A Stochastic Causality-Based Process Algebra

Ed Brinksma, Joost-Pieter Katoen, Rom Langerak and Diego Latella[1]

Department of Computing Science, University of Twente
PO Box 217, 7500 AE Enschede, The Netherlands
[1]*CNUCE Istituto del CNR, Via Santa Maria 36, 56100 Pisa, Italy*
Email: {brinksma, katoen, langerak}@cs.utwente.nl; d.latella@cnuce.cnr.it

This paper discusses stochastic extensions of a simple process algebra in a causality-based setting. Atomic actions are supposed to happen after a delay that is determined by a stochastic variable with a certain distribution. A simple stochastic type of event structures is discussed, restricting the distribution functions to be exponential. A corresponding operational semantics of this model is given and compared to existing (interleaved) approaches. Secondly, a stochastic variant of event structures is discussed where distributions are of a much more general nature, viz. of phase-type. This includes exponential, Erlang, Coxian and mixtures of exponential distributions.

1. INTRODUCTION

Though originally formal methods concentrated on the specification, design, and analysis of functional aspects of distributed systems, recently the study of quantitative aspects of such systems has come into focus. Several extensions of formal methods where the occurrence of actions can be assigned a (fixed) probability and/or the time of occurrence can be constrained are known from the literature (Nicollin & Sifakis, 1992; van Glabbeek *et al.*, 1990).

An important reason for enhancing formal methods with notions such as probabilities and time is to facilitate the analysis of performance characteristics of systems designs. In this way the efficiency of design alternatives can be assessed such that in early design stages designs can be rejected because of unsatisfactory performance characteristics, thus avoiding costly redesign at later stages. In addition, a formal specification incorporating quantitative aspects can be very useful for establishing a well-understood and effective way of developing performance models, such as Markov chains and queueing networks, from system specifications.

A lot of effort has been put into the extension of formal methods where the time of actions is specified in a completely *deterministic* way. In early stages of the design there is often no exact timing information available and in, for instance, multi-media systems phenomena like jitter and response times are not deterministically determined but much more of a stochastic nature. In these cases deterministic timed extensions of process algebras are not always appropriate. Therefore, in the field of *stochastic process algebras* the time of occurrence of actions is determined by stochastic (or random) variables, in particular exponentially distributed ones. Several approaches to enhancing process algebras with exponentially distributed delays have been developed (Bernardo *et al.*, 1994; Buchholz, 1994; Götz *et al.*, 1993a; Hillston, 1994a).

Current stochastic process algebras all use labelled transition systems as an underlying semantical model. This results in a semantics based on the *interleaving* of independent actions. Though the structure of transition systems closely resembles that of standard Markov chain representations, the *state explosion* problem is a serious drawback. *Truly concurrent* semantical models are less affected by this problem as parallelism leads to the sum of the components states, rather than to their product (as in interleaving). In addition, true concurrency models retain explicit information about the parallelism between system components. As performance models typically are based on abstractions of the control and/or data flow structure of systems, the use of true concurrency semantics is thought to be a direct way of narrowing the gap from functional models. Finally, true concurrent models enable the possibility of *local analysis*. This means that it is relatively easy to study only that part of a system in which one is interested, isolating it from the rest.

(Labelled) event structures (Winskel, 1989) constitute a major branch of true concurrency models and are well-suited as a semantical model for process algebras like CCS. The basic ingredients of event structures are *events* modelling occurrences of actions, and a *causality* relation indicating the causal dependencies between events. To fit the specific requirements of parallel composition with multi-way synchronisation—as used in process algebras like LOTOS and CSP—an adaptation of labelled event structures, *bundle event structures*, are appropriate (Langerak, 1992). Using bun-

dle event structures a compositional true concurrent semantics for LOTOS can be defined that is 'compatible' with the standard interleaving semantics.

In (Katoen *et al.*, 1995; Brinksma *et al.*, 1994) we extensively treated an extension of this type of event structures with deterministic times in which time is associated to causal relations (termed bundles in our event structure model) and to events. Bundle delays specify the relative delay between causally dependent events while event delays enable the specification of timing constraints on events that have no incoming bundle. In this timed model components may synchronise on a common action as soon as all participants are ready to engage, that is, when all individual timing constraints are met. The work presented in this paper is based on the generalisation of deterministic times in our timed true concurrent model towards *distribution functions.*

We start by investigating a generalisation in which we restrict to exponential distributions. This results in a simple model where rates are associated with events only. The principle that a synchronisation takes place as soon as all participants are ready for it means in a stochastic setting that the delay of such an action will be distributed as the product of the individual distributions (or, equivalently, as the maximum of the corresponding individual stochastic variables). As the class of exponential distributions is *not* closed under product, we abandon our synchronisation principle and take a pragmatic approach by computing the rate of a synchronisation simply as some arbitrary function of the individual rates—similar to existing stochastic process algebras. The resulting model is used to provide a true concurrent semantics of a stochastic process algebra. A corresponding interleaving semantics is provided which shows that our simple stochastic model closely resembles existing interleaved proposals of stochastic process algebras, thus providing evidence of the adequacy of our approach.

Only a few stochastic process algebras are known (to us) that allow for a more general class of distribution functions (Ajmone Marsan *et al.*, 1994; Götz *et al.*, 1993b). The elegant—memoryless—properties of exponential distributions enable a smooth incorporation of such distributions into transition systems, while the interleaving of independent actions seems to complicate the use of more general (non-memoryless) distributions considerably (Götz *et al.*, 1993b). When carefully investigating the replacement of deterministic times in our model by general distributions it turns out that it is possible to support a class of distributions which is closed under product (corresponding to the maximum of stochastic variables under the assumption of statistical independence), and which contains an identity element for product. These properties will be justified in this paper. As an interesting class of distribution functions that satisfies these constraints we propose the use of *phase-type* (*PH*-) distributions. PH-distributions can be considered as matrix generalisations of exponential distributions and are well-suited for numerical computation. They are used in many probabilistic models that have matrix-geometric solutions, have a rich theory (Neuts, 1981; Neuts, 1989), and include frequently used distributions such as hyper- and hypo-exponential, Erlang, and Cox distributions.

This paper is organised as follows. Section 2 briefly introduces bundle event structures and Section 3 shows how this model can be used to provide a causality-based semantics to a simple process algebra. A treatment of a deterministic time extension of this model is given in Section 4. The timed model is a simplified version of models elaborated in (Katoen *et al.*, 1995; Brinksma *et al.*, 1994). Section 5 reports on the study of exponential distributions in our model and relates this work to existing interleaved proposals. Section 6 investigates the use of more general distribution functions in bundle event structures. Finally, Section 7 contains conclusions and pointers for future work. Appendix A introduces PH-distributions and provides some important theoretic results that are relevant in the context of this paper.

2. BUNDLE EVENT STRUCTURES

Bundle event structures consist of *events* labelled with actions (an event modelling the occurrence of its action), together with relations of causality and conflict between events. System runs can be modelled as partial orders of events satisfying certain constraints posed by the causality and conflict relations between the events.

Conflict is a symmetric binary relation between events and the intended meaning is that when two events are in conflict, they can never both happen in a single system run. *Causality* is represented by a relation between a set of events X, that are pairwise in conflict, and an event e. The interpretation is that if e happens in a system run, exactly one event in X has happened before (and caused e). This enables us to uniquely define a causal ordering between the events in a system run. When there is neither a conflict nor a causal relation between events they are independent. Once enabled, independent events can occur in any order or in parallel.

Definition 2.1 A *bundle event structure* $\mathcal{E}$ is a quadruple $(E, \#, \mapsto, l)$ with E, a set of *events*, $\# \subseteq E \times E$, the (irreflexive and symmetric) *conflict* relation, $\mapsto \subseteq 2^E \times E$, the *causality* relation, and $l : E \to L$, the *action-labelling* function, where L is a set of action labels, such that $\mathcal{E}$ satisfies $\forall X \subseteq E, e \in E$:

$$X \mapsto e \;\Rightarrow\; (\forall e_i, e_j \in X : e_i \neq e_j \;\Rightarrow\; e_i \# e_j) \quad .$$

□

The constraint specifies that for bundle $X \mapsto e$ all events in X are in mutual conflict. Bundle event structures are graphically represented in the following way.

Events are denoted as dots; near the dot the action label is given. Conflicts are indicated by dotted lines between representations of events. A bundle (X, e) is indicated by drawing an arrow from each event in X to e and connecting all arrows by small lines. We often denote an event labelled a by e_a.

In the sequel we adopt the following notations. For sequences $\sigma = x_1 \ldots x_n$, let $\overline{\sigma}$ denote the set of elements in σ, that is, $\overline{\sigma} = \{x_1, \ldots, x_n\}$, and let σ_i denote the prefix of σ up to the $(i-1)$-th element, that is, $\sigma_i = x_1 \ldots x_{i-1}$, for $0 < i \leqslant n+1$.

Definition 2.2 For σ a sequence of events $e_1 \ldots e_n$ we define $\mathsf{cfl}(\sigma) = \{e \in E \mid \exists e_i \in \overline{\sigma} : e_i \# e\}$ and $\mathsf{sat}(\sigma) = \{e \in E \mid \forall X \subseteq E : X \mapsto e \Rightarrow X \cap \overline{\sigma} \neq \varnothing\}$. □

$\mathsf{cfl}(\sigma)$ is the set of events that are in conflict with some event in σ. $\mathsf{sat}(\sigma)$ is the set of events that have a causal predecessor in σ for all bundles pointing to them. That is, for events in $\mathsf{sat}(\sigma)$ all bundles are 'satisfied'.

The concept of a sequential observation of a system's behaviour is defined as follows. Event traces consist of distinct events (i.e. $e_i \notin \overline{\sigma_i}$, for all i) and are conflict-free ($e_i \notin \mathsf{cfl}(\sigma_i)$), for obvious reasons. In addition, each event in the event trace is preceded in the sequence by a causal predecessor for each bundle pointing to it (that is, $e_i \in \mathsf{sat}(\sigma_i)$). That is,

Definition 2.3 An *event trace* σ of $\mathcal{E}$ is a sequence of events $e_1 \ldots e_n$ with $e_i \in E$ $(0 < i \leqslant n)$, satisfying $e_i \in \mathsf{sat}(\sigma_i) \setminus (\mathsf{cfl}(\sigma_i) \cup \overline{\sigma_i})$, for all i. □

Event traces correspond to linearisations of system runs.

