Axiomatizing probabilistic processes : ACP with generative probabilities

Eindhoven University of Technology

Department of Mathematics and Computing Science

Axiomatizing Probabilistic Processes: ACP with Generative Probabilities

by

J.C.M.Baeten J.A.Bergstra S.A.Smolka

92/19

Computing Science Note 92/19
Eindhoven, September 1992

COMPUTING SCIENCE NOTES

This is a series of notes of the Computing Science Section of the Department of Mathematics and Computing Science Eindhoven University of Technology. Since many of these notes are preliminary versions or may be published elsewhere, they have a limited distribution only and are not for review. Copies of these notes are available from the author.

Copies can be ordered from:
Mrs. F. van Neerven
Eindhoven University of Technology
Department of Mathematics and Computing Science
P.O. Box 513
5600 MB EINDHOVEN
The Netherlands
ISSN 0926-4515

Axiomatizing Probabilistic Processes: ACP with Generative Probabilities*

J.C.M. Baeten
Dept. of Math and Computing Science, Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands

J.A. Bergstra
Programming Research Group, University of Amsterdam
P.O. Box 41882, 1009 DB Amsterdam, The Netherlands

S.A. Smolka
Department of Computer Science, SUNY at Stony Brook
Stony Brook, NY 11794-4400, U.S.A.

Abstract

This paper is concerned with finding complete axiomatizations of probabilistic processes. We examine this problem within the context of the process algebra ACP and obtain as our end-result the axiom system $pr\mathrm{ACP}_I^-$, a probabilistic version of ACP which can be used to reason algebraically about the reliability and performance of concurrent systems. Our goal was to introduce probability into ACP in as simple a fashion as possible. Optimally, ACP should be the homomorphic image of the probabilistic version in which the probabilities are forgotten.

We begin by weakening slightly ACP to obtain the axiom system ACP_I^-. The main difference between ACP and ACP_I^- is that the axiom $x + \delta = x$, which does not yield a plausible interpretation in the generative model of probabilistic computation, is rejected in ACP_I^-. We argue that this does not affect the usefulness of ACP_I^- in practice, and show how ACP can be reconstructed from ACP_I^- with a minimal amount of technical machinery.

$pr\mathrm{ACP}_I^-$ is obtained from ACP_I^- through the introduction of probabilistic alternative and parallel composition operators, and a process graph model for $pr\mathrm{ACP}_I^-$ based on *probabilistic bisimulation* is developed. We show that $pr\mathrm{ACP}_I^-$ is a sound and complete axiomatization of probabilistic bisimulation for finite processes, and that $pr\mathrm{ACP}_I^-$ can be homomorphically embedded in ACP_I^- as desired.

Our results for ACP_I^- and $pr\mathrm{ACP}_I^-$ are presented in a modular fashion by first considering several subsets of the signatures. We conclude with a discussion about the suitability of an internal probabilistic choice operator in the context of $pr\mathrm{ACP}_I^-$.

*A preliminary version of this paper appeared in *Proceedings of CONCUR '92 — Third International Conference on Concurrency Theory*, Vol. 630 of the Springer-Verlag series *Lecture Notes in Computer Science*, pp. 472–485, Aug. 1992. The research of the first and second authors was supported by ESPRIT Basic Research Action 7166, CONCUR2. The second author was also supported by RACE project 1046, SPECS. This document does not necessarily reflect the views of the SPECS consortium. The research of the third author was supported by NSF grants CCR–8704309, CCR–9120995, and CCR–9208585.

1 Introduction

It is intriguing to consider the notion of probability (or probabilistic behavior) within the context of process algebra: a formal system of algebraic, equational, and operational techniques for the specification and verification of concurrent systems. Through the introduction of probabilistic measures, one can begin to analyze — in an algebraic fashion — "quantitative" aspects of concurrency such as reliability, performance, and fault tolerance.

In this paper, we address this problem in terms of complete axiomatizations of probabilistic processes within the context of the axiom system ACP [BK84]. ACP models an asynchronous merge, with synchronous communication, by means of arbitrary interleaving. It uses an additional constant δ, which plays the role of *NIL* from CCS [Mil80] (CCS is a predecessor of ACP). The key axioms for δ are:

$x + \delta = x$	A6
$\delta \cdot x = \delta$	A7

The process δ represents an unfeasible option; i.e. a task that cannot be performed and therefore will be postponed indefinitely. The interaction with merge (parallel composition) is as follows:

$$x \parallel \delta = x \cdot \delta$$

(This is not provable from ACP but for each closed process expression p we find that ACP $\vdash p \parallel \delta = p \cdot \delta$.) Now δ represents deadlock according to the explanation of [BK84].

Our goal is to introduce probability into ACP in as simple a fashion as possible. Optimally we would like ACP to be the homomorphic image of the probabilistic version in which the probabilities are forgotten. To this end, we first develop a weaker version of ACP called ACP$_I^-$. This axiom system is just a minor alteration expressing almost the same process identities on finite processes. The virtues of this weaker axiom system are as follows:

(i) ACP$_I^-$ does not imply $x + \delta = x$. In fact, this axiom has often been criticized as being non-obvious for the interpretation δ=deadlock=inaction.

(ii) ACP$_I^-$ + $\{x + \delta = x\}$ implies the same identities on finite processes as ACP (but it is slightly weaker on identities between open processes).

(iii) ACP$_I^-$ has for all practical purposes the same expressiveness as ACP. I.e., if one can specify a protocol in ACP, this can be done just as well in ACP$_I^-$.

(iv) ACP$_I^-$ allows a probabilistic interpretation of $+$, and for this reason we need it as a point of departure for the development of a probabilistic version of ACP.

We introduce probability into ACP$_I^-$ by replacing the operators for alternative and parallel composition with probabilistic counterparts to obtain the axiom system *pr*ACP$_I^-$. Probabilistic choice in *pr*ACP$_I^-$ is of the *generative* variety, as defined in [vGSST90], in that a single probability distribution is ascribed to all alternatives. Consequently, choices involving possibly *different* actions are resolved probabilistically. In contrast, in the *reactive* model of probabilistic computation [LS89, vGSST90], a separate distribution is associated with each action, and choices involving different actions are resolved nondeterministically.

A property of the generative model of probabilistic computation is that, unlike the reactive model, the probabilities of alternatives are conditional with respect to the set of actions offered by the environment. A more detailed comparison of the reactive and generative models can be found in [vGSST90]. There the *stratified* model is also considered and it is shown that the generative model is an abstraction of the stratified model and the reactive model is an abstraction of the generative model.

Previous work on probabilistic process algebra [LS89, GJS90, vGSST90, Chr90, BM89, JL91, CSZ92] has has been primarily of an operational/behavioral nature. Three exceptions, however, are [JS90, Tof90, LS92]. In [JS90], a complete axiomatization of generative probabilistic processes built from a limited set of operators (NIL, action prefix, probabilistic alternative composition, and tail recursion) are provided, while in [Tof90], axioms for synchronously composed "weighted processes" are given. A complete axiomatization of an SCCS-like calculus with reactive probabilities is presented in [LS92].

Summary of Technical Results

We have obtained the following results toward our goal of finding complete axiomatizations of probabilistic processes.

- We first present the axiom system ACP_I^-, our point of departure from ACP. Its development is modular beginning with BPA (consisting of process constants, alternative composition, and sequential composition), to which we add a merge and left-merge operator to obtain PA. Finally, a communication merge operator, the constant δ, and an auxiliary *initials* operator I are added to PA to obtain ACP_I^-. In each case, we present a process graph model based on bisimulation and prove that the system is a sound and complete axiomatization of bisimulation for finite processes.
- We show in a technical sense how ACP can be reconstructed from ACP_I^- through the reintroduction of the axiom A6.
- The axiom systems *pr*BPA, *pr*PA, and *pr*ACP_I^- for probabilistic processes are considered next. In each case, we present a process graph model based on *probabilistic bisimulation*, Larsen and Skou's [LS89] probabilistic extension of strong bisimulation, and prove that the system is a sound and complete axiomatization of probabilistic bisimulation for finite probabilistic processes.
- Connections between ACP_I^- and its probabilistic counterpart are then explored. We show that ACP_I^- is the homomorphic image of *pr*ACP_I^- in which the probabilities are forgotten. This result is obtained for both the graph model — the homomorphism preserves the structure of the bisimulation congruence classes, and the proof theory — the homomorphic image of a valid proof in *pr*ACP_I^- is a valid proof in ACP_I^-.
- We show that certain technical problems arise when a probabilistic internal choice operator is added to *pr*ACP_I^-, and argue that a state operator should be introduced to remedy the situation.

The structure of the rest of this paper is as follows. Section 2 presents the equational specifications BPA, PA, and ACP_I^-, and their accompanying process graph models and completeness results. Section 3 treats the probabilistic versions of these axiom systems, namely, *pr*BPA, *pr*PA, and *pr*ACP_I^-. The homomorphic derivability of ACP_I^- from *pr*ACP_I^- is the subject of Section 4. Section 5 discusses the suitability of an internal probabilistic choice operator in the context of *pr*ACP_I^-, and, finally, Section 6 concludes. Note that we do not treat internal or τ-moves in this paper, so we stay within the setting of concrete process algebra.

