If $\langle p, \Delta\rangle$ and $\langle q, \Delta\rangle$ are related by a strong bisimulation we write $\langle p, \Delta\rangle \leftrightarroweq \langle q, \Delta\rangle$. We also note, that if we consider the time duration as a constant in the transition system, we obtain the probabilistic bisimulation given in [20].

Theorem 7. *The bisimulation relation $\leftrightarroweq$ is a congruence. [19]*

5 α-conversion

We proceed by analyzing a conflicting behaviour of the STSs defined so far that occurs when two racing delays are guided by the same random variable. Consider the following example.

Example 8. Suppose $p \equiv \sigma_X.\epsilon$. We observe $STS(p + p, (\Phi, \Delta))$. Consider the transition $\langle \sigma_X.\epsilon + \sigma_X.\epsilon, \Delta\rangle \overset{X}{\mapsto}_{d_X} \langle \epsilon + \sigma_X.\epsilon, \mathrm{res}(\mathrm{age}(\Delta, \{X\}, d_X), X)\rangle$. In the resulting environment, X is a random variable that guides both the winning and the losing stochastic delay. Such behavior leads to conflict because $\Delta(X)$ should contain both, $\perp$, because X won the race and d_X, because X lost the race. On the other hand, the term $p+p$ is not bisimilar to $\sigma_X.(\epsilon+\epsilon)$ because, in general, the distribution functions of X and $\min(X, X)$ are not equal. Therefore, we wish to express that the left and the right summand have equally distributed stochastic delays and the distribution function is provided by the random variable X.

We resolve the conflict by renaming one of the variables and ensuring that the original and the replacement have the same distribution. So, $\sigma_X.\epsilon + \sigma_X.\epsilon$ and $\sigma_X.\epsilon + \sigma_Y.\epsilon$ behave the same under the assumption that $F_X = F_Y$, because the behavior of the STS does not depend on the name of the variable, but on its distribution function. However, the second term has proper semantics, since there is no conflicting behavior in its STS. For an technical underpinning of this, we define the relation $\simeq_\alpha$ on $\mathcal{C}(\mathrm{BSP}^{\mathrm{dst}}) \times \mathrm{Env}$ as the least relation such that

$$\langle \delta, \Delta\rangle \simeq_\alpha \langle \delta, \Delta\rangle \qquad \langle \epsilon, \Delta\rangle \simeq_\alpha \langle \epsilon, \Delta\rangle \qquad \frac{\langle p, \Delta\rangle \simeq_\alpha \langle q, \Delta\rangle}{\langle a.p, \Delta\rangle \simeq_\alpha \langle a.q, \Delta\rangle}$$

$$\frac{\langle p, \Delta\rangle \simeq_\alpha \langle q, \Delta\rangle, \Phi(X) = \Phi(Y), \Delta(X) = \Delta(Y)}{\langle \sigma_X.p, \Delta\rangle \simeq_\alpha \langle \sigma_Y.q, \Delta\rangle} \qquad \frac{\langle p, \Delta\rangle \simeq_\alpha \langle q, \Delta\rangle, \langle p', \Delta\rangle \simeq_\alpha \langle q', \Delta\rangle}{\langle p + p', \Delta\rangle \simeq_\alpha \langle q + q', \Delta\rangle}$$

Clearly, $\simeq_\alpha$ is a congruence. In the literature, a relation as $\simeq_\alpha$ is referred to as α-congruence or α-conversion [13, 8].