Example 2.4 Some bundle event structures are depicted in Figure 1. Event structure (c) has bundles $\{e_a, e_b\} \mapsto e_c$, $\{e_b\} \mapsto e_d$, and $\{e_e\} \mapsto e_d$, and a conflict between e_a and e_b. Thus, event e_d is enabled once both e_e and e_b have happened, and e_c once either e_a or e_b has occurred before. Example event traces of this structure are $e_a e_e e_c$, $e_b e_c$, and $e_e e_b e_d e_c$. □

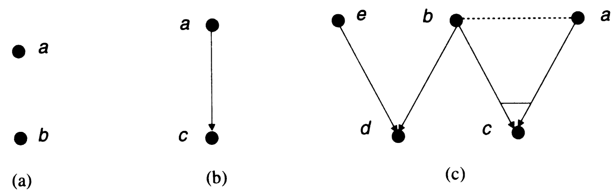


FIGURE 1. Some example bundle event structures.

3. A PROCESS ALGEBRA

In this section we introduce a (simple) process algebra and show how bundle event structures can be used to provide a causality-based semantics to a process algebra in a compositional way. Let L be a set of action labels, $a \in L \cup \{\tau\}$, where τ is a special label representing silent actions, $G \subseteq L$ (G finite), and $H : L \cup \{\tau\} \to L \cup \{\tau\}$ a relabelling function with $H(\tau) = \tau$ and $H(a) \neq \tau$ for $a \in L$. We consider the process algebra

$$B ::= \mathbf{0} \mid a ; B \mid B + B \mid B \,||_G\, B \mid B[H] \mid B \setminus G.$$

The precedences of the operators are, in increasing binding order: $||_G$, $+$, $;$, $[\,]$ and $\setminus$.

Actions are considered to be atomic and to occur instantaneously. $\mathbf{0}$ represents the behaviour that can perform no actions at all. $a ; B$ denotes a behaviour which may engage in a and after the occurrence of a behaves like B. $B_1 + B_2$ denotes the (standard) choice between behaviours B_1 and B_2, while $B_1 \,||_G\, B_2$ is their parallel composition where synchronisation is required for actions in G. $B[H]$ denotes a behaviour which is obtained by renaming the actions in B according to H. $B \setminus G$ behaves like B except that all actions in G are turned into invisible actions (i.e. τ).

A causality-based semantics can now be defined as follows. Let $\mathcal{E}[\![B_i]\!] = \mathcal{E}_i = (E_i, \#_i, \mapsto_i, l_i)$, i=1, 2 with $E_1 \cap E_2 = \varnothing$. For action-prefix $a ; B_1$, a new event e_a (labelled a) is introduced which causally precedes all initial events of $\mathcal{E}_1$ (cf. Figure 2). $\mathcal{E}[\![B_1 + B_2]\!]$ is

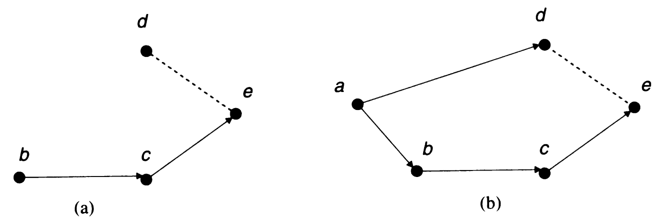


FIGURE 2. (a) $\mathcal{E}[\![B_1]\!]$ and (b) $\mathcal{E}[\![a ; B_1]\!]$.

equal to the union of $\mathcal{E}_1$ and $\mathcal{E}_2$ extended with mutual conflicts between all initial events of $\mathcal{E}_1$ and $\mathcal{E}_2$ such that only B_1 or B_2 can happen. $\mathcal{E}[\![B_1 \setminus G]\!]$ is identical to $\mathcal{E}[\![B_1]\!]$ except that events labelled with a label in G are now labelled with τ turning those events into internal ones. $\mathcal{E}[\![B_1[H]]\!]$ is defined similarly where events are relabelled according to H.

For parallel composition the events of $\mathcal{E}[\![B_1 \,||_G\, B_2]\!]$ are constructed as follows: an event e of $\mathcal{E}_1$ or $\mathcal{E}_2$ that does not need to synchronise is paired with the auxiliary symbol $*$, and an event which is labelled with an action in G is paired with all events (if any) in the other process that are equally labelled. Thus events are *pairs* of events of $\mathcal{E}_1$ and $\mathcal{E}_2$, or with one component equal to $*$. Two events are now put in conflict if any of their corresponding components are in conflict, or if different events have a common component different from $*$ (such events appear if two or more events in one process synchronise with the same event in the other process). A bundle is introduced such that if we take the projection on the i-th component (i=1, 2) of all events in the bundle we obtain a bundle in $\mathcal{E}_i$ (cf. Figure 3).

We suppose there is an infinite universe E_U of events. For $G \subseteq L$, set $E_i^s = \{e \in E_i \mid l_i(e) \in G\}$ is the set of synchronised events and $E_i^f = E_i \setminus E_i^s$ the set of unsynchronised events. The set of initial events of $\mathcal{E}$ are events that have no bundle pointing to them, i.e. $init(\mathcal{E}) = \{ e \in E \mid \neg(\exists X \subseteq E : X \mapsto e) \}$.

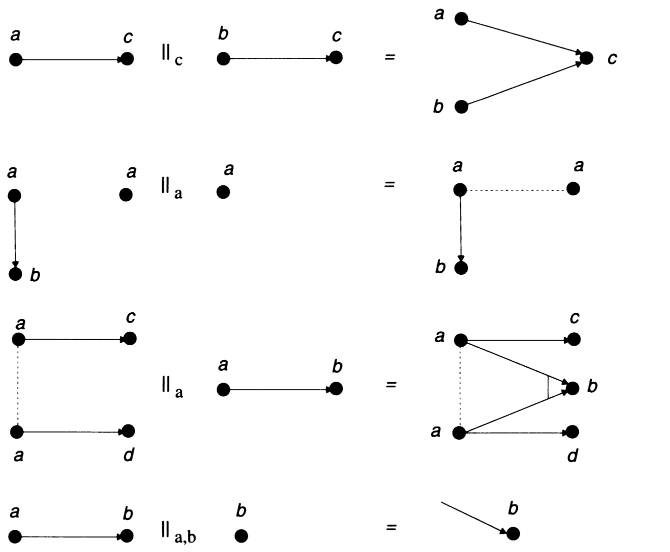


FIGURE 3. Examples of parallel composition.

Definition 3.1 $\mathcal{E}[\![\ ]\!]$ is defined as follows:

$$
\begin{array}{rcl}
\mathcal{E}[\![\mathbf{0}]\!] & = & (\varnothing, \varnothing, \varnothing, \varnothing) \\
\mathcal{E}[\![a\,; B_1]\!] & = & (E, \#_1, \mapsto, l) \text{ where} \\
E & = & E_1 \cup \{ e_a \} \text{ for } e_a \in E_U \setminus E_1 \\
\mapsto & = & \mapsto_1 \cup (\{ \{ e_a \} \} \times init(\mathcal{E}_1)) \\
l & = & l_1 \cup \{ (e_a, a) \} \\
\mathcal{E}[\![B_1 + B_2]\!] & = & (E_1 \cup E_2, \#, \mapsto, l_1 \cup l_2) \text{ where} \\
\# & = & \#_1 \cup \#_2 \cup (init(\mathcal{E}_1) \times init(\mathcal{E}_2)) \\
\mapsto & = & \mapsto_1 \cup \mapsto_2 \\
\mathcal{E}[\![B_1 \setminus G]\!] & = & (E_1, \#_1, \mapsto_1, l) \text{ where} \\
 & & (l_1(e) \in G \;\Rightarrow\; l(e) = \tau) \\
 & & \wedge\ (l_1(e) \notin G \;\Rightarrow\; l(e) = l_1(e)) \\
\mathcal{E}[\![B_1[H]]\!] & = & (E_1, \#_1, \mapsto_1, H \circ l_1) \\
\mathcal{E}[\![B_1 \,\|_G\, B_2]\!] & = & (E, \#, \mapsto, l) \text{ where} \\
E & = & (E_1^f \times \{ * \}) \cup (\{ * \} \times E_2^f) \cup \\
 & & \{ (e_1, e_2) \in E_1^s \times E_2^s \mid l_1(e_1) = l_2(e_2) \} \\
(e_1, e_2) \# (e_1', e_2') & \Leftrightarrow & (e_1 \#_1 e_1') \vee (e_2 \#_2 e_2') \vee \\
 & & (e_1 = e_1' \neq * \;\wedge\; e_2 \neq e_2') \vee \\
 & & (e_2 = e_2' \neq * \;\wedge\; e_1 \neq e_1') \\
X \mapsto (e_1, e_2) & \Leftrightarrow & \exists X_1 : (X_1 \mapsto_1 e_1 \;\wedge \\
 & & X = \{ (e_i, e_j) \in E \mid e_i \in X_1 \}) \vee \\
 & & \exists X_2 : (X_2 \mapsto_2 e_2 \;\wedge \\
 & & X = \{ (e_i, e_j) \in E \mid e_j \in X_2 \}) \\
l((e_1, e_2)) & = & \textsf{if } e_1 = * \textsf{ then } l_2(e_2) \textsf{ else } l_1(e_1)
\end{array}
$$

□

Here, $\circ$ denotes function composition.

4. TIMED EVENT STRUCTURES

Time is added to bundle event structures in two ways. To specify the relative delay between causally dependent events time is associated to bundles, and in order to facilitate the specification of timing constraints on events that have no bundle pointing to them (i.e. the initial events), time is also associated to events. Though it seems sufficient to only have time labels for initial events, synchronisation of events makes it necessary to allow for equipping all events with time labels, including the non-initial ones. Alternatively, we could explicitly model the start of the system by some fictitious event, ω say. Then the time associated to event e can be considered as the time associated to the bundle pointing from the fictitious event to e. We do not consider the introduction of such an event ω as the definitions become more complex—ω has to be treated differently from 'normal' events—and proof obligations become more severe—e.g. one has to prove that bundles $X \mapsto e$ satisfy $X = \{ \omega \}$, or $\omega \notin X$ and $e \neq \omega$.

We assume mappings $\mathcal{T}$ and $\mathcal{D}$ to associate a value of T, the time domain, to bundles and events, respectively. A bundle (X, e) with $\mathcal{T}((X, e)) = t$ is denoted by $X \overset{t}{\mapsto} e$; its interpretation is that if an event in X has happened at a certain time, then e is enabled t time units later. $\mathcal{D}$ associates time to events; the interpretation is that e with $\mathcal{D}(e) = t$ can happen after t time-units from the beginning of the system.