2 A Weaker Version of ACP

In this section we present the equational theory ACP_I^-, which, as described in Section 1, will be our point of departure for a probabilistic version of ACP. The main difference between ACP and ACP_I^- is that the axiom $x + \delta = x$, which does not yield a plausible interpretation in the generative model of probabilistic computation, is rejected in ACP_I^-.

As is the practice in ACP, we begin with the theory BPA (Basic Process Algebra) which describes processes constructed from constants, plus, and sequential composition. We will then add to BPA a notion of parallel composition (merge and left-merge) to obtain PA (Process Algebra). Finally, the theory $\mathrm{ACP}_I^-(A)$ is derived by extending BPA with the constant δ (for deadlock), a combined notion of parallel composition and communication, and a restriction operator.

2.1 BPA

2.1.1 Equational Specification

The signature $\Sigma(\mathrm{BPA}(A))$ consists of one sort $\mathbf{P}$ (for processes) and three types of operators: constant processes a, for each atomic action a, the sequential composition (or sequencing) operator '$\cdot$', and the alternative composition (or nondeterministic choice) operator '$+$'. The set of all constants is denoted by A, and is considered a parameter to the theory.

$$\Sigma(\mathrm{BPA}(A)) = \{a : \rightarrow \mathbf{P} | a \in A\} \cup \{+ : \mathbf{P} \times \mathbf{P} \rightarrow \mathbf{P}\} \cup \{\cdot : \mathbf{P} \times \mathbf{P} \rightarrow \mathbf{P}\}$$

The axiom system $\mathrm{BPA}(A)$ is given by:

$x + y = y + x$	A1
$(x + y) + z = x + (y + z)$	A2
$x + x = x$	A3
$(x + y) \cdot z = x \cdot z + y \cdot z$	A4
$(x \cdot y) \cdot z = x \cdot (y \cdot z)$	A5

Note the absence of the axiom $x \cdot (y + z) = x \cdot y + x \cdot z$, which does not hold in our bisimulation model.

2.1.2 Graph Model

We define a process graph model for $\mathrm{BPA}(A)$. The underlying notion of equivalence for process graphs is bisimulation, and we prove completeness of $\mathrm{BPA}(A)$ in this model. We will later extend our graph model to $\mathrm{PA}(A)$ and $\mathrm{ACP}_I^-(A)$. As before, let A be the set of atomic actions. We consider process graphs with labels from A.

Definition 2.1 *A* process graph *g is a triple $< V, r, \longrightarrow >$ such that*

- *V is the set of* nodes *(vertices) of g*
- *$r \in V$ is the* root *of g*

- $\longrightarrow \subseteq V \times A \times V$ *is the* transition relation *of* g

The *endpoints* of g are those nodes devoid of outgoing transitions representing successful termination. The major role played by endpoints is in the definition, given below, of the sequential composition of two process graphs. We often write $v \xrightarrow{a} v'$ to denote the fact that $(v, a, v') \in \longrightarrow$. We denote by $\mathcal{G}$ the family of all process graphs. Bisimulation, due to Milner and Park [Mil80, Par81], is the primary equivalence relation we consider on process graphs.

Definition 2.2 *Let* $g_1 = < V_1, r_1, \longrightarrow_1 >$, $g_2 = < V_2, r_2, \longrightarrow_2 >$ *be two process graphs. A* bisimulation *between* g_1 *and* g_2 *is a relation* $\mathcal{R} \subseteq V_1 \times V_2$ *with the following properties:*

- $\mathcal{R}(r_1, r_2)$
- $\forall v \in V_1,\ w \in V_2$ *with* $\mathcal{R}(v, w)$*:*
  $\forall a \in A$ *and* $v' \in V_1$,
  if $v \xrightarrow{a}_1 v'$ *then* $\exists w' \in V_2$ *with* $\mathcal{R}(v', w')$ *and* $w \xrightarrow{a}_2 w'$
- *and* vice versa *with the roles of* v *and* w *reversed.*

Graphs g_1 *and* g_2 *are said to be* bisimilar, *written* $g_1 \leftrightarrows g_2$, *if there exists a bisimulation between* g_1 *and* g_2.

We now define the operators from $\Sigma(\text{BPA}(A))$ on the domain $\mathcal{F}$ of *finite* process graphs, i.e., process graphs that are finitely branching and acyclic in their transition relations. Therefore, $\mathcal{F} \subset \mathcal{G}$. For this purpose, it is convenient to assume that a process graph root-node is not an endpoint. For the remainder of Section 2, unless otherwise stated, let $g_1 = < V_1, r_1, \longrightarrow_1 >$, $g_2 = < V_2, r_2, \longrightarrow_2 >$ be finite process graphs satisfying the non-endpoint root assumption such that $V_1 \cap V_2 = \emptyset$.

Definition 2.3 *The operators* $a \in A$, $+$, *and* $\cdot$ *are defined on* $\mathcal{F}$ *as follows:*

$a \in A$: *The process graph for each of these constants consists of a single transition and is given by* $< \{r_a, v\}, r_a, \{< r_a, a, v >\} >$.

$g_1 + g_2$: *is given by* $< V_1 \cup V_2 \cup \{r\}, r, \longrightarrow >$ *such that* $r \notin V_1 \cup V_2$ *and* $v \xrightarrow{a} v'$ *if one or more of following holds:*

- $r_1 \xrightarrow{a}_1 v'$ *and* $v = r$
- $r_2 \xrightarrow{a}_2 v'$ *and* $v = r$
- $v \xrightarrow{a}_1 v'$
- $v \xrightarrow{a}_2 v'$

$g_1 \cdot g_2$: *is obtained by appending a copy of* g_2 *at each endpoint of* g_1. *In detail,* $g_1 \cdot g_2$ *is given by* $< V_1 \times V_2, (r_1, r_2), \longrightarrow >$ *where* $(q_1, q_2) \xrightarrow{a} (q_1', q_2')$ *if one or more of the following holds:*

- $q_1 \xrightarrow{a}_1 q_1'$ *and* $q_2 = q_2' = r_2$
- $q_2 \xrightarrow{a}_2 q_2'$ *and* $q_1 = q_1'$ *is an endpoint*

For t a closed BPA(A) term, we write $graph(t) = \langle V_t, r_t, \longrightarrow_t \rangle$ to denote the process graph obtained inductively on t using Definition 2.3. We take the liberty to write expressions like $p \leftrightarrow q$, instead of the more precise $graph(p) \leftrightarrow graph(q)$, when this is clear from the context. The definition of $graph(t)$ and the just-mentioned notational liberty extend in the obvious way to the axiom systems PA(A) and $\mathrm{ACP}_I^-(A)$, to be considered later in this section.

In the setting of BPA, $\leftrightarrow$ is a congruence (see, e.g., [BW90]).

Proposition 2.1 *If $g_1 \leftrightarrow g_2$, then $g + g_1 \leftrightarrow g + g_2$, $g \cdot g_1 \leftrightarrow g \cdot g_2$, and $g_1 \cdot g \leftrightarrow g_2 \cdot g$.*

We have that $\mathcal{F}/\leftrightarrow$, the *graph model*, is indeed a model of the axiom system BPA(A), and that BPA(A) constitutes a complete axiomatization of process equivalence in $\mathcal{F}/\leftrightarrow$.

Theorem 2.1 ([BW90])

1. $\mathcal{F}/\leftrightarrow \models \mathrm{BPA}(A)$
2. *For all closed expressions p,q over* $\Sigma(\mathrm{BPA}(A))$*:*

$$\mathcal{F}/\leftrightarrow \models p = q \implies \mathrm{BPA}(A) \vdash p = q.$$

2.2 PA

2.2.1 Equational Specification

The signature $\Sigma(\mathrm{PA}(A))$ is obtained from $\Sigma(\mathrm{BPA}(A))$ by adding an interleaving *merge* operator $\|$ and a *left-merge* operator $⌊\!⌊$.

$$\Sigma(\mathrm{PA}(A)) = \Sigma(\mathrm{BPA}(A)) \cup \{\|: \mathbf{P} \times \mathbf{P} \to \mathbf{P}\} \cup \{⌊\!⌊: \mathbf{P} \times \mathbf{P} \to \mathbf{P}\}$$

Intuitively, the process $x \| y$ is obtained by interleaving (shuffling) the atomic actions of x and y together. Left-merge is an auxiliary operator in that it permits $\|$ to be specified in finitely many equations. The process $x ⌊\!⌊ y$ has the same meaning as $x \| y$, but with the restriction that the first step must come from x.

The axiom system PA(A) is given by:

BPA(A) +

$x \| y = x ⌊\!⌊ y + y ⌊\!⌊ x$	M1
$a ⌊\!⌊ x = a \cdot x$	M2
$(a \cdot x) ⌊\!⌊ y = a \cdot (x \| y)$	M3
$(x + y) ⌊\!⌊ z = x ⌊\!⌊ z + y ⌊\!⌊ z$	M4

2.2.2 Graph Model

The two new operators of PA(A) are now defined on finite process graphs (as before, with non-endpoint roots).