We define a function $\mathrm{cv}\colon \mathcal{C}(\mathrm{BSP}^{\mathrm{dst}}) \to 2^{\mathcal{V}}$ to identify conflicting random variables that guide multiple stochastic delays in the same race. The function cv is defined using structural induction.

$$\mathrm{cv}(\epsilon) = \emptyset \qquad \mathrm{cv}(\delta) = \emptyset \qquad \mathrm{cv}(a.p) = \emptyset \qquad \mathrm{cv}(\sigma_X.p) = \emptyset$$
$$\mathrm{cv}(p+q) = \mathrm{cv}(p) \cup (\mathrm{rd}(p) \cap \mathrm{rd}(q)) \cup \mathrm{cv}(q).$$

If a term does not contain conflicting variables, we say that it is conflict-free. We characterize such terms using a predicate cf that checks whether the set of conflict variables in the current step is empty. Given a process term p, $\mathrm{cf}(p)$ is true if and only if $\mathrm{cv}(p) = \emptyset$.

We add an α-conversion rule to the structural operational semantics, viz.

$$(\alpha) \quad \frac{\langle q, \Delta\rangle \overset{S}{\mapsto}_{d_S} \langle p', \Delta'\rangle,\ \mathrm{cf}(q),\ \mathrm{cf}(p'),\ p \simeq_\alpha q}{\langle p, \Delta\rangle \overset{S}{\mapsto}_{d_S} \langle p', \mathrm{res}(\Delta', \mathrm{rd}(p') \setminus \mathrm{rd}(q))\rangle}\,.$$

This rule guarantees that the stochastic delay transitions are performed as a result of a race which does not lead to conflicting behavior. This is achieved by finding an α-converted term that is conflict-free and performing the race with it. Note that the non-racing terms of p' can get an age in the process of α-converting p to q, so we have to reset them in the resulting environment. For example, α-converting $\sigma_Y.\sigma_X.\epsilon + \sigma_X.\epsilon$ to $\sigma_Y.\sigma_U.\epsilon + \sigma_X.\epsilon$, where $\Delta(X) \neq \bot$ results in $\Delta(U) \neq \bot$, but U has not participated in any race. In order to exclude conflicting behavior, we use the predicate cf.

This means that we have to adapt the operational semantics by adding an extra conflict-freeness condition for every state that has the option to perform a stochastic delay. For example, the adapted version of Rule 11 is:

$$11\alpha \quad \frac{\langle p, \Delta\rangle \overset{S}{\mapsto}_{d_S^+} \langle p', \Delta'\rangle,\ \langle q, \Delta\rangle \not\mapsto,\ \mathrm{cf}(p+q)}{\langle p+q, \Delta\rangle \overset{S}{\mapsto}_{d_S} \langle p', \Delta'\rangle}.$$

The obtained theory is denoted as $\mathrm{BSP}_\alpha^{\mathrm{dst}}$. In the following we give an example of a STS in order to illustrate the operational semantics rules.

Example 9. In Fig. 3 we give the $STS(\sigma_X.\sigma_X.\epsilon + \sigma_X.a.\epsilon, (\Phi, \Delta_0))$, where initially $\Phi = \{X \mapsto F, Y \mapsto F, Z \mapsto F\}$ and $\Delta_0 = \{X \mapsto \bot, Y \mapsto \bot, Z \mapsto \bot\}$. Note that we give possible α-conversions in brackets for clarification, but it is not a part of the transition system.

Because of lack of space we do not present the equational theory of $\mathrm{BSP}_\alpha^{\mathrm{dst}}$, for which the reader is referred to [19]. Next, we investigate the behaviour of the parallel composition in the current setting.

6 Parallel Composition

We add an ACP-style parallel composition to the theory $\mathrm{BSP}_\alpha^{\mathrm{dst}}$ and obtain the theory of Basic Communication Processes with Discrete Stochastic Time and α-conversion $\mathrm{BCP}_\alpha^{\mathrm{dst}}(\mathcal{A}, \mathcal{V}, \gamma)$, where γ is the ACP-style communication function. As the parallel composition allows both interleaving and communication of immediate actions, in the present setting it should also cater for interleaving and synchronization of stochastic delays. Similarly to real-time PA's, we merge the delays in case the processes perform stochastic delays of different duration. We