Definition 4.1 A *timed* event structure is a triple $\langle \mathcal{E}, \mathcal{T}, \mathcal{D} \rangle$ with $\mathcal{E}$ a bundle event structure $(E, \#, \mapsto, l)$, $\mathcal{T} : \mapsto \to T$, the *timing* function, and $\mathcal{D} : E \to T$, the *delay* function. □

For depicting timed event structures we use the following conventions. The time associated with a bundle and event is a non-negative real and is depicted near to a bundle and event, respectively. For convenience, zero delays are omitted.

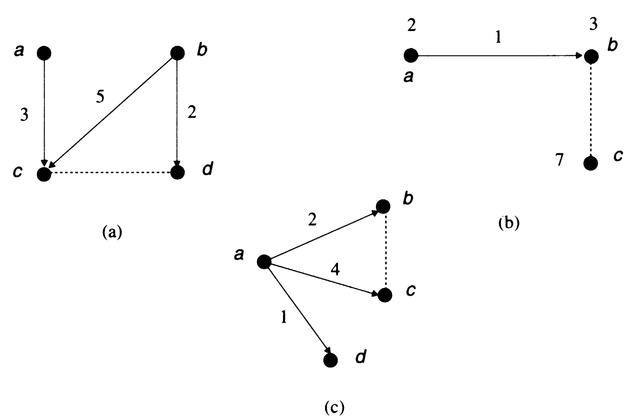


FIGURE 4. Some example timed event structures.

Example 4.2 Some example timed event structures are depicted in Figure 4. Figure 4(a) has bundles $\{e_a\} \overset{3}{\mapsto} e_c$, $\{e_b\} \overset{5}{\mapsto} e_c$, $\{e_b\} \overset{2}{\mapsto} e_d$, and a conflict between e_c and e_d. For Figure 4(b) we have $\mathcal{D}(e_a) = 2$, $\mathcal{D}(e_b) = 3$ and $\mathcal{D}(e_c) = 7$. Note that e_b is a non-initial event having a delay associated with it. □

As a generalisation of the notion of event trace we define the notion of timed event trace. As a shorthand notation for sequences of timed events $\sigma = (e_1, t_1) \dots (e_n, t_n)$ let $\lfloor\sigma\rfloor$ denote the set of events in σ, i.e. $\lfloor\sigma\rfloor = \{e_1, \dots, e_n\}$. A timed sequential observation of the system is now defined as an untimed sequential observation where each event has a correct timing associated with it.

Definition 4.3 A *timed event trace* of $\langle\mathcal{E}, \mathcal{T}, \mathcal{D}\rangle$ is a sequence σ of timed events $(e_1, t_1) \dots (e_n, t_n)$ with $e_i \in E$, $t_i \in T$ $(0 < i \leqslant n)$, satisfying

1. $e_1 \dots e_n$ is an event trace of $\mathcal{E}$
2. $\forall e_i : t_i \geqslant max(\mathcal{D}(e_i), h_i)$ where h_i equals
$Max\{t{+}t_j \mid \exists X \subseteq E : X \overset{t}{\mapsto} e \ \wedge \ X \cap \lfloor\sigma_i\rfloor = \{e_j\}\}$.
□

Max of the empty set is 0. Note that—according to the last constraint—events can happen at any time from the moment they are enabled. The motivation for this non-urgency is that in general an event may be subject to interaction (e.g. with the environment) which may introduce further delays. e is enabled if at least its delay, $\mathcal{D}(e)$, has elapsed and the time relative to all its causal predecessors.

Some timed event traces of the timed event structure of Figure 4(a) are:

$$(e_a, t_a)(e_b, t_b)(e_d, t_d) \quad \text{with} \quad t_d \geqslant t_b{+}2, \text{ and}$$
$$(e_a, t_a)(e_b, t_b)(e_c, t_c) \quad \text{with} \quad t_c \geqslant max(t_a{+}3, t_b{+}5) \quad .$$

Notice that event traces do respect causality, but not necessarily time. That is, two (or more) independent events can occur in the trace in either order regardless of their timing. For example, $(e_b, 1)(e_a, 3)$ and $(e_a, 3)(e_b, 1)$ are event traces of the structure in Figure 4(a). The possible choices correspond to the possible interleavings of the (independent) events.

Theorem 4.4 $\sigma\ (e_i, t_i)(e_{i+1}, t_{i+1})\ \sigma'$ with $t_{i+1} < t_i$ is a timed event trace of $\langle\mathcal{E}, \mathcal{T}, \mathcal{D}\rangle$ iff $\sigma\ (e_{i+1}, t_{i+1})(e_i, t_i)\ \sigma'$ is a timed event trace of $\langle\mathcal{E}, \mathcal{T}, \mathcal{D}\rangle$.

Proof: '$\Rightarrow$' : assume $\sigma\ (e_i, t_i)(e_{i+1}, t_{i+1})\ \sigma'$ is a timed event trace and let $t_{i+1} < t_i$. Then prove that $\sigma\ (e_{i+1}, t_{i+1})(e_i, t_i)\ \sigma'$ is also a trace by checking the conditions of being a timed event trace (by contradiction). The same procedure applies for '$\Leftarrow$'. □

For a more extensive discussion on 'ill-timed' but 'well-caused' timed traces we refer to (Aceto & Murphy, 1993).

5. SIMPLE STOCHASTIC EVENT STRUCTURES

5.1. Exponential Distributions

Exponential distributions are defined as follows.

Definition 5.1 A probability distribution function F, defined by $F(x) = 1 - e^{-\lambda x}$ for $x \geqslant 0$ and $F(x) = 0$, for $x < 0$, is an *exponential distribution* with *rate* λ $(\lambda \in \mathbb{R}^+)$. □

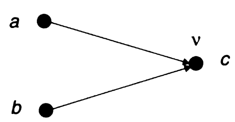
Evidently, a rate uniquely characterises an exponential distribution. A well-known property of exponential distributions is the memoryless property.

Lemma 5.2 For U an exponentially distributed stochastic variable, $x, y \geqslant 0$ we have $Pr\{U \leqslant x + y \mid U > y\} = Pr\{U \leqslant x\}$. This property is known as the *memoryless* (or Markovian) property. □

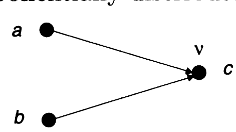
Informally, it states that the probability of U being at most $x{+}y$ given that it is larger than y is independent of y and equal to the probability of U being at most x. So, the fact that $U > y$ is completely irrelevant.

5.2. The Model

In this section we develop a simple stochastic variant of bundle event structures by associating rates to events. The motivation for only associating rates to events, and not to bundles too, is that when choosing to remain in the domain of exponential distributions it turns out to be sufficient to attach rates to events only. Consider, for example, the following event structure in which rates are associated to bundles:



The interpretation is that a rate associated to bundle X pointing to e determines the time of e's enabling relative to the time of occurrence of its causal predecessor in X. The above structure specifies that the time period between the enabling of e_c and the occurrence of e_a (e_b) is exponentially distributed with rate λ (μ). Given that we want to stay in the domain of exponential distributions this is equivalent to saying that the time between the last occurrence of an event preceding e_c and the enabling of e_c is exponentially distributed with rate ν where ν is determined by λ and μ. Due to the memoryless property this is statistically equivalent to saying that the period between the start of the system and the enabling of e_c is exponentially distributed with rate ν:



Therefore we choose to associate rates to events only. In this way we also keep close to the stochastic transition systems that underly stochastic process algebra based on interleaving (see also Section 5.4). Thus,

Definition 5.3 A *simple stochastic* bundle event structure is a tuple $\langle \mathcal{E}, \mathcal{R} \rangle$ with $\mathcal{E}$ a bundle event structure $(E, \#, \mapsto, l)$ and $\mathcal{R} : E \to \mathbb{R}^+$, the *rate* function. □

As a generalisation of the notion of event trace we define the notion of stochastic event trace.

Definition 5.4 A *stochastic event trace* of $\langle \mathcal{E}, \mathcal{R} \rangle$ is a sequence σ of rated events $(e_1, \lambda_1) \ldots (e_n, \lambda_n)$ with $e_i \in E$, $\lambda_i \in \mathbb{R}^+$ $(0 < i \leqslant n)$ iff $e_1 \ldots e_n$ is an event trace of $\mathcal{E}$ and $\lambda_i = \mathcal{R}(e_i)$ for all i. □

The set of stochastic event traces of stochastic bundle event structure $\mathcal{E}$ is denoted $STr(\mathcal{E})$.

5.3. A Simple Stochastic Process Algebra

Let the syntax of the language SL of simple finite stochastic behaviours be defined as follows:

$$B ::= \mathbf{0} \mid (\lambda)\ a\,; B \mid B + B \mid B \,||_G\, B \mid B[H] \mid B \setminus G.$$

Actions are considered to be atomic and to occur instantaneously. $(\lambda)\ a\,; B$ denotes a behaviour which may engage in a from a time period relative to the beginning of the system with an exponential distributed length (of rate λ) and after the occurrence of a behaves like B. λ specifies the rate of the exponential distribution of a *relative* delay of an action.

In the deterministic timing case a set of behaviours may synchronise on a common action as soon as all participants are ready to engage in this action. For example, in an expression like $(t)\ a \,||_a\, (t')\ a$ the resulting action a is enabled from $max(t, t')$. In the case where the delay of actions (in fact, events) is determined by a stochastic variable, it seems natural—and a straightforward generalisation of the deterministic time case—to let the enabling time of a synchronisation be determined by the maximum of the stochastic variables that determine the local delay of this action. From basic probability theory (Kobayashi, 1978) we know that the distribution of the maximum of two (or more) independent stochastic variables corresponds to the product of their distribution functions. That is,

Theorem 5.5 Let $U_1, \ldots, U_n$ $(n \geqslant 1)$ be independent stochastic variables where U_i has distribution F_{U_i}, and $W = Max\{ U_1, \ldots, U_n \}$. Then the probability distribution function of W equals

$$F_W(x) = \prod_{i=1}^{n} F_{U_i}(x) \quad ,$$

and its probability density function

$$F'_W(x) = \sum_{i=1}^{n} \left(F'_{U_i}(x) \cdot \prod_{j=1, j\neq i}^{n} F_{U_j}(x) \right) \quad .$$

□

Unfortunately, the product of two exponential distributions is not an exponential distribution (see also Example 7.5). Therefore, we take in this section a pragmatic approach by combining individual distributions in such a way that the resulting distribution of a synchronisation action is again exponential. This is achieved by computing the rate of the resulting action from the individual rates of the components according to $\circledast : \mathbb{R}^+ \times \mathbb{R}^+ \to \mathbb{R}^+$. E.g., action a in the composite behaviour $(\lambda)\ a \,||_a\, (\mu)\ a$ will have rate $\lambda \circledast \mu$. Different choices for $\circledast$ are possible. For an extensive discussion on these possibilities, their (stochastic) interpretation, and desired algebraic properties of $\circledast$ we refer to (Götz, 1994; Hillston, 1994b).