Definition 2.4 *The operators* $\|$ *and* $\lfloor\!\lfloor$ *are defined on* $\mathcal{F}$ *as follows:*

$g_1 \| g_2$: *is given by* $< V_1 \times V_2, (r_1, r_2), \longrightarrow >$ *where* $(v_1, v_2) \xrightarrow{a} (v_1', v_2')$ *if either of the following holds:*

- $v_1 \xrightarrow{a}_1 v_1'$ *and* $v_2 = v_2'$
- $v_2 \xrightarrow{a}_2 v_2'$ *and* $v_1 = v_1'$

$g_1 \lfloor\!\lfloor g_2$: *As* $g_1 \| g_2$ *but without transitions of the form* $(r_1, r_2) \xrightarrow{a} (r_1, v)$.

Again one may notice that $\leftrightarrows$ is a congruence, $\mathcal{F}/\leftrightarrows \models$ PA(A) and that PA(A) constitutes a complete axiomatization of process equivalence in $\mathcal{F}/\leftrightarrows$ [BW90].

2.3 ACP without A6

2.3.1 Equational Specification

The equational system ACP$_I^-(A)$ treats the operators of BPA(A) as well as the new constant δ representing deadlock; a *communication merge* operator $|$ describing the result of a communication between any two atomic actions; a *merge* operator $\|$ and *left-merge* operator $\lfloor\!\lfloor$ like those of PA(A) but which additionally admit the possibility of communication; and a family of restriction operators ∂_H, $H \subseteq A$. We will also need an auxiliary operator I that defines the initial actions that a process can perform.

Letting $A_\delta = A \cup \{\delta\}$, the signature of ACP$_I^-(A)$ extends that of PA(A) as follows:

$$\begin{aligned}\Sigma(\mathrm{ACP}_I^-(A)) \;=\; & \Sigma(\mathrm{PA}(A)) \;\cup\; \{\delta :\to \mathbf{P}\} \;\cup\; \{| : \mathbf{P} \times \mathbf{P} \to \mathbf{P}\} \;\cup\; \{\partial_H : \mathbf{P} \to \mathbf{P} \mid H \subseteq A\} \;\cup \\ & \{I : \mathbf{P} \to 2^{A_\delta}\}\end{aligned}$$

It is convenient to define communication merge as a binary commutative and associative function on atomic actions (i.e., $| : A_\delta \times A_\delta \to A_\delta$) with δ acting as a multiplicative zero. This is accomplished with axioms C1–3 below. We further require $|$ to be total and to capture this axiomatically we need a way to effectively enumerate all the constant processes. For this purpose, we define the characteristic predicate $\overline{A_\delta}$ of A_δ in the usual way:

$$\overline{A_\delta}(x) = \bigvee_{a \in A_\delta} (x = a)$$

The totality of $|$ is now given by the following axiom: axiom:[1]

$$\boxed{\forall a, b \in \mathbf{P}\ \overline{A_\delta}(a) \wedge \overline{A_\delta}(b) \implies \exists c \in \mathbf{P}\ \overline{A_\delta}(c) \wedge a|b = c \quad \mathrm{C0}}$$

[1]Axiom C0 is often replaced by choosing a total function $\gamma : A_\delta \times A_\delta \to A_\delta$ and having all identities of the graph of γ as axioms: $a|b = \gamma(a,b)$. In this way, γ becomes another parameter to the theory (see, e.g., [BW90]).

The axioms of $\mathrm{ACP}_I^-(A)$ are now given. In this system, a,b,c range over A_δ, $H_\delta = H \cup \{\delta\}$, and $\cap$,$\cup$ are used on 2^{A_δ} without further specification.

BPA(A) +

$\delta \cdot x = \delta$	A7

+

C0 +

$a \vert b = b \vert a$	C1
$(a \vert b) \vert c = a \vert (b \vert c)$	C2
$\delta \vert a = \delta$	C3

+

$x \parallel y = x \mathbin{\lfloor\!\lfloor} y + y \mathbin{\lfloor\!\lfloor} x + x \vert y$	CM1
$a \mathbin{\lfloor\!\lfloor} x = a \cdot x$	CM2
$(a \cdot x) \mathbin{\lfloor\!\lfloor} y = a(x \parallel y)$	CM3
$(x + y) \mathbin{\lfloor\!\lfloor} z = (x \mathbin{\lfloor\!\lfloor} z) + (y \mathbin{\lfloor\!\lfloor} z)$	CM4
$a \vert (b \cdot x) = (a \vert b) \cdot x$	CM5
$(a \cdot x) \vert b = (a \vert b) \cdot x$	CM6
$(a \cdot x) \vert (b \cdot y) = (a \vert b) \cdot (x \parallel y)$	CM7
$(x + y) \vert z = x \vert z + y \vert z$	CM8
$x \vert (y + z) = x \vert y + x \vert z$	CM9

+

$I(a) = \{a\}$	I1
$I(x \cdot y) = I(x)$	I2
$I(x + y) = I(x) \cup I(y)$	I3

+

$a \in H \implies \partial_H(a) = \delta$	D1
$a \notin H \implies \partial_H(a) = a$	D2
$I(x) \subseteq H_\delta \implies \partial_H(x + y) = \partial_H(y)$	D3.1
$I(x + y) \cap H_\delta = \emptyset \implies \partial_H(x + y) = \partial_H(x) + \partial_H(y)$	D3.2
$\partial_H(x \cdot y) = \partial_H(x) \cdot \partial_H(y)$	D4

Comments: $\mathrm{ACP}_I^-(A)$ differs from ACP by the absence of A6 and the presence of D3.1-2 instead of axiom D3: $\partial_H(x+y) = \partial_H(x) + \partial_H(y)$. The following examples illustrate the new axiom system.

$$\begin{array}{rcll}
\partial_{\{c\}}(a+(b+c)) & = & \partial_{\{c\}}(c+(a+b)) & \text{(by A1 and A2)} \\
& = & \partial_{\{c\}}(a+b) & \text{(by D3.1)} \\
& = & \partial_{\{c\}}(a) + \partial_{\{c\}}(b) & \text{(by D3.2)} \\
& = & a+b & \text{(by D2 twice)}
\end{array}$$

$$\begin{array}{rcll}
\partial_{\{a\}}(a+\delta) & = & \partial_{\{a\}}(\delta+a) & \text{(by A1)} \\
& = & \partial_{\{a\}}(a) & \text{(by D3.1)} \\
& = & \delta & \text{(by D1)}
\end{array}$$

$$\begin{array}{rcll}
\partial_{\{a\}}(a+\delta) & = & \partial_{\{a\}}(\delta) & \text{(by D3.1)} \\
& = & \delta & \text{(by D2)}
\end{array}$$

2.3.2 Graph Model

Let *initials*$(v) \subseteq A_\delta$ be the set of actions $\{a \in A_\delta \mid \exists v' \; v \xrightarrow{a} v'\}$ for v a process graph node. The operators of $\mathrm{ACP}_I^-(A)$, beyond those of BPA(A), are now defined on finite process graphs (with non-endpoint roots).

Definition 2.5 *The* $\mathrm{ACP}_I^-(A)$ *operators* δ, $\|$, $\lfloor\!\lfloor$, $|$, ∂_H *(for* $H \subseteq A$*), and* I *are defined on* $\mathcal{F}$ *as follows:*

δ: *is given by* $<\{r_\delta, v_\delta\}, r_\delta, \{<r_\delta, \delta, v_\delta>\}>$.

$g_1 \parallel g_2$: *is given by* $<V_1 \times V_2, (r_1, r_2), \longrightarrow>$ *where* $(v_1, v_2) \xrightarrow{a} (v_1', v_2')$ *if one or more of the following holds:*

- $v_1 \xrightarrow{a}_1 v_1'$ *and* $v_2 = v_2'$
- $v_2 \xrightarrow{a}_2 v_2'$ *and* $v_1 = v_1'$
- $v_1 \xrightarrow{b}_1 v_1'$, $v_2 \xrightarrow{c}_2 v_2'$, *and* $a = b|c$ *(for some* b *and* c*)*

$g_1 \lfloor\!\lfloor g_2$: *As* $g_1 \parallel g_2$ *but without transitions of the form* $(r_1, r_2) \xrightarrow{a} (r_1, v)$.

$g_1|g_2$: *As* $g_1 \parallel g_2$ *but without transitions of the form* $(r_1, r_2) \xrightarrow{a} (v, r_2)$ *or* $(r_1, r_2) \xrightarrow{a} (r_1, v)$.

$\partial_H(g_1)$: *is given by* $<V_1, r_1, \longrightarrow>$ *where*

$$\begin{array}{rl}
\longrightarrow \; = & \{(v, a, v') \in \longrightarrow_1 \mid a \notin H_\delta\} \;\cup \\
& \{(v, \delta, v') \mid (v, a, v') \in \longrightarrow_1 \text{ and } \mathit{initials}(v) \subseteq H_\delta\}
\end{array}$$

$I(g_1)$: *gives the set of actions initials*(r_1).

Our algebra of process graphs is standard (see, e.g., [BW90]) with the exception of restriction. This operator removes all edges labeled with actions from the set of restricted actions H. It also removes δ-edges, which it must do to ensure the soundness of D3.1. In case a node in g_1 qualifies to have all its edges removed, then these edges are not removed but rather renamed into δ-transitions.