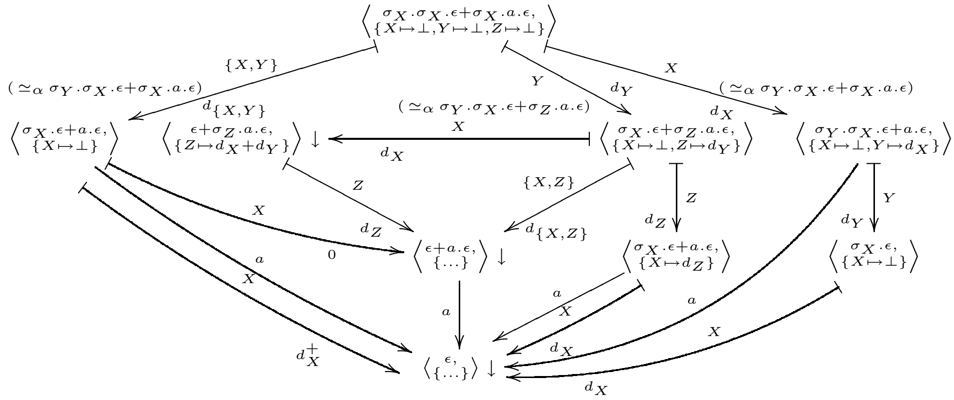

Fig. 3. Stochastic transition system of $\sigma_X.\sigma_X.\epsilon + \sigma_X.a.\epsilon$

synchronize the processes in case their delays are of the same duration. Immediate actions always take precedence over time in the parallel composition, except when performing zero duration delays. It is important to perform all possible zero delays and afterwards the immediate actions because otherwise we may lose communication options. For example, $\sigma_X.a.\epsilon \parallel b.\epsilon$ should allow a and b to communicate if $F_X(0) \neq 0$.

The definitions of rd, cv and $\simeq_\alpha$ are extended straightforwardly to apply to a parallel process $p \parallel q$. We give the operational semantics of the parallel composition in the following table:

$$17\ \frac{\langle p, \Delta\rangle\downarrow, \langle q, \Delta\rangle\downarrow}{\langle p \parallel q, \Delta\rangle\downarrow} \qquad 18\ \frac{\langle p, \Delta\rangle \overset{S}{\mapsto}_0 \langle p', \Delta'\rangle,\ \langle q, \Delta\rangle \not\mapsto}{\langle p \parallel q, \Delta\rangle \overset{S}{\mapsto}_0 \langle p' \parallel q, \Delta'\rangle}$$

$$20\ \frac{\langle p, \Delta\rangle \overset{a}{\to} \langle p', \Delta'\rangle,\ \langle q, \Delta\rangle \not\mapsto}{\langle p \parallel q, \Delta\rangle \overset{a}{\to} \langle p' \parallel q, \Delta'\rangle} \qquad 22\ \frac{\langle p, \Delta\rangle \overset{a}{\to} \langle p', \Delta'\rangle,\ \langle q, \Delta\rangle \overset{T}{\mapsto}_{d_T^+} \langle q', \Delta''\rangle}{\langle p \parallel q, \Delta\rangle \overset{a}{\to} \langle p' \parallel q, \mathrm{age}(\Delta', \mathrm{rd}(q), 0)\rangle}$$

$$24\ \frac{\langle p, \Delta\rangle \overset{a}{\to} \langle p', \Delta'\rangle,\ \langle q, \Delta\rangle \overset{b}{\to} \langle q', \Delta'\rangle,\ \gamma(a,b) = c}{\langle p \parallel q, \Delta\rangle \overset{c}{\to} \langle p' \parallel q', \Delta'\rangle}$$

$$25\ \frac{\langle p, \Delta\rangle \overset{S}{\mapsto}_{d_S} \langle p', \Delta'\rangle,\ \langle q, \Delta\rangle \overset{T}{\mapsto}_{d_T} \langle q', \Delta''\rangle,\ d_S < d_T}{\langle p \parallel q, \Delta\rangle \overset{S}{\mapsto}_{d_S} \langle p' \parallel q, \Delta'''\rangle}, \quad \text{where } \Delta''' = \mathrm{age}(\Delta', \mathrm{rd}(q), d_S)$$