We provide a semantics of SL by defining a mapping $\mathcal{X}[\![B]\!]$ which associates a simple stochastic bundle event structure with each expression B of SL. $\mathcal{X}$ is an orthogonal extension of the mapping of process algebra to bundle event structures (cf. Definition 3.1). Let Φ be a function associating to a stochastic behaviour B its corresponding non-stochastic behaviour $\Phi(B)$ by simply omitting the rates in B. In the rest of this section let $\mathcal{X}[\![B_i]\!] = \langle (E_i, \#_i, \mapsto_i, l_i), \mathcal{R}_i \rangle$, for $i = 1, 2$, with $E_1 \cap E_2 = \varnothing$. We assume $\circledast$ to be commutative, associative and have an identity element, denoted $\mathbf{u}$. That is, for all $\lambda \in \mathbb{R}^+$ we have $\lambda \circledast \mathbf{u} = \mathbf{u} \circledast \lambda = \lambda$.

Definition 5.6 $\mathcal{X}[\![\ ]\!]$ is defined recursively as follows:

$$\begin{array}{rcl}
\mathcal{X}[\![\mathbf{0}]\!] & = & \langle \mathcal{E}[\![\Phi(\mathbf{0})]\!], \varnothing \rangle \\
\mathcal{X}[\![(\lambda)\ a\,; B_1]\!] & = & \langle \mathcal{E}[\![\Phi((\lambda)\ a\,; B_1)]\!], \mathcal{R}_1 \cup \{ (e_a, \lambda) \} \rangle \\
\mathcal{X}[\![B_1 + B_2]\!] & = & \langle \mathcal{E}[\![\Phi(B_1 + B_2)]\!], \mathcal{R}_1 \cup \mathcal{R}_2 \rangle \\
\mathcal{X}[\![B_1 \setminus G]\!] & = & \langle \mathcal{E}[\![\Phi(B_1 \setminus G)]\!], \mathcal{R}_1 \rangle \\
\mathcal{X}[\![B_1[H]]\!] & = & \langle \mathcal{E}[\![\Phi(B_1[H])]\!], \mathcal{R}_1 \rangle \\
\mathcal{X}[\![B_1 \,||_G\, B_2]\!] & = & \langle \mathcal{E}[\![\Phi(B_1 \,||_G\, B_2)]\!], \mathcal{R} \rangle \text{ where} \\
\mathcal{R}((e_1, e_2)) & = & \mathcal{R}_1(e_1) \circledast \mathcal{R}_2(e_2) \text{ s.t. } \mathcal{R}_i(*) = \mathbf{u}.
\end{array}$$

□

The definition of $\mathcal{X}$ is exemplified by providing the semantics of the following stochastic behaviours (cf. Figure 5):

$$\begin{array}{ll}
(a)\ B_1 \ = \ (\lambda_1)\ a\,; (\lambda_2)\ b \,||_b\, (\lambda_3)\ c\,; (\lambda_4)\ b & , \\
(b)\ B_2 \ = \ (\mu_1)\ a\,; (\mu_2)\ b \,||_b\, ((\mathbf{u})\ b + (\mu_3)\ d) & , \text{ and} \\
(c)\ B_1 \,||_{\{a,b\}}\, B_2 & .
\end{array}$$

Actions with rate $\mathbf{u}$, the identity of $\circledast$, do not contribute to the resulting rate of a synchronisation. That is, $(\mathbf{u})\ a \,||_a\, (\lambda)\ a$ results in action a with rate λ. Such actions are referred to as *passive* and often occur in performance modelling to model service-like activities.

We conclude this section by discussing *immediate* actions. In performance modelling actions that are irrelevant from a performance evaluation point of view are often considered to take place immediately thus not imposing any additional delay on the system's execution. This led to the notion of immediate transitions

$$\frac{}{(\lambda)\, a_\xi\, ;\, B \xrightarrow{(\xi,a,\lambda)} B}$$

$$\frac{B_1 \xrightarrow{(\xi,a,\lambda)} B_1'}{B_1 + B_2 \xrightarrow{(\xi,a,\lambda)} B_1'} \qquad \frac{B_2 \xrightarrow{(\xi,a,\lambda)} B_2'}{B_1 + B_2 \xrightarrow{(\xi,a,\lambda)} B_2'}$$

$$\frac{B_1 \xrightarrow{(\xi,a,\lambda)} B_1'}{B_1 \,||_G\, B_2 \xrightarrow{((\xi,*),a,\lambda)} B_1' \,||_G\, B_2} \quad (a \notin G) \qquad \frac{B_2 \xrightarrow{(\xi,a,\lambda)} B_2'}{B_1 \,||_G\, B_2 \xrightarrow{((*,\xi),a,\lambda)} B_1 \,||_G\, B_2'} \quad (a \notin G)$$

$$\frac{B_1 \xrightarrow{(\xi,a,\lambda)} B_1' \;\wedge\; B_2 \xrightarrow{(\psi,a,\mu)} B_2'}{B_1 \,||_G\, B_2 \xrightarrow{((\xi,\psi),a,\lambda \circledast \mu)} B_1' \,||_G\, B_2'} \quad (a \in G)$$

$$\frac{B \xrightarrow{(\xi,a,\lambda)} B'}{B \setminus G \xrightarrow{(\xi,a,\lambda)} B' \setminus G} \quad (a \notin G) \qquad \frac{B \xrightarrow{(\xi,a,\lambda)} B'}{B \setminus G \xrightarrow{(\xi,\tau,\lambda)} B' \setminus G} \quad (a \in G)$$

$$\frac{B \xrightarrow{(\xi,a,\lambda)} B'}{B[H] \xrightarrow{(\xi,H(a),\lambda)} B'[H]}$$

TABLE 1. Event transition system for *SL*.

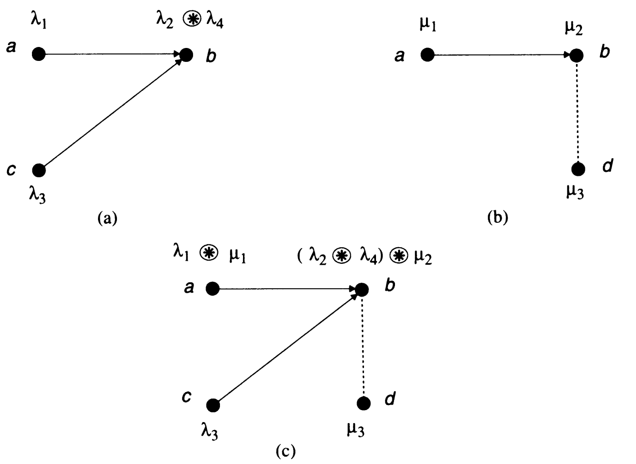


FIGURE 5. Some example simple stochastic event structure semantics.

in stochastic Petri nets, and similarly to the notion of immediate actions (i.e., actions with rate ∞) in stochastic process algebras (e.g. (Bernardo *et al.*, 1994; Götz, 1994)). In our model such actions can easily be incorporated by extending the definition of $\circledast$ such that $\lambda \circledast \infty = \infty$ for all $\lambda \in \mathbb{R}^+ \cup \{\infty\}$.

5.4. Operational Semantics

Various stochastic extensions of process algebras are known that are based on an interleaving semantics. In order to compare our approach to these existing approaches and to investigate the 'compatibility' of our proposal with the standard semantics of process algebras we define an operational semantics for SL that corresponds to the non-interleaving semantics. The approach we follow in this paper is a straightforward generalisation of the case for untimed LOTOS (Langerak, 1992) and quite similar to the approach taken for the deterministic timing case (Katoen *et al.*, 1995). The basic idea is to define a transition system (in the sense of (Plotkin, 1981)) in which we keep track of the occurrence of actions in an expression of SL. This results in a (stochastic) *event transition system.*

In order to define an event transition system each occurrence of an action-prefix is subscripted with an arbitrary but unique event occurrence identifier, denoted by a Greek letter. These occurrence identifiers play the role of event names. E.g. an expression like $a\,;\, b+b$ becomes $a_\xi\,;\, b_\psi + b_\chi$. For parallel composition new event names are created. If e is an event name of B and e' an event name in B', then possible new names for events in $B \,||_G\, B'$ are $(e,*)$ and $(*,e')$ for unsynchronized events and (e,e') for synchronized events.

The transition relation $\longrightarrow$ is defined as the smallest relation closed under all inference rules defined in Table 1. $B \xrightarrow{(e,a,\lambda)} B'$ means that behaviour B can perform event e, labelled a with rate λ and evolve into B'.

Using the transition relation $\longrightarrow$ the notion of (stochastic) event trace can be defined in the usual way. As the transition system induced by $\longrightarrow$ is deterministic, the transition system for B can be represented by its set of stochastic event traces $\mathcal{T}[\![B]\!]$. This set can be characterized in a denotational way, and proven to coincide with the set of stochastic event traces of the corresponding event structure $\mathcal{X}[\![B]\!]$. This proves the consistency between the operational semantics and denotational semantics in terms of event structures.

Theorem 5.7 $\forall B \in SL : STr(\mathcal{X}[\![B]\!]) = \mathcal{T}[\![B]\!]$.

Proof: In a similar way as for the deterministic timing case (Katoen *et al.*, 1995). □

From the event transition system defined by $\longrightarrow$ we can easily obtain the standard inference rules for process algebras like CSP and LOTOS by omitting the rates and event identifiers. In addition, the transition rules strongly resemble the operational semantics of existing stochastic process algebras, and for some of these algebras we obtain identical rules when substituting the appropriate operator for $\circledast$. This provides adequacy for our stochastic causality-based model.