The presence of δ-transitions, which intuitively represent the capability for a process to deadlock, requires a new definition of bisimulation in which a weaker condition is imposed on δ-transitions.

Definition 2.6 *Let $g_1 = < V_1, r_1, \longrightarrow_1 >$, $g_2 = < V_2, r_2, \longrightarrow_2 >$ be two process graphs. A δ-*bisimulation *between g_1 and g_2 is a relation $\mathcal{R} \subseteq V_1 \times V_2$ with the following properties:*

- $\mathcal{R}(r_1, r_2)$
- $\forall v \in V_1,\ w \in V_2$ *with* $\mathcal{R}(v, w)$:
  - $\forall a \in A$ *and* $v' \in V_1$,
    if $v \xrightarrow{a}_1 v'$ *then* $\exists w' \in V_2$ *with* $\mathcal{R}(v', w')$ *and* $w \xrightarrow{a}_2 w'$
  - *if* $v \xrightarrow{\delta}_1 v'$, *for some* v', *then* $w \xrightarrow{\delta}_2 w'$, *for some* w'
- *and* vice versa *with the roles of v and w reversed.*

*Graphs g_1 and g_2 are δ-*bisimilar, *written $g_1 \leftrightarrows_\delta g_2$, if there exists a δ-bisimulation between g_1 and g_2.*

This definition is the same as Definition 2.2 with the additional stipulation that for two nodes v, w related by a δ-bisimulation, v possesses a δ-edge iff w does. We once again have that $\leftrightarrows_\delta$ is a congruence.

Proposition 2.2 *If $g_1 \leftrightarrows_\delta g_2$, then $g \parallel g_1 \leftrightarrows_\delta g \parallel g_2$, $g \mathbin{⌊\!⌊} g_1 \leftrightarrows_\delta g \mathbin{⌊\!⌊} g_2$, $g_1 \mathbin{⌊\!⌊} g \leftrightarrows_\delta g_2 \mathbin{⌊\!⌊} g$, $g|g_1 \leftrightarrows_\delta g|g_2$ and $\partial_H(g_1) \leftrightarrows_\delta \partial_H(g_2)$, for all $H \subseteq A$.*

Proof: The proof for all operators, except ∂_H, follows the standard arguments of ACP (see, e.g., [BW90]). For ∂_H, $H \subseteq A$, the proof proceeds as follows. Suppose $g_1 \leftrightarrows_\delta g_2$ and let $\mathcal{R} \subseteq V_1 \times V_2$ be a δ-bisimulation between g_1 and g_2. We show that $\mathcal{R}$ is also a δ-bisimulation between $\partial_H(g_1)$ and $\partial_H(g_2)$, $H \subseteq A$.

Let $(v_1, v_2) \in \mathcal{R}$. There are three cases to consider:

initials$(v_1) \not\subseteq H_\delta$: then in $\partial_H(g_1)$ the transitions of v_1 are of the form $v_1 \xrightarrow{a} v_1'$ with $a \notin H_\delta$. Since $g_1 \leftrightarrows_\delta g_2$, in $\partial_H(g_2)$ there exists a v_2' with $\mathcal{R}(v_1', v_2')$ and $v_2 \xrightarrow{a} v_2'$.

initials$(v_1) \neq \emptyset \subseteq H_\delta$: then in $\partial_H(g_1)$ all transitions of v_1 are of the form $v_1 \xrightarrow{\delta} v_1'$. Since $g_1 \leftrightarrows_\delta g_2$, in $\partial_H(g_2)$ all transitions of v_2 are likewise of the form $v_2 \xrightarrow{\delta} v_2'$. By the weaker condition on δ-transitions in a δ-bisimulation, this is enough.

initials$(v_1) = \emptyset$: then *initials*$(v_1) = \emptyset$ in $\partial_H(g_1)$ and, since $g_1 \leftrightarrows_\delta g_2$, *initials*$(v_2) = \emptyset$ in $\partial_H(g_2)$.

By considering the same three cases with the roles of v_1 and v_2 reversed, we are done. □

To prove the completeness of $\mathrm{ACP}_I^-(A)$ for finite processes, we first introduce the notion of a "basic term" for closed $\mathrm{ACP}_I^-(A)$ terms. We will subsequently prove an "elimination theorem" stating that any closed $\mathrm{ACP}_I^-(A)$ term can be reduced to a basic term using the axioms of $\mathrm{ACP}_I^-(A)$. Combined with the completeness of BPA(A), this will be enough to prove the completeness of $\mathrm{ACP}_I^-(A)$.

Definition 2.7 *A* basic term *is defined inductively as follows:*

- *$a \in A_\delta$ is a basic term.*
- *Let t_1, t_2 be basic and $a \in A$. Then $t_1 + t_2$ and $a \cdot t_1$ are basic.*

Note that a basic term uses a restricted form of sequential composition known as action prefixing, and that a basic term is a BPA(A_δ) term; i.e., a BPA(A) term treating δ as an additional atomic action.

To prove the elimination theorem we introduce a term rewriting system based on $\mathrm{ACP}_I^-(A)$ for which we prove a strong normalization result. The rewrite system $\mathrm{RACP}_I^-(A)$ consists of axioms A1-5, A7, C3, CM1-9, I1-3, and D1-2, treated as rewrite rules with left-to-right orientation, plus the rules

$$\begin{array}{ll}
x + (y + z) \longrightarrow (x + y) + z & \mathrm{A2'} \\
a|b = c \implies a|b \longrightarrow c & \mathrm{C0'} \\
a|\delta \longrightarrow \delta & \mathrm{C3'} \\
c \in H_\delta \implies \partial_H(c + x) \longrightarrow \partial_H(x) & \mathrm{D3.1'} \\
c \in H_\delta \implies \partial_H(c \cdot x + y) \longrightarrow \partial_H(y) & \mathrm{D3.1''} \\
I(x + y) \cap H_\delta = \emptyset \implies \partial_H(x + y) \longrightarrow \partial_H(x) + \partial_H(y) & \mathrm{D3.2'} \\
\partial_H(a \cdot x) \longrightarrow \partial_H(a) \cdot \partial_H(x) & \mathrm{D4'}
\end{array}$$

Notice that all these rules follow easily from $\mathrm{ACP}_I^-(A)$. The normal forms of the rewrite system $\mathrm{RACP}_I^-(A)$ are defined as follows.

Definition 2.8 *A closed* $\mathrm{ACP}_I^-(A)$ *term t is in* normal form *if for all* $\mathrm{RACP}_I^-(A)$ *reduction paths of the form*

$$t = t_0 \longrightarrow t_1 \longrightarrow t_2 \longrightarrow \cdots$$

t_{i+1} follows from t_i through the application of either rule A1, A2, *or* A2′ *(and no other), for all $i \geq 0$.*

Proposition 2.3 *A normal form is a basic term.*

Proof: Let t be a normal form and suppose t is not basic. Let t' be a minimal subterm of t that is not basic. Then t' has one of the following forms:

1. $p \parallel q$
2. $p \mathbin{\lfloor\!\lfloor} q$
3. $p|q$
4. $\partial_H(p)$
5. $p \cdot q$ (with p not an atom or $p = \delta$)

and both p and q basic terms due to minimality. We show that in each case a rule of $\mathrm{RACP}_I^-(A) - \{\mathrm{A1}, \mathrm{A2}, \mathrm{A2'}\}$ can still be applied, thereby proving the result by contradiction. Take, for example, the second case. Since p is a basic term, there are three subcases to consider:

(a) p is of the form $p_1 + p_2$. Apply CM4.

(b) p is an atomic action $a \in A_\delta$. Apply CM2.

(c) p is of the form $a \cdot p_1$, $a \in A_\delta$. Apply CM3.

The other four cases are proved similarly. □

Note that the converse of this result does not hold, e.g., $a + a$ is basic but not in normal from.

Lemma 2.1 *The rewrite system* $\mathrm{RACP}_I^-(A)$ *is strongly normalizing modulo* A1, A2, A2′, *i.e., every infinite reduction path contains* A1, A2, A2′ *steps only from some point onwards.*

Proof: Let $\Pi = (\gamma_0, t_0) \longrightarrow (\gamma_1, t_1) \longrightarrow (\gamma_2, t_2) \longrightarrow \cdots$ be an infinite reduction path in $\mathrm{RACP}_I^-(A)$ where γ_i is the (possibly empty) condition associated with rewriting t_i into t_{i+1}. We omit from Π any steps having to do with normalizing the expression $I(x+y)$ in the condition to D3.2′-steps. We prove that only finitely many of the steps in Π can differ from A1, A2, A2′.