$$27\ \frac{\langle p, \Delta\rangle \overset{S}{\mapsto}_{d_S} \langle p', \Delta'\rangle,\ \langle q, \Delta\rangle \overset{T}{\mapsto}_{d_T} \langle q', \Delta''\rangle,\ d_S = d_T}{\langle p \parallel q, \Delta\rangle \overset{S\cup T}{\mapsto}_{d_{S\cup T}} \langle p' \parallel q', \Delta'''\rangle}, \quad \text{where } \Delta''' = \mathrm{res}(\mathrm{age}(\Delta, \mathrm{rd}(p \parallel q), d_{S\cup T}), S \cup T).$$

We briefly discuss the new rules. Rule 17 states when the parallel composition has the termination option. Rule 18 enables zero delays before immediate actions similar to the alternative composition. Rules 20 and 22 enable interleaving of actions, by allowing the left operand to perform an immediate action if the right one cannot delay or it can delay with positive duration, in which case the zero durations are disabled by aging of 0 in Rule 22. Rule 24 states that synchronization of actions can occur, only if their communication is defined by the communication function γ. Rule 25 enables the race condition, similar to the Rule 14 for the alternative composition. Rule 27 enables simultaneous passage of time for the left and right operand which allows synchronization of stochastic delays that exhibit the same duration. (Rules 19, 21, 23 and 26 are omitted as analogous to the rules 18, 20, 22 and 25.)

It is easily observed that the parallel operator is both commutative and associative. The proof for the action transitions is standard. Regarding stochastic delays, the properties follow immediately from the structural operational semantics. Note that the race imposed by the parallel operator is the same as for the alternative composition. In the following example we illustrate some problems introduced by the weak choice and the α-conversion for the parallel operator, ultimately leading to absence of a standard expansion law.

Example 10 (No expansion law for $\mathrm{BCP}_\alpha^{\mathrm{dst}}$*).* Let $p \equiv \sigma_X.\epsilon$ and $q \equiv \sigma_Y.\epsilon$. We observe their parallel composition $p \parallel q$ and $p \mathbin{⫽} q + q \mathbin{⫽} p + p \mid q$ as its standard expansion. Note that $p \mathbin{⫽} q$ can perform a delay guided by X if $P(X < Y) > 0$. Same holds for $q \mathbin{⫽} p$, whereas $p \mid q$ performs a delay if $P(X = Y) > 0$. Suppose (Φ, Δ) is the environment. Then $\langle \sigma_X.\epsilon, \Delta\rangle \overset{X}{\mapsto}_{d_X} \langle \epsilon, \Delta'\rangle$ and $\langle \sigma_Y.\epsilon, \Delta\rangle \overset{Y}{\mapsto}_{d_Y} \langle \epsilon, \Delta'\rangle$. Let us assume that $d_X < d_Y$. Then one obtains the transition $\langle \sigma_X.\epsilon \parallel \sigma_Y.\epsilon, \Delta\rangle \overset{X}{\mapsto}_{d_X} \langle \epsilon \parallel \sigma_Y.\epsilon, \Delta''\rangle$, where $\Delta''(Y) = d_X$ and the transition system deadlocks.

Next, let us observe the process term obtained by the standard expansion law $\sigma_X.\epsilon \mathbin{⫽} \sigma_Y.\epsilon + \sigma_Y.\epsilon \mathbin{⫽} \sigma_X.\epsilon + \sigma_X.\epsilon \mid \sigma_Y.\epsilon$. This term has semantics only if it is first α-converted to $\sigma_X.\epsilon \mathbin{⫽} \sigma_Y.\epsilon + \sigma_{Y'}.\epsilon \mathbin{⫽} \sigma_{X'}.\epsilon + \sigma_{X''}.\epsilon \mid \sigma_{Y''}.\epsilon$, where $F_X = F_{X'} = F_{X''}$ and $F_Y = F_{Y'} = F_{Y''}$. Now, it is straightforward to observe that the parallel composition and its standard expansion do not have the same transition systems. For example, due to the weak choice the standard expansion term can do a stochastic delay guided by X, followed by a stochastic delay guided by Y' and aged by d_X and afterwards it finally deadlocks.