In MTIPP (Götz *et al.*, 1993a; Herrmanns & Rettelbach, 1994) the rate of a synchronized action is simply the product of the rates of the components, thus $\lambda \circledast \mu = \lambda\, \mu$. For B-MPA (Bernardo *et al.*, 1994) the resulting rate is the maximum of the individual rates under the condition that at least one of the participating behaviours must be passive with respect to the interaction, thus, $\lambda \circledast \mu = max(\lambda, \mu)$ given that $\lambda = \mathbf{u}$ or $\mu = \mathbf{u}$. In the first proposal for PEPA (Hillston, 1993) the expected delay (i.e., the reciprocal of the rate) of the interaction is assumed to be the sum of the expected duration of the action in each of the participants, i.e. $\lambda \circledast \mu = (\lambda\, \mu)/(\lambda + \mu)$. In the final proposal for PEPA (Hillston, 1994a) the rate of an interaction is computed by taking into account the total capacity of a behaviour to participate in actions with a certain label (the so-called apparent rate). Since apparent rates are based on the entire behaviour of a participant rather than solely on the (local) rate of an event this synchronization policy cannot be modelled using $\circledast$. In D-MPA (Buchholz, 1994) a somewhat different approach is taken—each action label a is assigned a fixed transition rate μ_a, and $(r)\ a\, ;\, B$ ($r \in \mathbb{R}^+$) denotes a behaviour that may engage in a where the time before a is performed is exponentially distributed with rate $r\, \mu_a$. When $(r_1)\ a$ and $(r_2)\ a$ synchronize the time before interaction a happens is distributed with rate $r_1\, r_2\, \mu_a$. Using $\circledast$ as product on r_i (rather than on rates) assuming that μ_a is given the same scheme can be obtained with the rules of Table 1.

As noted before, desired algebraic properties of $\circledast$ are associativity, commutativity and the existence of an identity element. (Algebraically speaking, this means that $\langle \mathbb{R}^+, \circledast \rangle$ is a commutative, or Abelian, monoid.) Besides these two properties (Götz, 1994; Hillston, 1994b) require $\circledast$ to be distributive over the addition of rates in order to consider $(\lambda)\ a + (\mu)\ a$ and $(\lambda{+}\mu)\ a$ to be equivalent, also in the context of parallel composition (which leads to the distributivity). It is interesting to note that in our model rates are associated to events rather than to actions, and the two a actions in the choice expression above are modelled by distinct events. So, it seems that distributivity of $\circledast$ over $+$ is not a necessary requirement in our model unless distinct events are identified by some congruence relation.

6. GENERALISED STOCHASTIC EVENT STRUCTURES

The main advantage of the model of the previous section is that it is a rather simple extension of bundle event structures which corresponds quite closely to existing stochastic process algebras such as MTIPP (Götz *et al.*, 1993a), PEPA (Hillston, 1993), D-MPA (Buchholz, 1994), and B-MPA (Bernardo *et al.*, 1994) (depending on the choice for $\circledast$). Unfortunately, for keeping the model within the domain of exponential distributions we were unable to let the stochastic variable that determines the delay of the synchronized action be the maximum of the individual stochastic variables, whilst this seems quite reasonable and would be a straightforward generalisation of our deterministic timing model.

In addition, exponential distributions are a bit restrictive in performance modelling and there is a considerable need for more realistic (i.e., non-memoryless) distributions. Especially in the analysis of high-speed communication systems or multi-media applications where the correlation between successive packet arrivals is no longer negligible and packets tend to have a constant length the usual Poisson arrivals and exponential packet lengths are no longer valid assumptions.

In this section we replace the deterministic times associated to bundles and events in our deterministic timing model (cf. Section 4) by stochastic variables having arbitrary distributions, and investigate what the required (algebraic) properties of such distributions are given that the treatment of synchronization is similar to the deterministic case.

6.1. The Model

Distribution functions are added to bundle event structures in two ways. A distribution function associated with event e determines the time between the start of the system and the occurrence of e, while a distribution function associated to bundle $X \mapsto e$ determines the relative time between the occurrence of e and its causal predecessor in X.

The interpretation of bundle $\{ e_a \} \mapsto e_b$ decorated with distribution F is that if e_a has happened at a certain time t_a then the time at which e_b is enabled is determined by $t_a{+}U$ where U is a stochastic variable with distribution F.

If more than one bundle points to an event the following interpretation is chosen. For instance, suppose $\{ e_a \} \mapsto e_c$ and $\{ e_b \} \mapsto e_c$ with distribution F and G, respectively. Now, if e_a (e_b) happens at t_a (t_b) then the time of enabling of e_c is determined by the stochastic variable $max(t_a+U, t_b+V)$, where U (V) has distribution F (G).

As a final example, consider $\{e_a\} \mapsto e_b$ decorated with distribution F and e_b having distribution G. Using a similar reasoning as above, we infer that the stochastic variable $max(U, t_a+V)$ determines the time of enabling of e_b given that e_a happens at time t_a.

Let DF denote an arbitrary class of distribution functions.

Definition 6.1 A *stochastic* bundle event structure Γ is a triple $\langle \mathcal{E}, \mathcal{F}, \mathcal{G}\rangle$ with $\mathcal{E}$ a bundle event structure $(E, \#, \mapsto, l)$, and $\mathcal{F}: E \rightarrow \mathsf{DF}$ and $\mathcal{G}: \mapsto \rightarrow \mathsf{DF}$, associating a distribution function of class DF to events and bundles, respectively. □

We denote a bundle (X, e) with $\mathcal{G}((X, e)) = F$ by $X \overset{F}{\mapsto} e$. Event traces are considered as sequences of events where each event e_i is associated with a stochastic variable U_i that uniquely determines the minimal enabling time of event e_i. The stochastic variable U_i is determined by the distribution function associated with e_i (i.e. $\mathcal{F}(e_i)$), the distributions linked to all bundles pointing to e_i and the stochastic variables U_j of the causal predecessors of e_i in the trace (as these determine the time of occurrence of e_j).

Definition 6.2 A *random event trace* of Γ is a sequence σ of events $(e_1, U_1)\ldots(e_n, U_n)$ with $e_i \in E$, and U_i $(0 < i \leqslant n)$ a stochastic variable with distribution function in class DF iff

1. $e_1 \ldots e_n$ is an event trace of $\mathcal{E}$, and
2. $\forall e_i : U_i = max(U_{\mathcal{F}(e_i)}, V_i)$ where V_i equals

$$Max\{U_G + U_j \mid \exists X : X \overset{G}{\mapsto} e_i \ \wedge\ X \cap \lfloor \sigma_i \rfloor = \{e_j\}\}.$$

□

Notice the resemblance of this definition to Definition 4.3. For distribution function F, U_F denotes the corresponding stochastic variable. V_i is the maximum of n $(n \geqslant 1)$ stochastic variables for n bundles pointing to e_i. In general it is not straightforward to obtain a closed formula for U_i since statistical independence of its constituents cannot always be guaranteed. The stochastic variable $\overline{U} = (U_1, \ldots, U_n)$ spans an n-dimensional hyperspace and has joint distribution function

$$F_{\overline{U}}(\overline{x}) = \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_n} F'_{\overline{U}}(y_1, \ldots, y_n) dy_n \ldots dy_1.$$

Example 6.3 Consider the stochastic event structures in Figure 6. The event distribution of event e_a is denoted F_a and is omitted in the figure for simplicity. For (a) legal traces are $(e_a, U_a)(e_b, U_b)$ and $(e_b, U_b)(e_a, U_a)$ with $U_a = U_{F_a}$ and $U_b = U_{F_b}$. Note that the stochastic variables are equal for both traces. For (b) $(e_a, U_a)(e_b, U_b)$ is a trace with $U_a = U_{F_a}$ and $U_b = max(U_{F_b}, U_G+U_a)$. Finally, for (c) $(e_a, U_a)(e_b, U_b)(e_c, U_c)$ is a trace with $U_a = U_{F_a}$, $U_b = U_{F_b}$ and $U_c = Max\{U_{F_c}, U_G+U_a, U_H+U_b\}$. □

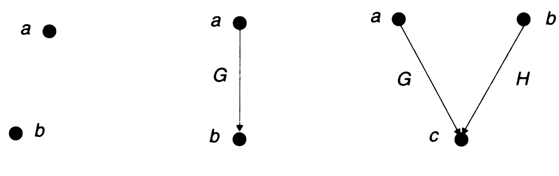


FIGURE 6. Some stochastic bundle event structures.

6.2. A Generalised Stochastic Process Algebra

In this section we use the model of the previous section as a semantical model for a generalised stochastic process algebra. The aim of this exercise is to investigate what the desired algebraic properties of distribution functions are. Let F be a distribution in class DF. The syntax of behaviours is now defined as follows:

$$B \ ::= \ \mathbf{0} \,\big|\, (F)\ a ; B \,\big|\, B + B \,\big|\, B \,||_G\, B \,\big|\, B[H] \,\big|\, B \setminus G.$$

This syntax is identical to the stochastic process algebra of Section 5 except that rates are replaced by arbitrary distribution functions.

In a similar way as for the exponential distribution case we define a mapping $\mathcal{S}[\![B]\!]$ which associates a stochastic bundle event structure to expression B. In the following definition let $\mathcal{S}[\![B_i]\!] = \Gamma_i = \langle (E_i, \#_i, \mapsto_i, l_i), \mathcal{F}_i, \mathcal{G}_i\rangle$, for $i = 1, 2$, with $E_1 \cap E_2 = \varnothing$. We assume that the stochastic variables corresponding to the bundle and event distributions in Γ_1 and Γ_2 are statistically independent. The positive events of Γ are those events that have a distribution function different from $\mathbf{u}$, i.e. $pos(\Gamma) = \{e \in E \mid \mathcal{F}(e) \neq \mathbf{u}\}$. Let $pin(\Gamma) = pos(\Gamma) \cup init(\Gamma)$.