We transform the reduction sequence Π into a reduction sequence Π' of RACP(A) [BK84] as follows:

- Expand each D3.1′ step of the form $(\gamma_i, t_i) \longrightarrow (\gamma_{i+1}, t_{i+1})$ into a finite valid rewriting of RACP(A) depending on the condition γ_i as follows:
  - $c = \delta$: $\partial_H(\delta + x) \xrightarrow{\mathrm{A1}} \partial_H(x + \delta) \xrightarrow{\mathrm{A6}} (\gamma_{i+1}, \partial_H(x))$
  - $c \in H$: $(c \in H, \partial_H(c + x)) \xrightarrow{\mathrm{A3}} (c \in H, \partial_H(c) + \partial_H(x)) \xrightarrow{\mathrm{D2}} \delta + \partial_H(x) \xrightarrow{\mathrm{A1}} \partial_H(x) + \delta \xrightarrow{\mathrm{A6}} (\gamma_{i+1}, \partial_H(x))$
- Expand each D3.1″ step of the form $(\gamma_i, t_i) \longrightarrow (\gamma_{i+1}, t_{i+1})$ into a finite valid rewriting of RACP(A) depending on the condition γ_i as follows:
  - $c = \delta$: $\partial_H(\delta \cdot x + y) \xrightarrow{\mathrm{A7}} \partial_H(\delta + y) \xrightarrow{\mathrm{A1}} \partial_H(y + \delta) \xrightarrow{\mathrm{A6}} (\gamma_{i+1}, \partial_H(y))$
  - $c \in H$: $(c \in H, \partial_H(c \cdot x + y)) \xrightarrow{\mathrm{D3}} \partial_H(c \cdot x) + \partial_H(y) \xrightarrow{\mathrm{D4}} (c \in H, \partial_H(c) \cdot \partial_H(x) + \partial_H(y)) \xrightarrow{\mathrm{D2}} \delta \cdot \partial_H(x) + \partial_H(y) \xrightarrow{\mathrm{A7}} \delta + \partial(y) \xrightarrow{\mathrm{A1}} \partial_H(y) + \delta \xrightarrow{\mathrm{A6}} (\gamma_{i+1}, \partial_H(y))$
- Transform each D3.2′ step of the form $(\gamma_i, t_i) \longrightarrow (\gamma_{i+1}, t_{i+1})$ into the conditionless step $t_i \longrightarrow (\gamma_{i+1}, t_{i+1})$, as D3.2′ is valid in RACP(A) in all cases (i.e., restriction distributes over plus).[2]

Now we obtain an infinite reduction path in RACP(A) and from [BW90] it follows that this reduction path contains finitely many non-A1, A2, A2′ steps. But the same must hold for the original reduction sequence. □

Note that in the transformation of a $\mathrm{RACP}_I^-(A)$ reduction sequence to a RACP(A) reduction sequence, each non-A1, A2, A2′ step is replaced by at most six non-A1, A2, A2′ steps.

We now present the "elimination theorem" for $\mathrm{ACP}_I^-(A)$.

Lemma 2.2 *Let p be a closed* $\mathrm{ACP}_I^-(A)$ *term. Then using* $\mathrm{RACP}_I^-(A)$, *p can be reduced in finitely many steps to a basic term.*

[2] One could, of course, leave condition γ_i intact and still have a valid reduction step in RACP(A).

Proof: If p is a basic term we are done. Otherwise, by Proposition 2.3, p is not in normal form. By Definition 2.8, there exists a reduction sequence

$$p = p_0 = t_0^0 \longrightarrow t_1^1 \longrightarrow \cdots \longrightarrow t_{n_0}^0 = p_1$$

such that $t_{n_0-1}^0 \longrightarrow t_{n_0}^0$ is not an A1, A2, A2′ reduction. If p_1 is basic we are done. Otherwise there exists another reduction sequence

$$p_1 = t_0^1 \longrightarrow t_1^1 \longrightarrow \cdots \longrightarrow t_{n_1}^1 = p_2$$

such that $t_{n_1-1}^1 \longrightarrow t_{n_1}^1$ is not an A1, A2, A2′ reduction. This line of reasoning cannot proceed indefinitely: due to strong normalization (Lemma 2.1) p_i, for some $i \geq 0$, is a basic term. Otherwise, an infinite reduction with infinitely many non-A1, A2, A2′ steps would have been constructed which is impossible. □

Theorem 2.2

1. $\mathcal{F}/\leftrightarrows_\delta \models \mathrm{ACP}_I^-(A)$
2. *For all closed expressions p,q over* $\Sigma(\mathrm{ACP}_I^-(A))$*:*

$$\mathcal{F}/\leftrightarrows_\delta \models p = q \implies \mathrm{ACP}_I^-(A) \vdash p = q.$$

Proof: For part 1, we consider axioms A7 and D1-D4. The fact that $\mathcal{F}/\leftrightarrows_\delta$ is a model of the rest of the axioms of $\mathrm{ACP}_I^-(A)$ follows standard arguments as presented, e.g., in [BW90]. For A7, both $\delta \cdot x$ and δ initially can perform but a single δ-transition. Since $\leftrightarrows_\delta$ matches one δ-transition with any other δ-transition (i.e., without regard to the destination states), we are done. The soundness of D1 and D2 is trivial since in both cases the left- and right-hand side terms represent isomorphic processes.

For D3.1, the initial transitions of x will be deleted from the root of $x + y$ by the ∂_H operation, thereby again resulting in isomorphic processes. D3.2 could fail only if $x, y \neq \delta$ and either $\partial_H(x) = \delta$ or $\partial_H(y) = \delta$. The condition to the axiom ensures against this. Note that D3.2 is still sound under the weaker condition

$$I(x) - H_\delta \neq \emptyset \text{ and } I(y) - H_\delta \neq \emptyset$$

but the natural probabilistic extension of the resulting axiom is not sound (see Section 3.4), and is thus rejected. Finally, D4 also represents isomorphic processes.

For part 2, suppose $p \leftrightarrows_\delta q$. Reduce p, q to normal forms p', q' using $\mathrm{RACP}_I^-(A)$; by Lemma 2.2, p', q' are basic terms. By part 1, $p' \leftrightarrows_\delta p \leftrightarrows_\delta q \leftrightarrows_\delta q'$, and thus $p' \leftrightarrows_\delta q'$. In reducing p, q to their normal forms, we have been rewriting by A7 whenever possible. We may therefore conclude that $p' \leftrightarrows q'$ (treating δ as just another atomic action), and by Theorem 2.1, $\mathrm{BPA}(A_\delta) \vdash p' = q'$. Then $\mathrm{ACP}_I^-(A) \vdash p = p' = q' = q$. □

2.3.3 Connections Between ACP and ACP_I^-

Let $\mathbf{A}$ be the usual bisimulation model for ACP(A), and let $\mathbf{A}^- = \mathcal{F}/\leftrightarrows_\delta$ be the bisimulation model for $\mathrm{ACP}_I^-(A)$. Then for p,q closed expressions over $\Sigma(\mathrm{ACP}(A))$ we have the following results, which we state without proof.

1. *Completeness of* $\mathrm{ACP}_I^-(A)$: $\mathbf{A}^- \models p = q \implies \mathrm{ACP}_I^-(A) \vdash p = q$
   (This is just part 2 of Theorem 2.2.)

2. *Completeness of* ACP*(A)* [BW90]: $\mathbf{A} \models p = q \implies \mathrm{ACP}(A) \vdash p = q$

3. $\mathbf{A}^- \models p = q \implies \mathbf{A} \models p = q$. This implies that $\mathbf{A}^-$ can be homomorphically embedded in $\mathbf{A}$ using the identity mapping.

4. $\mathbf{A} \models p = q \implies \mathbf{A}^- \models \partial_\emptyset(p) = \partial_\emptyset(q)$. This implies that $\mathbf{A}$ can be homomorphically embedded in $\mathbf{A}^-$ using the homomorphism $\varphi : \mathbf{A} \longrightarrow \mathbf{A}^-$, such that $\varphi(x) = \partial_\emptyset(x)$.

5. $\mathrm{ACP}(A) \vdash \partial_\emptyset(p) = p$

6. $\mathrm{ACP}(A) \vdash p = q \implies \mathrm{ACP}_I^-(A) + \{x + \delta = x\} \vdash p = q$

7. $\mathrm{ACP}_I^-(A) \vdash \partial_\emptyset(x + \delta) = \partial_\emptyset(x)$

3 A Probabilistic Version of ACP

Our discussion of probabilistic ACP will proceed in a manner similar to before. For each of the axiom systems $AX \in \{\mathrm{BPA}(A), \mathrm{PA}(A), \mathrm{ACP}_I^-(A)\}$, a probabilistic version $prAX$ will be introduced, along with a probabilistic version of its process graph model. Completeness in these models will also be demonstrated.

3.1 Probabilistic BPA

3.1.1 Equational Specification

Notation: As usual, $(0,1)$ denotes the open interval of the real line $\{r \in \Re \mid 0 < r < 1\}$, and $[0,1]$ denotes the closed interval of the real line $\{r \in \Re \mid 0 \leq r \leq 1\}$. We let π, ρ, σ, and θ, possibly subscripted, range over these intervals.