Based on the previous observations we conclude that the lack of total order on the durations of the stochastic delays and the presence of weak choice and α-conversion made it difficult to obtain a standard expansion law. However, because we retained the weak choice we are able to embed real-time in the STS's, which is presented in the following section.

7 Embedding Real Time in Stochastic Time

We consider the embedding of $\mathrm{BCP}^{\mathrm{srt}}$ into $\mathrm{BCP}_\alpha^{\mathrm{dst}}$. $\mathrm{BCP}^{\mathrm{srt}}(\mathcal{A}, \gamma)$ is a real-time extension of $\mathrm{BSP}(\mathcal{A})$ with parallel composition that allows synchronization of time delays with the same duration. It is a variant of the process algebra $\mathrm{TCP}_{\mathrm{srt}}(\mathcal{A}, \gamma)$ of [18] without sequential composition. Its semantics is given in terms of *timed transition systems* (TTS's).

Definition 11. *TTS is a structure* $TTS = (\mathcal{S}, \rightarrow, \mapsto, \downarrow)$ *where*

- $\mathcal{S}$ *is a set of states labeled by closed* $\mathrm{BCP}^{\mathrm{srt}}$*-terms;*
- $\rightarrow \subseteq \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ *is a labeled transition relation;*
- $\mapsto \subseteq \mathcal{S} \times \mathrm{I\!R}_0^+ \times \mathcal{S}$ *is a timed transition relation;*
- $\downarrow \subseteq \mathcal{S}$ *is an immediate termination predicate.*

Similarly to STSs, we use infix notation for $\rightarrow$ and $\mapsto$. By $\overset{t}{\mapsto}$ we denote that time $t \geq 0$ has passed. The TTS of a term p is denoted by $TTS(p)$. We denote the set of TTSs by $\mathcal{TTS}$.

The embedding of TTSs into STSs is given by an embedding of $\mathrm{BCP}^{\mathrm{srt}}$-terms in $\mathrm{BCP}_{\alpha}^{\mathrm{dst}}$-terms that will effectively replace each timed delay of duration d by a stochastic delay guided by a degenerated random variable X_{d}, such that $P(X_{\mathrm{d}} = d) = 1$. The restrictions to degenerated random variables are denoted by a subscript deg. The embedding is given by the mapping $\xi\colon \mathcal{TTS} \rightarrow \mathcal{STS}$:

$$\xi(TTS(p)) = STS(\varepsilon(p), (\Phi_{\mathrm{deg}}, \Delta_{\perp})),$$

where Φ_{deg} is restricted to degenerated distributions, $\Delta_{\perp}(X) = \perp$, for all $X \in \mathcal{V}_{\mathrm{deg}}$ and the mapping $\varepsilon\colon \mathcal{C}(\mathrm{BCP}^{\mathrm{srt}}) \rightarrow \mathcal{C}(\mathrm{BCP}_{\alpha}^{\mathrm{dst}})$ is given by:

$$\begin{array}{lll} \varepsilon(\epsilon) = \epsilon & \varepsilon(\delta) = \delta & \varepsilon(a.p) = a.\varepsilon(p) \\ \varepsilon(\sigma^{\,t}.p) = \sigma_{X_t}.\varepsilon(p) & \varepsilon(p+q) = \varepsilon(p) + \varepsilon(q) & \varepsilon(p \parallel q) = \varepsilon(p) \parallel \varepsilon(q). \end{array}$$

Note that because of the degenerated distributions the stochastic transition system only deals with the probabilities 0 and 1. Therefore, in that setting our bisimulation coincides with strong timed bisimulation of [18], were only the durations of delays are required to match. We observe that only one of the operational rules 12, 13 and 14 is applicable at the same time and the stochastic delay with the shortest duration wins. Moreover, we realize that in this setting there is no need for α-conversion, since all stochastic delays guided by the same random variable either win the race together or age the same duration of time together. The behavior of the zero delay is captured by the rules 8 and 10 and the weak choice by the rules 9 and 11. The time interpolation of the real-time PA's is embedded by aging the racing delays by the interpolation time.