Definition 6.4 $\mathcal{S}[\![\ ]\!]$ is defined recursively as follows:

$$\begin{array}{rcl}
\mathcal{S}[\![\mathbf{0}]\!] & = & \langle \mathcal{E}[\![\Phi(\mathbf{0})]\!], \varnothing, \varnothing\rangle \\
\mathcal{S}[\![(F)\ a ; B_1]\!] & = & \langle \mathcal{E}[\![\Phi((F)\ a ; B_1)]\!], \mathcal{F}, \mathcal{G}\rangle \text{ where} \\
\mapsto & = & \mapsto_1 \cup (\{\{e_a\}\} \times pin(\Gamma_1)) \\
\mathcal{F} & = & (E_1 \times \{\mathbf{u}\}) \cup \{(e_a, F)\} \\
\mathcal{G} & = & \mathcal{G}_1 \cup \\
 & & \{((\{e_a\}, e), \mathcal{F}_1(e)) \mid e \in pin(\Gamma_1)\} \\
\mathcal{S}[\![B_1 + B_2]\!] & = & \langle \mathcal{E}[\![\Phi(B_1 + B_2)]\!], \mathcal{F}, \mathcal{G}\rangle \text{ where} \\
\mathcal{F} & = & \mathcal{F}_1 \cup \mathcal{F}_2 \\
\mathcal{G} & = & \mathcal{G}_1 \cup \mathcal{G}_2 \\
\mathcal{S}[\![B_1 \setminus G]\!] & = & \langle \mathcal{E}[\![\Phi(B_1 \setminus G)]\!], \mathcal{F}_1, \mathcal{G}_1\rangle \\
\mathcal{S}[\![B_1[H]]\!] & = & \langle \mathcal{E}[\![\Phi(B_1[H])]\!], \mathcal{F}_1, \mathcal{G}_1\rangle \\
\mathcal{S}[\![B_1 \,||_G\, B_2]\!] & = & \langle \mathcal{E}[\![\Phi(B_1 \,||_G\, B_2)]\!], \mathcal{F}, \mathcal{G}\rangle \text{ where} \\
\mathcal{F}((e_1, e_2)) & = & \mathcal{F}_1(e_1)\mathcal{F}_2(e_2) \text{ with } \mathcal{F}_i(*) = \mathbf{u} \\
\mathcal{G}(X, (e_1, e_2)) & = & H_1 H_2 \text{ with}
\end{array}$$

$$\begin{aligned} H_1 &= \text{if } \exists X_1 : X_1 \overset{G_1}{\longmapsto}_1 e_1 \;\wedge \\ &\qquad X = \{ (e_i, e_j) \in E \mid e_i \in X_1 \} \\ &\quad \text{then } G_1 \text{ else } \mathbf{u} \\ H_2 &= \text{if } \exists X_2 : X_2 \overset{G_2}{\longmapsto}_2 e_2 \;\wedge \\ &\qquad X = \{ (e_i, e_j) \in E \mid e_j \in X_2 \} \\ &\quad \text{then } G_2 \text{ else } \mathbf{u} \quad . \end{aligned}$$

□

From this definition we infer that the class DF of distribution functions is required to be closed under multiplication and to have an identity element **u** for multiplication. (Recall that the product of distribution functions corresponds to the maximum of their stochastic variables under the assumption of statistical independence.)

Here, in $\mathcal{S}[\![(F)\ a\,;\ B_1]\!]$ a bundle is introduced from a new event e_a (labelled a) to all initial events of Γ_1 and, in addition, to all events in Γ_1 that have a distribution function different from **u**. The distribution of these events is now relative to e_a, so each bundle $\{ e_a \} \mapsto e$ is associated with a distribution $\mathcal{F}_1(e)$, and the distribution $\mathcal{F}(e)$ is made **u**. The distribution $\mathcal{F}(e_a)$ becomes F. In the untimed and exponential case (cf. Definitions 3.1 and 5.6) it suffices to only introduce bundles from e_a to the initial events of Γ_1. Introducing bundles from e_a to all events in $pin(\Gamma_1)$ is, however, semantically equivalent and is used here only to make distributions of events relative to e_a. To exemplify this, Figure 7 depicts (a) $\mathcal{S}[\![B_1]\!]$, and (b) $\mathcal{S}[\![(F)\ a\,;\ B_1]\!]$ (Compare with Figure 2).

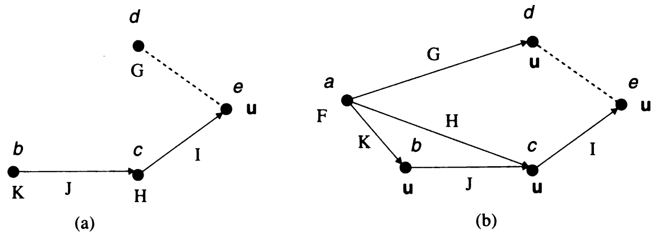


FIGURE 7. Example of timed action prefix.

Finally, we explain the semantics of the parallel composition operator. The events of $\mathcal{S}[\![B_1 \,||_G\, B_2]\!]$ are constructed in the same way as in Definition 5.6. The distribution associated with a bundle is equal to the product of the distribution functions associated with the bundles we get by projecting on the i-th components (i=1, 2) of the events in the bundle, if this projection yields a bundle in $\mathcal{S}[\![B_i]\!]$. The distribution of an event is the product of the distributions of its components that are not equal to $*$.

6.3. Recursion

In this section we extend the behaviours under consideration with recursion. To that end we extend the syntax with the construct $B ::= P$ where P denotes a process instantiation. We assume a behaviour is always considered in the context of a set of process definitions of the form $P := B$ where B is a behaviour possibly containing occurrences of P.

$\mathcal{S}[\![P]\!]$ for $P := B$ is defined in the following way by using standard fixed point theory. A complete partial order (c.p.o.) $\trianglelefteq$ is defined on stochastic bundle event structures with the empty event structure (i.e., $\mathcal{S}[\![\mathbf{0}]\!]$) as the least element $\perp$. Then for each definition $P := B$ a function $\mathcal{F}_B$ is defined that substitutes a stochastic event structure for each occurrence of P in B, interpreting all operators in B as operators on stochastic event structures. $\mathcal{F}_B$ is shown to be continuous, which means that $\mathcal{S}[\![P]\!]$ can be defined as the least upper bound (l.u.b.) of the chain (under $\trianglelefteq$) $\perp$, $\mathcal{F}_B(\perp)$, $\mathcal{F}_B(\mathcal{F}_B(\perp)), \ldots$. For this paper we just define the appropriate ordering $\trianglelefteq$ and the corresponding l.u.b. Given these ingredients it is rather straightforward to define a continuous function $\mathcal{F}_B$ in a similar way as for the non-stochastic case (Langerak, 1992, Chapter 8).

Definition 6.5 Let $\Gamma_i = \langle (E_i, \#_i, \mapsto_i, l_i), \mathcal{F}_i, \mathcal{G}_i \rangle$ for $i = 1, 2$. Then $\Gamma_1 \trianglelefteq \Gamma_2$ iff

1. $E_1 \subseteq E_2$
2. $\#_1 \subseteq \#_2 \;\wedge\; \forall e, e' \in E_1 : e \,\#_2\, e' \;\Rightarrow\; e \,\#_1\, e'$
3. (I) $\forall X_1, e \in E_1$:
   $X_1 \overset{F}{\longmapsto}_1 e \;\Rightarrow\; \exists X_2 : X_2 \overset{F}{\longmapsto}_2 e \;\wedge\; X_1 = X_2 \cap E_1$
   (II) $\forall X_2, e \in E_1 : X_2 \overset{F}{\longmapsto}_2 e \;\Rightarrow\; (X_2 \cap E_1) \overset{F}{\longmapsto}_1 e$
4. $l_2 \upharpoonright E_1 = l_1$
5. $\mathcal{F}_2 \upharpoonright E_1 = \mathcal{F}_1$
6. $\mathcal{G}_2 \upharpoonright \mapsto_1 = \mathcal{G}_1$.

□

where $\upharpoonright$ denotes restriction. The constraints $E_1 \subseteq E_2$ and $\#_1 \subseteq \#_2$ are self-explanatory. In addition we require for conflicts that no new conflicts should appear in Γ_2 between events that are already in Γ_1. Similarly, 3. (II) forbids the introduction of bundles in Γ_2 for events in Γ_1 for which there exists no bundle in Γ_1. These conditions guarantee that a 'larger' stochastic event structure (under $\trianglelefteq$) does allow more event traces. Constraint 3. (I) allows for bundles to grow in such a way that the old bundle is contained and the associated distribution remains the same. It is now straightforward to verify that $\trianglelefteq$ with $\perp = \langle (\varnothing, \varnothing, \varnothing, \varnothing), \varnothing, \varnothing \rangle$ is a c.p.o. A useful property is

Theorem 6.6 $(\Gamma_1 \trianglelefteq \Gamma_2 \wedge E_1 = E_2) \;\Rightarrow\; \Gamma_1 = \Gamma_2$.

Proof: Straightforward by systematically checking the equality of the components of Γ_1 and Γ_2. □

The l.u.b. of a chain $\Gamma_1 \trianglelefteq \Gamma_2 \trianglelefteq \ldots$, denoted $\bigsqcup_i \Gamma_i$, is defined as follows. For the set of events, conflicts, labelling function, and event distributions we simply take the union of all events, conflicts, labellings and event distributions of the event structures in the chain.

As bundles may grow this approach does not apply to the set of bundles. Suppose some Γ_j has bundle $X_j \mapsto_j e$. According to the definition of $\trianglelefteq$ there is a series of bundles $X_j \mapsto_j e$, $X_{j+1} \mapsto_{j+1} e, \ldots$ satisfying $X_{k+1} \cap E_k = X_k$ for $k \geqslant j$. Then the l.u.b. contains bundle $(\bigcup_n X_{j+n}) \mapsto e$. As all bundles in a series retain the same distribution the bundle distribution is simply the union of the bundle distributions of the structures in the chain. Thus,

Definition 6.7 Let $\Gamma_1 \trianglelefteq \Gamma_2 \trianglelefteq \ldots$ be a chain, then $\bigsqcup_i \Gamma_i = \langle (\bigcup_i E_i, \bigcup_i \#_i, \mapsto, \bigcup_i l_i), \bigcup_i \mathcal{F}_i, \bigcup_i \mathcal{G}_i \rangle$ with

$$\begin{aligned} \mapsto \; = \; \{ (X,e) \mid \exists j : \forall k \geqslant j : X_k \mapsto_k e \;\wedge \\ X_{k+1} \cap E_k = X_k \;\wedge\; X = \textstyle\bigcup_k X_k \} \end{aligned}$$

□

It now follows that $\bigsqcup_i \Gamma_i$ indeed is a l.u.b. of the chain $\Gamma_1 \trianglelefteq \Gamma_2 \trianglelefteq \ldots$.

Example 6.8 As an example of the semantics of a recursive process definition, consider $P :=$ (**u**) a; $((F)\ b;\ P + (G)\ c;\ (H)\ d;\ P)$. $\perp$ is the empty bundle event structure. $\mathcal{F}_B(\perp)$ is depicted in Figure 8(a). By repeated substitution we obtain the event structure depicted in Figure 8(b). □

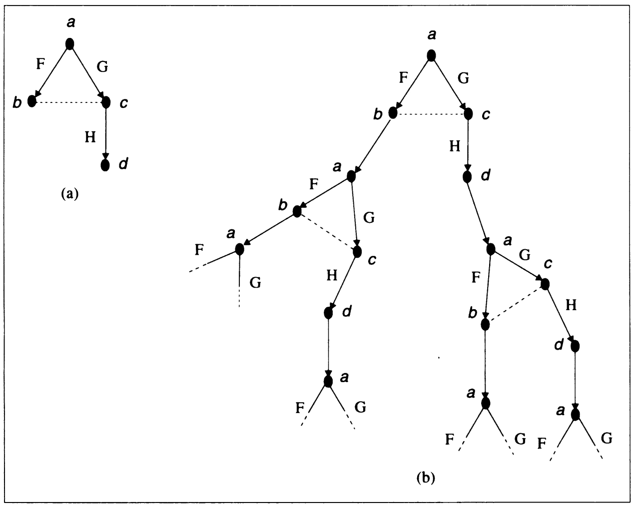


FIGURE 8. Example semantics for process definition.