The signature $\Sigma(pr\mathrm{BPA}(A))$ over the sort $pr\mathbf{P}$ (for probabilistic processes) is given by:

$$\begin{aligned}\Sigma(pr\mathrm{BPA}(A)) \quad = \quad & \{a : \rightarrow pr\mathbf{P} | a \in A\} \cup \{+_\pi : pr\mathbf{P} \times pr\mathbf{P} \rightarrow pr\mathbf{P} \mid \pi \in (0,1)\} \cup \\ & \{\cdot : pr\mathbf{P} \times pr\mathbf{P} \rightarrow pr\mathbf{P}\}\end{aligned}$$

The operator $+$ has been replaced by the family of operators $+_\pi$, for each probability π in the interval $(0,1)$, and is now called ***probabilistic alternative composition.*** Intuitively, the expression $x +_\pi y$ behaves like x with probability π and like y with probability $1 - \pi$. Probabilistic alternative composition is *generative* [vGSST90] in that a single distribution (viz. the discrete probability distribution $\{p, 1-p\}$) is associated with the two alternatives x and y. As mentioned in Section 1, these probabilities are conditional with respect to the set of actions permitted by the environment. This will become clear in Section 3.4 with the introduction of the restriction operator ∂_H in the setting of probabilistic ACP.

We have the following axioms for $pr\mathrm{BPA}(A)$:

$x +_\pi y = y +_{1-\pi} x$	*pr*A1
$x +_\pi (y +_\rho z) = (x +_{\pi/(\pi+\rho-\pi\rho)} y) +_{\pi+\rho-\pi\rho} z$	*pr*A2
$x +_\pi x = x$	*pr*A3
$(x +_\pi y) \cdot z = x \cdot z +_\pi y \cdot z$	*pr*A4
$(x \cdot y) \cdot z = x \cdot (y \cdot z)$	*pr*A5

Axiom prA2 has a left-to-right orientation in that the probability indices on the right-hand side are derived from probability indices π,ρ on the left-hand side. A right-to-left version of prA2, which will prove useful later, is given by:

$$(x +_\pi y) +_\rho z = x +_{\pi\rho} (y +_{\frac{(1-\pi)\rho}{(1-\pi\rho)}} z)$$

3.2 Probabilistic Graph Model

As in Section 2.1.2, we consider process graphs, with labels from A, as a model for prBPA(A). Additionally, a probability distribution will be ascribed to each node's outgoing transitions.

Definition 3.1 *A* probabilistic process graph g *is a triple* $<V, r, \mu>$ *such that* V *and* r *are as in Definition 2.1 and* $\mu : (V \times A \times V) \to [0,1]$, *the* transition distribution function *of* g, *is a total function satisfying the following* stochasticity condition*:*

$$\forall v \in V \quad \sum_{\substack{a \in A, \\ v' \in V}} \mu(v, a, v') \ \in \ \{0, 1\}$$

Intuitively, $\mu(v, a, v') = \pi$ means that, with probability π, node v can perform an a-transition to node v'. A node in a stochastic probabilistic process graph performs some transition with probability 1, unless it is an endpoint. Predicate *endpoint*(v) is true iff v is an endpoint. We denote by $pr\mathcal{G}$ the family of all probabilistic process graphs.

The notion of strong bisimulation for nondeterministic processes has been extended by Larsen and Skou [LS89] to reactive probabilistic processes in the form of *probabilistic bisimulation*. Here we define probabilistic bisimulation on generative probabilistic processes and to do so we first need to lift the definition of the transition distribution function as follows:

$$\mu : (V \times A \times 2^V) \longrightarrow [0,1] \text{ such that } \mu(v, a, S) = \sum_{v' \in S} \mu(v, a, v')$$

Intuitively, $\mu(v, a, S) = \rho$ means that node v, with total probability ρ, can perform an a-transition to some node in S.

Definition 3.2 **([LS89])** *Let* $g_1 = <V_1, r_1, \mu_1>$, $g_2 = <V_2, r_2, \mu_2>$ *be probabilistic process graphs. A* probabilistic bisimulation *between* g_1 *and* g_2 *is an equivalence relation* $\mathcal{R} \subseteq (V_1 \cup V_2) \times (V_1 \cup V_2)$ *with the following properties:*

- $\mathcal{R}(r_1, r_2)$
- $\forall v \in V_1,\ w \in V_2$ *such that* $\mathcal{R}(v, w)$*:*
  $\forall a \in A,\ S \in (V_1 \cup V_2)/\mathcal{R},\quad \mu_1(v, a, S \cap V_1) = \mu_2(w, a, S \cap V_2)$

Graphs g_1 *and* g_2 *are* probabilistically bisimilar, *written* $g_1 \leftrightarrows^{pr} g_2$, *if there exists a probabilistic bisimulation between* g_1 *and* g_2.

Intuitively, two nodes are probabilistically bisimilar if, for all actions in A, they transit to probabilistic bisimulation classes with equal probability. Note the somewhat subtle use of recursion in the definition.

We now define the operators of prBPA(A) on the domain $pr\mathcal{F}$ of *finite* probabilistic process graphs, i.e., probabilistic process graphs that are finitely branching and acyclic in terms of their transitions of non-zero probability. Therefore, $pr\mathcal{F} \subset pr\mathcal{G}$. For this purpose, it is convenient to assume, as in the non-probabilistic case, that the root nodes of probabilistic process graphs are not endpoints. For the remainder of Section 3, unless otherwise stated, let $g_1 = <V_1, r_1, \mu_1>$, $g_2 = <V_2, r_2, \mu_2>$ be finite probabilistic process graphs satisfying the non-endpoint root assumption such that $V_1 \cap V_2 = \emptyset$.

Definition 3.3 *The operators $a \in A$, $+_\pi$, and $\cdot$ are defined on $pr\mathcal{F}$ as follows:*

$a \in A$: *The process graph for each of these constants is given by $<\{r_a, v\}, r_a, \mu_a>$, where $\mu_a(r_a, a, v) = 1$ is the only transition with non-zero probability.*

$g_1 +_\pi g_2$: *is given by $<V_1 \cup V_2 \cup \{r\} - \{r_1, r_2\}, r, \mu>$ where $r \notin V_1 \cup V_2$ and*

$$\begin{array}{ll} \mu(r,a,v') = \pi \cdot \mu_1(r_1,a,v') & \textit{if } v' \in V_1 \\ \mu(r,a,v') = (1-\pi)\cdot \mu_2(r_2,a,v') & \textit{if } v' \in V_2 \\ \mu(v,a,v') = \mu_1(v,a,v') & \textit{if } v,v' \in V_1 \\ \mu(v,a,v') = \mu_2(v,a,v') & \textit{if } v,v' \in V_2 \\ \mu(v,a,v') = 0 & \textit{otherwise} \end{array}$$

$g_1 \cdot g_2$: *is obtained by appending a copy of g_2 at each endpoint of g_1, and is analogous to sequential composition in the non-probabilistic setting (Definition 2.3). In detail, $g_1 \cdot g_2$ is given by $<V_1 \cup V_2 - \{r_2\}, r_1, \mu>$ where*

$$\mu(v,a,v') = \begin{cases} \mu_1(v,a,v') & \textit{if } v,v' \in V_1 \\ \mu_2(r_2,a,v') & \textit{if } v \in V_1,\ \textit{endpoint}(v),\ v' \in V_2 \\ \mu_2(v,a,v') & \textit{if } v,v' \in V_2 \\ 0 & \textit{otherwise} \end{cases}$$

So, in the definition of $g_1 +_\pi g_2$, the transitions from r_1,r_2 are now assumed by the new root r, with their probability of occurrence weighted appropriately. Similarly, the transitions of r_2 in $g_1 \cdot g_2$ are assumed by each endpoint of g_1, with their original probabilities intact.

As in the non-probabilistic case, for t a closed prBPA(A) term, we write $graph(t) = <V_t, r_t, \mu_t>$ to denote the probabilistic process graph obtained inductively on t using Definition 3.3. We also write $p \leftrightarrows^{pr} q$ as shorthand for $graph(p) \leftrightarrows^{pr} graph(q)$. The definition of $graph(t)$ and the just-mentioned notational shorthand extend in the obvious way to the axiom systems prPA(A) and prACP$_I^-$(A) considered later in this section.

We will subsequently prove that the axioms of prBPA(A) are complete in this model. To admit sound equational reasoning, in particular, the substitution of equals for equals, we first show that $\leftrightarrows^{pr}$ is a congruence in prBPA(A). Let V be an arbitrary set with $v \in V$. For any equivalence relation $\mathcal{R}$ over V we use $[v]_\mathcal{R}$ to denote the set $\{w \in V \mid (v,w) \in \mathcal{R}\}$; i.e., $[v]_\mathcal{R}$ is the equivalence class of v induced by $\mathcal{R}$. Also, $Id_V = \{(v,v) \mid v \in V\}$ denotes the identity relation on V.

Proposition 3.1 *If $g_1 \leftrightarrows^{pr} g_2$, then $g +_\pi g_1 \leftrightarrows^{pr} g +_\pi g_2$, $g \cdot g_1 \leftrightarrows^{pr} g \cdot g_2$, and $g_1 \cdot g \leftrightarrows^{pr} g_2 \cdot g$.*

Proof: Let $g = <V, r, \mu>$ such that $V \cap (V_1 \cup V_2) = \emptyset$, assume $g_1 \leftrightarrows^{pr} g_2$, and let $\mathcal{R}$ be a probabilistic bisimulation between g_1 and g_2. We now consider each of the operators in succession.