Taking all together we have the following theorem.

Theorem 12. *The mapping* $\xi\colon \mathcal{TTS} \rightarrow \mathcal{STS}$ *is an embedding.*

The proof of the theorem can be found in [19]. Next we give an example to illustrate the embedding.

Example 13. In Fig. 4, we have for the term $p \equiv \sigma^{\,t+s}.a.\epsilon \parallel \sigma^{\,t}.(\sigma^{\,s}.b.\epsilon + a.\epsilon)$, for $s, t > 0$ and $\gamma(a, b) = c$ the original $TTS(p)$ on the left, and its embedding, the $STS(\varepsilon(p), (\Phi_{\mathrm{deg}}, \Delta_{\perp}))$ on the right, where $\varepsilon(p) = \sigma_{X_{t+s}}.a.\epsilon \parallel \sigma_{X_t}.(\sigma_{X_s}.b.\epsilon + a.\epsilon)$. We represent only the important part of the environment.

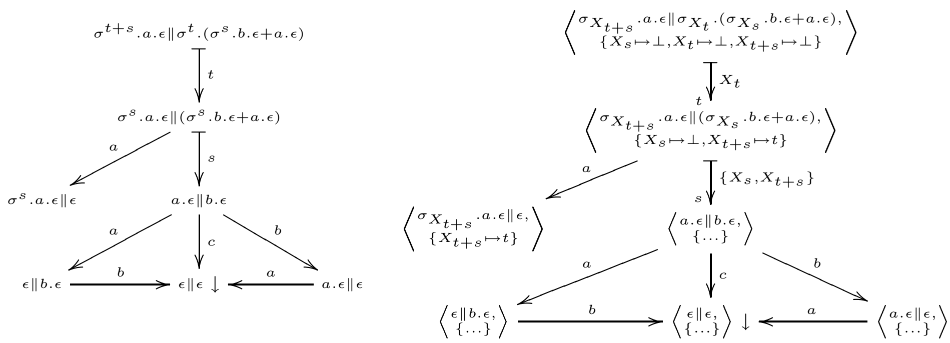

Fig. 4. Example embedding

8 Conclusion and Future Work

We have proposed a stochastic process algebra with immediate actions, termination and deadlock, and discrete distributions as an extension of un-timed process algebra. We introduced a notion of a stochastic transition system and gave a definition of strong bisimulation in that setting that conforms to the probabilistic bisimulation when considering the time as a constant and it corresponds to strong timed bisimulation when only considering probabilities of 0 and 1. We have argued that the bisimulation is a congruence. We showed conflicting behavior of the STS's and introduced α-conversion in order to deal with stochastic delays that are guided by conflicting variables.

We considered extending the algebra with parallel composition. However, expansion of the parallel operator using the alternative composition with weak choice turned out to be problematic. We identified the lack of total ordering on the durations observed by the stochastic delays as the main reason for failure of the standard expansion law when considering alternative composition with weak choice and α-conversion. However, because we retained the weak choice, we were able to propose an intuitive embedding of TTS into stochastic ones by restricting to discrete degenerated stochastic delays.

As future work we schedule an alternative way to obtain an expansion law for the parallel composition, as part of the identification of an axiomatic theory that conservatively extends the underlying real-time theory. Because of the semantical basis, we do not expect major difficulties when incorporating recursion. Also, we plan to extend the current setting with continuous stochastic time. Afterwards, we will consider case studies, especially in protocol verification (e.g. sliding window protocols), since successful modeling of real-time delays paves the way for an easy specification of time-outs.