6.4. Appropriate Distributions

We conclude that the desired properties of the class of distribution functions that is of interest to us are that it should be closed under product and have an identity element for product. An interesting class of distribution functions that satisfy these constraints are the *phase-type (PH-)* distributions. PH-distributions can be considered as matrix generalisations of exponential distributions and are well-suited for numerical computation. They are used in many probabilistic models that have matrix-geometric solutions, have a richly developed theory (Neuts, 1981; Neuts, 1989), and include frequently used distributions such as hyper- and hypo-exponential, Erlang, and Cox distributions. An introduction to PH-distributions and a review of the main results that are of interest to our work (such as the computation of the product of two PH-distributions) is given in Appendix A.

Another interesting class of distribution functions that is closed under product is introduced in (Sahner & Trivedi, 1987). Here, the product of distribution functions of 'exponential polynomial form'

$$F(x) = \sum_i a_i x^{k_i} e^{b_i x} \text{ for } x \geqslant 0.$$

for k_i a natural and a_i, b_i real or complex numbers, is used to model the concurrent execution of groups of tasks. Coxian, exponential, Erlang, and mixtures of exponential distributions also fall into this class of distributions. The applicability of such distributions in the context of our work is for further study.

7. CONCLUDING REMARKS

In this paper we have made an investigation of stochastic extensions of a process algebra in a causality-based setting. We presented a simple event structure model restricted to exponential distributions and a more general one involving PH-distributions. The simple semantic model is shown to be compatible with the standard operational semantics of (ordinary) process algebras like LOTOS and CSP and to closely resemble existing stochastic extensions of interleaved models like MTIPP, B-MPA, D-MPA and a preliminary version of PEPA.

The model involving PH-distributions evolved from a straightforward generalisation of earlier work of the authors in a deterministic timed setting (Katoen *et al.*, 1995; Brinksma *et al.*, 1994). This results in associating distributions to events and bundles. It would be interesting to investigate under which conditions it would be possible to simplify this model and avoid, for instance, distributions being linked with events (e.g. by avoiding timing constraints on initial events).

To our knowledge only a few process algebras exist supporting a much wider class of distribution functions than exponential ones. (Ajmone Marsan *et al.*, 1994) define a stochastic extension of LOTOS in which random variables with arbitrary density functions specify the time lapse between actions. Once an action becomes enabled an experiment is carried out, the outcome of which represents the actual delay of the action. The main limitation of this proposal is that all stochastic timing constraints must be specified at 'top level', thus reducing compositionality and avoiding the issue of how to combine local density functions in case of synchronisation. (Götz *et al.*, 1993b) discuss a generalisation of MTIPP which supports arbitrary distribution functions. In order to associate the appropriate distribution function to actions in the interleaved semantic model, they introduce the notion of 'start references'. Such references are used to keep track of residual lifetimes of stochastic variables. In our model a similar notion is not needed.

Though this paper provides the first basic ingredients to study the (semi-) automated development of performance models out of system specifications in a true concurrent setting, there are a number of issues to be settled. To mention a few, we did not yet address the issue of how to obtain a performance model from an event structure representation while exploiting the explicit parallelism present in the semantics. Some examples of how this could be done starting from an event structure with deterministic times and probabilistic choices can be found in (Brinksma *et al.*, 1994; Katoen *et al.*, 1994). It has to be investigated how this approach carries over to the stochastic case. A comparison with Petri nets is also considered to be useful. The relationship of bundle event structures with Petri nets has been studied by (Boudol & Castellani, 1991) and it would be interesting to extend this study to (non-exponential) stochastic Petri nets.

ACKNOWLEDGEMENTS

The authors like to thank Victor Nicola and Boudewijn Haverkort for helpful discussions and suggestions. The reviewers are kindly acknowledged for their comments on a draft of this paper. The work presented in this paper has been partially funded by C.N.R. - Progetto Bilaterale: Estensioni probabilistiche e temporali dell'algebra di processi LOTOS basate su strutture di eventi, per la specifica e analisi quantitative di sistemi distribuiti, by C.N.R. - Progetto Coordinato: Strumenti per la specifica e verifica di proprieta' critiche di sistemi concorrenti e distribuiti, and by the EU as part of the ESPRIT BRA project 6021: Building Correct Reactive Systems.

REFERENCES

Aceto, L., & Murphy, D. 1993. On the Ill-Timed but Well-Caused. *Pages 97–111 of:* Best, E. (ed), *Concur' 93.* LNCS, vol. 715. Springer-Verlag.

Ajmone Marsan, M., Bianco, A., Ciminiera, L., Sisto, R., & Valenzano, A. 1994. A LOTOS Extension for the Performance Analysis of Distributed Systems. *IEEE/ACM Transactions on Networking*, **2(2)**, 151-164.

Bernardo, M., Donatiello, L., & Gorrieri, R. 1994. Modeling and Analyzing Concurrent Systems with MPA. *Pages 175–189 of:* (Herzog & Rettelbach, 1994).

Boudol, G., & Castellani, I. 1991. *Flow Models of Distributed Computations: Event Structures and Nets.* INRIA Research Rep. 1482, Sophia Antipolis.

Brinksma, E., Katoen, J.-P., Langerak, R., & Latella, D. 1994. Performance Analysis and True Concurrency Semantics. *Pages 309–337 of:* Rus, T., & Rattray, C. (eds), *Theories and Experiences for Real-Time System Development.* AMAST Series in Computing, vol. 2. World Scientific. (ext. abstract in: pp. 157–174 of (Herzog & Rettelbach, 1994).)

Katoen, J.-P., Latella, D., Langerak, R., Brinksma, E. & Bolognesi, T. 1995. A consistent causality-based and interleaved view on a timed process algebra including timeouts. (Submitted for publication.)

Buchholz, P. 1994. Markovian Process Algebra: Composition and Equivalence. *Pages 11–30 of:* (Herzog & Rettelbach, 1994).

Davio, M. 1981. Kronecker Products and Shuffle Algebra. *IEEE Transactions on Computers*, **C-30 (2)**, 116–125.

van Glabbeek, R., Smolka, S., Steffen, B., & Tofts, C. 1990. Reactive, Generative, and Stratified Models of Probabilistic Processes. *Pages 130–141 of: Proc. 5th IEEE Symposium on Logic in Computer Science.*

Götz, N. 1994. *Stochastische Prozessalgebren–Integration von Funktionalem Entwurf und Leistungsbewertung Verteilter Systeme.* Ph.D. thesis, Universität Erlangen-Nürnberg.

Götz, N., Herzog, U., & Rettelbach, M. 1993a. Multiprocessor and Distributed System Design: The Integration of Functional Specification and Performance Analysis Using Stochastic Process Algebras. *In:* Donatiello, L., & Nelson, R. (eds), *Performance Evaluation of Computer and Communication Systems.* LNCS, vol. 729. Springer-Verlag.

Götz, N., Herzog, U., & Rettelbach, M. 1993b. TIPP – Introduction and Application to Protocol Performance Analysis. *In:* König, H. (ed), *Formale Beschreibungstechniken für Verteilte Systeme.* FOKUS series. Munich: Saur publishers.

Hermanns, H., & Rettelbach, M. 1994. Syntax, Semantics, Equivalences, and Axioms for MTIPP. *Pages 71–88 of:* (Herzog & Rettelbach, 1994).

Herzog, U., & Rettelbach, M. (eds). 1994. *Proceedings of the 2nd Workshop on Process Algebras and Performance Modelling.* Erlangen: Universität Erlangen-Nürnberg.

Hillston, J. 1993. *PEPA: Performance Enhanced Process Algebra.* Tech. Rep. CSR-24-93, University of Edinburgh.

Hillston, J. 1994a. *A Compositional Approach to Performance Modelling.* Ph.D. thesis, University of Edinburgh.

Hillston, J. 1994b. The Nature of Synchronisation. *Pages 51–70 of:* (Herzog & Rettelbach, 1994).

Katoen, J.-P., Langerak, R., & Latella, D. 1994. *Modeling Systems by Probabilistic Process Algebra: An Event Structures Approach. Pages 253–268 of:* Tenney, R., Amer, P., & Uyar, M. (eds), *Formal Description Techniques, VI.* IFIP Transactions, vol. C-22. North-Holland.

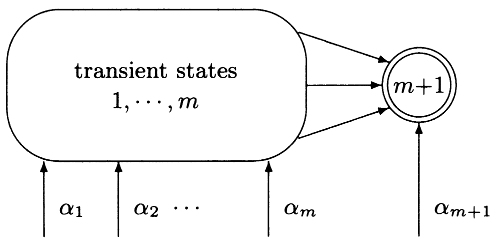


FIGURE 9. Schematic view of a PH-distribution.

Kobayashi, H. 1978. *Modeling and Analysis: An Introduction to System Performance Evaluation Methodology.* Addison-Wesley.

Langerak, R. 1992. *Transformations and Semantics for LOTOS.* Ph.D. thesis, University of Twente.

Neuts, M.F. 1981. *Matrix-geometric Solutions in Stochastic Models–An Algorithmic Approach.* The Johns Hopkins University Press.

Neuts, M.F. 1989. *Structured Stochastic Matrices of M/G/1 Type and Their Applications.* Marcel Dekker, Inc.

Nicollin, X., & Sifakis, J. An Overview and Synthesis of Timed Process Algebras. *Pages 526–548 of:* de Bakker, J.W. (ed), *Real-Time: Theory in Practice.* LNCS, vol. 600. Springer-Verlag.

Plotkin, G.D. 1981. *A Structured Approach to Operational Semantics.* Tech. Rep. DAIMI FM-19, Aarhus University.

Sahner, R.A., & Trivedi, K.S. 1987. Performance and Reliability Analysis Using Directed Acyclic Graphs. *IEEE Transactions on Software Engineering*, **SE-13 (10)**, 1105–1114.

Winskel, G. 1989. An Introduction to Event Structures. *Pages 364–397 of:* de Bakker, J.W., de Roever, W.-P., & Rozenberg, G. (eds), *Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency.* LNCS, vol. 354. Springer-Verlag.