For $+_\pi$, let r_i^+ be the root and μ_i^+ be the tdf of $g +_\pi g_i$, $i = 1,2$. We show that

$$\mathcal{R}' = \{(r_1^+, r_2^+), (r_2^+, r_1^+)\} \cup \mathcal{R} \cup Id_{V \cup \{r_1^+, r_2^+\}}$$

is a probabilistic bisimulation between $g +_\pi g_1$ and $g +_\pi g_2$. First note that because $\mathcal{R}$ is an equivalence relation, so is $\mathcal{R}'$. By the nature of $\mathcal{R}'$, we are left to show that the "carrier condition" (the second condition of Definition 3.3) holds for (r_1^+, r_2^+). For $a \in A$, the only a-transitions of r_1^+ of non-zero probability are of the form:

1. $\mu_1^+(r_1^+, a, [v']_{\mathcal{R}'}) = \mu(r, a, v') \cdot \pi$, where $v' \in V$; or
2. $\mu_1^+(r_1^+, a, [v_1']_{\mathcal{R}'}) = \mu_1(r_1, a, [v_1']_{\mathcal{R}}) \cdot (1 - \pi)$, where $v_1' \in V_1$.

Well, we also have $\mu_2^+(r_2^+, a, [v']_{\mathcal{R}'}) = \mu(r, a, v') \cdot \pi$ and, because $g_1 \leftrightarrows^{pr} g_2$, $\mu_2^+(r_2^+, a, [v_1']_{\mathcal{R}'}) = \mu_1(r_1, a, [v_1']_{\mathcal{R}}) \cdot (1 - \pi)$. This completes the case for $+_\pi$.

For both cases of sequential composition, a straightforward argument demonstrates that $\mathcal{R} \cup Id_V$ is an appropriate probabilistic bisimulation. □

The graph model for prBPA(A) is now given by $pr\mathcal{F}/\leftrightarrows^{pr}$. To prove completeness of prBPA(A), we introduce the notation

$$\sum_{i=1}^{n} [\pi_i] x_i$$

with $\sum \pi_i = 1$ and $\pi_i > 0$ for all i. So, in particular, when $n = 1$, $\pi_1 = 1$. This notation abbreviates right-nested probabilistic alternative composition expressions as follows:

$$\sum_{i=1}^{1} [\pi_i] x_i = x_1 \qquad \text{and} \qquad \sum_{i=1}^{n+1} [\pi_i] x_i = x_1 +_{\pi_1} \left(\sum_{i=1}^{n} \left[\frac{\pi_{i+1}}{1 - \pi_1} \right] x_{i+1} \right)$$

Note that in this notation $\sum_{i=1}^{n} [\pi_i]$ is a derived n-ary operator with operands x_i. To illustrate, the left-hand side of equation prA2 may be written:

$$\sum_{i=1}^{3} [\pi_i] x_i$$

where $\pi_1 = \pi$, $\pi_2 = (1 - \pi)\rho$, $\pi_3 = (1 - \pi)(1 - \rho)$, and $x_1 = x$, $x_2 = y$, $x_3 = z$.

This *summation form* notation is useful as it directly reflects the transition structure of the probabilistic process graph underlying the nested probabilistic alternative composition. That is, consider the summation form $\sum [\pi_i] a_i \cdot x_i$ of action-prefixed processes. The graph of this summation will have, for each i, a probability-π_i a_i-transition from its root to the node representing the root of $graph(x_i)$.

The following two lemmas for manipulating summation forms, the proofs of which appear in Appendix A, will prove useful in the completeness proof for prBPA(A). The first allows summands to be reordered arbitrarily, retaining their original probabilities, while the second allows two syntactically identical summands to be merged into one summand, summing the probabilities in the process.

Lemma 3.1 *For any permutation ξ of $\{1, \cdots, n\}$, $n \geq 2$,*

$$pr\mathrm{BPA}(A) \vdash \sum_{i=1}^{n} [\pi_i] x_i = \sum_{i=1}^{n} [\pi_{\xi(i)}] x_{\xi(i)}$$

Lemma 3.2 *In the summation form $\sum_{i=1}^{n+1}[\pi_i]x_i$, let x_1 and x_2 be syntactically identical. Then*

$$pr\mathrm{BPA}(A) \vdash \sum_{i=1}^{n+1}[\pi_i]x_i = \sum_{i=1}^{n}[\rho_i]y_i$$

where $\rho_1 = \pi_1 + \pi_2$, $y_1 = x_1$, and $\rho_i = \pi_{i+1}$, $y_i = x_{i+1}$, $2 \leq i \leq n$.

We now use summation-form notation to define a kind of normal form for closed prBPA(A) terms.

Definition 3.4 *A* probabilistic basic term *is a summation form $\sum_{i=1}^{n}[\pi_i]t_i$ where t_i is either some $a \in A$ or of the form $b \cdot t_i'$, where $b \in A$ and t_i' is a probabilistic basic term. A* probabilistic normal form *is a probabilistic basic term $\sum_{i=1}^{n}[\pi_i]t_i$ such that $t_i \not\leftrightarrow^{pr} t_j$, $1 \leq i \neq j \leq n$.*

Note that a probabilistic basic term, like a basic $\mathrm{ACP}_I^-(A)$ term of Section 2.3, uses action prefixing, while a probabilistic normal form bears the additional constraint that its summands are pairwise inequivalent.

The *depth* of a probabilistic basic term t, denoted $d(t)$, is essentially the maximum number of nested prefixes in t. The inductive definition of d is as follows:

- $d(a) = 1$
- $d(a \cdot t) = 1 + d(t)$
- $d(\sum_i[\pi_i]t_i) = max_i(d(t_i))$

Lemma 3.3 *For every closed prBPA(A) term t, there is a probabilistic normal form s such that $pr\mathrm{BPA}(A) \vdash t = s$.*

Proof: The proof has two parts. In the first part, we prove that a closed term t can be proven equal to a probabilistic basic term. The second part handles the constraint that the summands are pairwise inequivalent. The first part is simpler and follows the line of reasoning in [BW90]. That is, we use a term rewriting system to convert t into a term whose only instances of sequential composition are of the form $a \cdot t'$, i.e., action prefixing. The rewrite system is based on prBPA(A) axioms prA4 and prA5 and is given by:

$$(x -_\pi y) \cdot z \longrightarrow x \cdot z -_\pi y \cdot z$$
$$(x \cdot y) \cdot z \longrightarrow x \cdot (y \cdot z)$$

It is not hard to see that this term rewrite system is confluent and strongly normalizing, and that a normal form of a closed term uses only action prefixing. Therefore, given a closed prBPA(A) term t, we can convert it into a probabilistic basic term by:

1. Reduce t until a normal form is reached.
2. Use prA3, with right-to-left orientation, to rewrite all instances of left-nested summations into right-nested summations. The resulting term can then be expressed as a summation form.

By the first part of the proof, assume t is a probabilistic basic term of the form $\sum_{i=1}^{n}[\pi_i]t_i$ and consider the partition $\{B_1,\ldots,B_k\}$ of $\{1,\ldots,n\}$ such that $(i,i') \in B_j$ if $t_i \underline{\leftrightarrow}^{pr} t_{i'}$. We prove by induction on the depth of t that:

$$prBPA(A) \vdash t = \sum_{j=1}^{m}[\rho_j]t'_j$$

where $\rho_j = \sum\{\pi_i \mid i \in B_j\}$, $t'_j = t_i$ for an arbitrarily chosen $i \in B_j$, and $m \leq n$. Note that the term on the right-hand-side of this equation is indeed a probabilistic normal form. If the depth of t is 1 then each t_i is a constant and the indices in a block B_j correspond to (all of the) multiple occurrences of a constant a. If $|B_j| = 1$ then we are done. Otherwise, apply the following procedure $|B_j| - 1$ times: move two instances of a to the two left-most positions within the summation form using Lemma 3.1. Merge the two instances into one, occupying the left-most position in the resulting summation form, using Lemma 3.2. The associated probability of this single instance of a will be the sum of the probabilities of the original two instances, as desired.

Next, assume the result for probabilistic basic terms of depth k and let $d(t) = k+1$. There are two cases.

1. The indices in a block B_j correspond to the multiple occurrences of a constant a. The base case reasoning suffices here.

2. The indices in a block B_j correspond to equivalent terms of the form $a \cdot t'$, $b \cdot t''$, where t', t'' are basic. If $|B_j| = 1$ then we are done. Otherwise, apply the following procedure $|B_j| - 1$ times. Choose two instances $a \cdot t'$, $b \cdot t''$ of equivalent terms from B_j. Since $a \cdot t' \underline{\leftrightarrow}^{pr} b \cdot t''$, then $a = b$ and $t' \underline{\leftrightarrow}^{pr} t''$, and, by the induction hypothesis, $prBPA(A) \vdash t' = t''$. By substitution of equals for equals, we have $prBPA(A) \vdash a \cdot t' = b \cdot t''$, and, as in the first case, we can use Lemmas 3.1 and 3.2 to merge these two summands into a single summand, either $a \cdot t'$ or $b \cdot t''$, the choice being arbitrary. The associated probability of the merged term will be the sum of the associated probabilities of $a \cdot t'$ and $b \cdot t''$, as desired.