Acknowledgment We are grateful to Jos Baeten for his support, reviews and comments and for the many fruitful discussions on the topic.

References

1. Baeten, J., Middelburg, C.: Process Algebra with Timing. Monographs in Theoretical Computer Science. Springer (2002)
2. Nicollin, X., Sifakis, J.: An overview and synthesis of timed process algebras. In de Bakker, J., et al, eds.: Real-Time: Theory in Practice. Volume 600 of LNCS. Springer (1992) 526–548
3. Jonsson, B., Larsen, K.G.: Probabilistic extensions of process algebras. In Bergstra, J.A., Ponse, A., Smolka, S.A., eds.: Handbook of Process Algebras. Elsevier (2001)
4. Andova, S.: Time and probability in process algebra. In: AMAST '00, Springer (2000) 323–338
5. Bernardo, M., Gorrieri, R.: A tutorial on EMPA: A theory of concurrent processes with nondeterminism, priorities, probabilities and time. Theoretical Computer Science **202**(1–2) (1998) 1–54
6. Hillston, J.: A compositional approach to performance modelling. Cambridge University Press (1996)
7. Hermanns, H.: Interactive Markov Chains And the Quest For Quantified Quantity. Volume 2428 of LNCS. Springer (2002)
8. D'Argenio, P., Katoen, J.P.: A theory of stochastic systems, part II: Process algebra. Information and Computation **203**(1) (2005) 39–74
9. Bravetti, M.: Specification and Analysis of Stochastic Real-time Systems. PhD thesis, Universita di Bologna (2002)
10. Lopez, N., Nunez, M.: NMSPA: A non-markovian model for stochastic processes. In: DSVV'2000, IEEE Computer Society Press (2000)
11. D'Argenio, P., Katoen, J.P.: A theory of stochastic systems, part I: Stochastic automata. Information and Computation **203**(1) (2005) 1–38
12. Marsan, M.A., Bianco, A., Ciminiera, L., Sisto, R., Valenzano, A.: A LOTOS extension for the performance analysis of distributed systems. IEEE/ACM Trans. Netw. **2**(2) (1994) 151–165
13. Priami, C.: Stochastic π-calculus with general distributions. In Ribaudo, M., ed.: 4th Workshop on PAPM, Torino, Italy (1996) 41–57
14. Hermanns, H., Mertsiotakis, V., Rettelbach, M.: Performance analysis of distributed systems using TIPP. In Pooley, Hillston, King, eds.: UKPEW'94, University of Edinburgh (1994) 131–144
15. Katoen, J.P., D'Argenio, P.R.: General distributions in process algebra. In Brinksma, H., et al., eds.: Lectures on formal methods and performance analysis. Volume 2090 of LNCS. Springer (2001) 375–429
16. Bravetti, M., D'Argenio, P.: Tutte le algebre insieme. In Baier, C., et al, eds.: Validation of Stochastic Systems. Volume 2925 of LNCS. Springer (2004) 44–88
17. D'Argenio, P.: From stochastic automata to timed automata: Abstracting probability in a compositional manner. In Fiore, M., Fridlender, D., eds.: Proceedings of WAIT 2003. Associated to the 32 JAIIO., Buenos Aires, Argentina (2003)
18. Baeten, J.C.M., Reniers, M.A.: Timed process algebra (with a focus on explicit termination and relative-timing). In Bernardo, M., Corradini, F., eds.: Formal Methods for the Design of Real-Time Systems. Volume 3185 of LNCS. Springer (2004) 59–97
19. Markovski, J., de Vink, E.P.: Embedding real time in stochastic time process algebras. Technical Report CS 06/15, Technische Universiteit Eindhoven (2006)
20. Larsen, K.G., Skou, A.: Bisimulation through probabilistic testing. Information and Computation **94**(1) (1991) 1–28