APPENDIX

PH-DISTRIBUTIONS

Intuitively, a PH-distribution is characterised by the time until absorption in a finite-state continuous-time Markov process with a single absorbing state. Consider a continuous time Markov chain (cf. Figure 9) with transient states $\{1, \ldots, m\}$ and absorbing state $m+1$, initial probability vector $(\underline{\alpha}, \alpha_{m+1})$ with $\underline{\alpha}\underline{1} + \alpha_{m+1} = 1$, and (infinitesimal) generator matrix

$$\mathbf{Q} = \begin{pmatrix} \mathbf{T} & \underline{T}^0 \\ 0 & 0 \end{pmatrix} \quad ,$$

where $\mathbf{T}$ is a square matrix of order m such that $\mathbf{T}(i,i) < 0$ and $\mathbf{T}(i,j) \geqslant 0\,(i \neq j)$. The row sums of $\mathbf{Q}$ equal zero, i.e. $\mathbf{T}\underline{1} + \underline{T}^0 = \underline{0}$.

$\mathbf{T}(i,j)$ $(i \neq j)$ can be interpreted as the rate at which the current state changes from transient state i to transient state j. Stated otherwise, starting from state i it takes an exponentially distributed time with mean $1/\mathbf{T}(i,j)$ to reach state j. $\underline{T}^0(i)$ is the rate at which the system can move from transient state i to the absorbing state, state $m+1$. $-\mathbf{T}(i,i)$ is the total rate of departure from state i, or, equivalently, the residence time in state i is exponentially distributed with rate $-1/\mathbf{T}(i,i)$. In general, the transition rates may depend on the time at which a system is considered. In this paper we confine ourselves to Markov chains whose behaviour is invariant to time-shifts. That is, at any time the rate to go from one state to another is the same. Such processes are often referred to as *time-homogeneous* Markov chains.

The probability distribution $F(x)$ of the time until absorption in state $m+1$ is now given by *

$$F(x) = 1 - \underline{\alpha}e^{\mathbf{T}x}\underline{1} \quad ,$$

for $x \geqslant 0$, and $F(x) = 0$, for $x < 0$. The pair $(\underline{\alpha}, \mathbf{T})$ is called a *representation* of F. The corresponding probability density function equals

$$F'(x) = \underline{\alpha}e^{\mathbf{T}x}\underline{T}^0 \quad ,$$

for $x \geqslant 0$, and $F'(x) = 0$, for $x < 0$. The moments μ_i of $F(x)$ are finite and given by

$$\mu_i = (-1)^i i!(\underline{\alpha}\mathbf{T}^{-i}\underline{1}) \text{ for } i = 1, 2, \ldots \quad .$$

The first moment of a stochastic variable corresponds to its expectation, and the difference between the second moment and the square of the first moment corresponds to its variance.

Note the resemblance of the expressions for $F(x)$, $F'(x)$ and μ_i to the corresponding expressions for exponential distributions. In fact, for m=1 we obtain the results for regular exponential distribution. PH-distributions can thus be considered as *matrix generalisations* of the exponential distributions, which makes them suitable for numeric computations.

Definition 7.1 A continuous distribution function F on $[0, \infty)$ is called of *phase-type* (*PH*-distribution) iff it is the distribution of time to absorption in a continuous-time Markov chain as defined above. □

Example 7.2 Example PH-distributions are the exponential, Erlang, hyper- and hypo-exponential, and Coxian distributions. Important to note is that these well-known (PH-type) distributions are acyclic while the definition of PH-type distributions also allows for cyclic Markov chains. Figure 10 illustrates an (a) exponential distribution with rate λ, (b) a 3-stage hyper-exponential distribution with rates λ_i, (c) a 2-stage hypo-exponential distribution with rates λ_i, and (d)

*For square matrix $\mathbf{T}$ of order m, $e^{\mathbf{T}x}$ is defined by $e^{\mathbf{T}x} = \mathbf{I}_m + \mathbf{T}x + \mathbf{T}^2\frac{x^2}{2!} + \mathbf{T}^3\frac{x^3}{3!} + \ldots$, where $\mathbf{I}_m$ denotes the identity matrix of order m and $\mathbf{T}^k\frac{x^k}{k!}$ is matrix $\mathbf{T}^k$ with each element multiplied by $\frac{x^k}{k!}$.

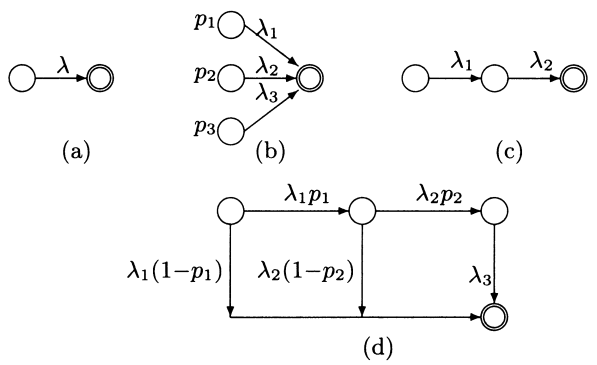


FIGURE 10. Some example PH-distributions.

a 3-phase Coxian distribution. Representations of (b) and (d) are $\underline{\alpha}_{(b)} = (p_1, p_2, p_3)$ with $p_1+p_2+p_3 = 1$, $\underline{\alpha}_{(d)} = (1, 0, 0)$, and

$$\begin{pmatrix} -\lambda_1 & 0 & 0 \\ 0 & -\lambda_2 & 0 \\ 0 & 0 & -\lambda_3 \end{pmatrix}, \quad \begin{pmatrix} -\lambda_1 & \lambda_1 p_1 & 0 \\ 0 & -\lambda_2 & \lambda_2 p_2 \\ 0 & 0 & -\lambda_3 \end{pmatrix}$$

for $\mathbf{T}_{(b)}$ and $\mathbf{T}_{(d)}$, respectively. □

If U and V are statistically independent stochastic variables with PH-distributions G and H respectively, then the distribution F of $W = max(U, V)$ is equal to the product of G and H and is again a PH-distribution. The product of two PH-distributions is calculated as follows (Neuts, 1981, Chapter 2).

Theorem 7.3 Let PH-distributions G, H have representations $(\underline{\alpha}, \mathbf{T})$ and $(\underline{\beta}, \mathbf{S})$ of orders m and n, respectively. Then $F(x) = G(x)H(x)$ is a PH-distribution with representation $(\underline{\gamma}, \mathbf{L})$ of order $mn + m + n$ given by

$$\underline{\gamma} = (\underline{\alpha} \otimes \underline{\beta}, \beta_{n+1}\underline{\alpha}, \alpha_{m+1}\underline{\beta}) \text{ and}$$

$$\mathbf{L} = \begin{pmatrix} \mathbf{T} \otimes \mathbf{I}_n + \mathbf{I}_m \otimes \mathbf{S} & \mathbf{I}_m \otimes \underline{S^0} & \underline{T^0} \otimes \mathbf{I}_n \\ 0 & \mathbf{T} & 0 \\ 0 & 0 & \mathbf{S} \end{pmatrix}.$$

□

$\otimes$ denotes the *tensor* (or Kronecker) *product* and is defined below. Note that $\mathbf{T} \otimes \mathbf{I}_n + \mathbf{I}_m \otimes \mathbf{S}$ is sometimes also referred to as the *tensor sum* of $\mathbf{T}$ and $\mathbf{S}$, denoted $\mathbf{T} \oplus \mathbf{S}$. $\mathbf{T} \oplus \mathbf{S}$ represents the generator matrix of a Markov process which is the cartesian product of the Markov processes represented by $\mathbf{T}$ and $\mathbf{S}$. Tensor algebra is extensively discussed in (Davio, 1981). The PH-distribution consisting only of the absorbing state is the identity under product.

Definition 7.4 The tensor (or Kronecker) product of two matrices $\mathbf{A}$ and $\mathbf{B}$ of orders $r_1 \times c_1$ and $r_2 \times c_2$, respectively, is defined as $\mathbf{C} = \mathbf{A} \otimes \mathbf{B}$ with $\mathbf{C}$ of order $r_1 r_2 \times c_1 c_2$ and

$$\mathbf{C}((i_1-1)r_2 + i_2, (j_1-1)c_2 + j_2) = \mathbf{A}(i_1, j_1)\mathbf{B}(i_2, j_2) \quad ,$$

where $0 < i_k \leqslant r_k$, $0 < j_k \leqslant c_k$ $(k=1, 2)$. □

The resulting matrix $\mathbf{C}$ can be considered as consisting of $r_1 c_1$ blocks each having dimension $r_2 \times c_2$, that is, the dimension of $\mathbf{B}$:

$$\mathbf{C} = \begin{pmatrix} \mathbf{A}(1,1)\mathbf{B} & \mathbf{A}(1,2)\mathbf{B} & \ldots & \mathbf{A}(1,c_1)\mathbf{B} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \mathbf{A}(r_1,1)\mathbf{B} & \mathbf{A}(r_1,2)\mathbf{B} & \ldots & \mathbf{A}(r_1,c_1)\mathbf{B} \end{pmatrix}.$$

The maximum of two PH-distributions is exemplified in the following example.

Example 7.5 Exponential distributions G and H with rates λ and μ have representations $((1), (-\lambda))$ and $((1), (-\mu))$, respectively. The maximum F of these distributions has representation $(\underline{\gamma}, \mathbf{L})$ with $\underline{\gamma} = (1, 0, 0)$ and

$$\mathbf{L} = \begin{pmatrix} -(\lambda+\mu) & \mu & \lambda \\ 0 & -\lambda & 0 \\ 0 & 0 & -\mu \end{pmatrix}.$$

As a second example let G be an exponential distribution with rate λ and H a 2-stage hyperexponential distribution with rates μ_1 and μ_2, and initial probabilities p_1, p_2 with $p_1+p_2 = 1$ (cf. Figure 11(a) and (b)). The maximum F has representation $(\underline{\gamma}, \mathbf{L})$ with $\underline{\gamma} = (p_1, p_2, 0, 0, 0)$ and

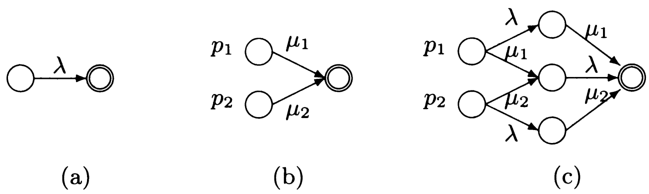


FIGURE 11. Maximum of a 1- and 2-stage hyperexponential distribution.

$$\mathbf{L} = \begin{pmatrix} -(\lambda+\mu_1) & 0 & \mu_1 & \lambda & 0 \\ 0 & -(\lambda+\mu_2) & \mu_2 & 0 & \lambda \\ 0 & 0 & -\lambda & 0 & 0 \\ 0 & 0 & 0 & -\mu_1 & 0 \\ 0 & 0 & 0 & 0 & -\mu_2 \end{pmatrix}.$$

The corresponding Markov process is depicted in Figure 11(c). □