$\square$

The relationship observed above between a probabilistic summation form and its underlying probabilistic process graph can be strengthened in the case of probabilistic normal forms.

Proposition 3.2 *For t a probabilistic normal form, t has a summand $a \in A$, with associated probability π, iff $\mu_t(r_t, a, [v]_{\underline{\leftrightarrow}^{pr}}) = \pi$ where v is an endpoint. Also, t has a summand $a \cdot t'$, with associated probability π, iff $\mu_t(r_t, a, [r^t_{t'}]_{\underline{\leftrightarrow}^{pr}}) = \pi$ where $r^t_{t'} \in V_t$ is the node in $graph(t)$ representing the root of $graph(t')$.*

We now prove that our algebra $pr\mathcal{F}/\underline{\leftrightarrow}^{pr}$ is a model of $prBPA(A)$ and that $prBPA(A)$ constitutes a complete axiomatization of process equivalence in $pr\mathcal{F}/\underline{\leftrightarrow}$.

Theorem 3.1

1. $pr\mathcal{F}/\underline{\leftrightarrow}^{pr} \models prBPA(A)$

2. For all closed expressions s,t over $\Sigma(prBPA(A))$:

$$pr\mathcal{F}/\underline{\leftrightarrow}^{pr} \models s = t \implies prBPA(A) \vdash s = t.$$

Proof: For part 1, consider first prA1 and prA2. In both cases the left- and right-hand side terms represent isomorphic probabilistic process graphs, with the transitions from the root of x weighted by π and the transitions from the root of y weighted by $1-\pi$, in the case of prA1; and the root transitions of x weighted by π, the root transitions of y rooted by $(1-\pi)\rho$, and the root transitions of z weighted by $(1-\pi)(1-\rho)$, in the case of prA2.

Graph isomorphism arguments also suffice for prA4 and prA5, while the soundness of prA3 is established by the probabilistic bisimulation $\{(r_{x+_\pi x}, r_x), (r_x, r_{x+_\pi x})\} \cup Id_{V_x \cup \{r_{x+_\pi x}\}}$.

For part 2, assume $s \underline{\leftrightarrow}^{pr} t$ and also (relying on Lemma 3.3) that s and t are probabilistic normal forms, $s = \sum_i [\pi_i] s_i$ and $t = \sum_j [\rho_j] t_j$. We prove the result by induction on the maximum depth of s and t. If the maximum depth is 1 then each summand of s is a constant from A. Let $s_i = a$. Since $s \underline{\leftrightarrow}^{pr} t$ and t is a probabilistic normal form, by Proposition 3.2, t also has a summand $t_j = a$ with $\rho_j = \pi_i$. A symmetric argument matches each constant summand of t with a summand of s. Thus, $pr\mathrm{BPA}(A) \vdash s = t$ by using Lemma 3.1 to reorder summands as necessary.

Next, assume the result for maximum depth k and let the maximum depth of s,t be $k+1$. There are two cases.

1. The term s has a constant summand. Here the base case reasoning suffices.

2. The term s has a summand s_i of the form $a \cdot s'$ and, by Proposition 3.2, $\mu_s(r_s, a, [r^s_{s'}]_{\underline{\leftrightarrow}^{pr}}) = \pi_i$. Since $s \underline{\leftrightarrow}^{pr} t$, $\mu_t(r_t, a, [r^t_{s'}]_{\underline{\leftrightarrow}^{pr}}) = \pi_i$. But t is a probabilistic normal form so, by Proposition 3.2 again, t has a summand $t_j = a \cdot t'$ such that $t' \underline{\leftrightarrow}^{pr} s'$ and $\rho_j = \pi_i$. By induction, $pr\mathrm{BPA}(A) \vdash s' = t'$ and therefore (using substitution of equals for equals), $pr\mathrm{BPA}(A) \vdash s_i = t_j$. A symmetric argument matches each action-prefixed summand of t with a summand of s.

From the two cases, it follows that every summand of s can be proved equal to a summand of t and vice versa. Thus, $pr\mathrm{BPA}(A) \vdash s = t$, by using Lemma 3.1 to reorder summands as necessary.
□

We also prove the following proposition:

Proposition 3.3 *The various forms of $+_\pi$ distribute over one another:*

$$(x +_\pi y) +_\rho z = (x +_\rho z) +_\pi (y +_\rho z)$$

Proof:

$$\begin{aligned}
(x +_\rho z) +_\pi (y +_\rho z) &= x +_{\rho\pi} (z +_{\frac{(1-\rho)\pi}{1-\rho\pi}} (y +_\rho z)) && (pr\mathrm{A2})\\
&= x +_{\rho\pi} (z +_{\frac{(1-\rho)\pi}{1-\rho\pi}} (z +_{1-\rho} y)) && (pr\mathrm{A1})\\
&= x +_{\rho\pi} ((z +_\pi z) +_{\frac{1-\rho}{1-\rho\pi}} y) && (pr\mathrm{A2})\\
&= x +_{\rho\pi} (z +_{\frac{1-\rho}{1-\rho\pi}} y) && (pr\mathrm{A3})\\
&= x +_{\rho\pi} (y +_{\frac{\rho(1-\pi)}{1-\rho\pi}} z) && (pr\mathrm{A1})\\
&= (x +_\pi y) +_\rho z && (pr\mathrm{A2})
\end{aligned}$$

□

Note that the last step makes direct use of the right-to-left oriented version of prA2.

3.3 Probabilistic PA

3.3.1 Equational Specification

The signature $\Sigma(pr\mathrm{PA}(A))$ extends that of $pr\mathrm{BPA}(A)$.

$$\begin{aligned}\Sigma(pr\mathrm{PA}(A)) &= \Sigma(pr\mathrm{BPA}(A)) \cup \{\|_\sigma: pr\mathbf{P} \times pr\mathbf{P} \to pr\mathbf{P} \mid \sigma \in (0,1)\} \cup \\ & \{\lfloor\!\lfloor_\sigma: pr\mathbf{P} \times pr\mathbf{P} \to pr\mathbf{P} \mid \sigma \in (0,1)\}\end{aligned}$$

Intuitively, $\|_\sigma$ is a *probabilistic merge* operator, with the left operand receiving relative probability σ and the right operand relative probability $1-\sigma$. As in PA(A), $\lfloor\!\lfloor_\sigma$ is a restricted version of $\|_\sigma$ in which the first step must come from the left operand.

The axiom system $pr\mathrm{PA}(A)$ is obtained by adding to $pr\mathrm{BPA}(A)$ the following axioms for probabilistic merge and left-merge:

$x \|_\sigma y = x \lfloor\!\lfloor_\sigma y +_\sigma y \lfloor\!\lfloor_{(1-\sigma)} x$	prM1
$a \lfloor\!\lfloor_\sigma y = a \cdot y$	prM2
$(a \cdot x) \lfloor\!\lfloor_\sigma y = a \cdot (x \|_\sigma y)$	prM3
$(x +_\pi y) \lfloor\!\lfloor_\sigma z = (x \lfloor\!\lfloor_\sigma z) +_\pi (y \lfloor\!\lfloor_\sigma z)$	prM4

3.3.2 Graph Model

As for $pr\mathrm{BPA}(A)$, we provide a bisimulation model for $pr\mathrm{PA}(A)$, and prove the completeness of the axioms on finite probabilistic processes.

Definition 3.5 *The operators $\|_\sigma$ and $\lfloor\!\lfloor_\sigma$ are defined on $pr\mathcal{F}$ as follows:*

$g_1 \|_\sigma g_2$: *is given by* $< V_1 \times V_2, (r_1, r_2), \mu >$ *where for all* $a \in A$, $v_1, v_1' \in V_1$, $v_2, v_2' \in V_2$

$$\mu((v_1,v_2),a,(v_1',v_2)) = \begin{cases} \sigma \cdot \mu_1(v_1,a,v_1') & \textit{if } \neg \textit{endpoint}(v_2) \\ \mu_1(v_1,a,v_1') & \textit{otherwise} \end{cases}$$

$$\mu((v_1,v_2),a,(v_1,v_2')) = \begin{cases} (1-\sigma) \cdot \mu_2(v_2,a,v_2') & \textit{if } \neg \textit{endpoint}(v_1) \\ \mu_2(v_2,a,v_2') & \textit{otherwise} \end{cases}$$

$g_1 \lfloor\!\lfloor_\sigma g_2$: *is given by* $< V_1 \times V_2, (r_1, r_2), \mu >$ *where for all* $a \in A$, $v_1, v_1' \in V_1$, $v_2, v_2' \in V_2$

- $\mu((r_1,r_2),a,(v_1',r_2)) = \mu_1(r_1,a,v_1')$
- *if* $v_1 \neq r_1$ *or* $v_2 \neq r_2$

$$\mu((v_1,v_2),a,(v_1',v_2)) = \begin{cases} \sigma \cdot \mu_1(v_1,a,v_1') & \textit{if } \neg \textit{endpoint}(v_2) \\ \mu_1(v_1,a,v_1') & \textit{otherwise} \end{cases}$$

$$\mu((v_1,v_2),a,(v_1,v_2')) = \begin{cases} (1-\sigma) \cdot \mu_2(v_2,a,v_2') & \textit{if } \neg \textit{endpoint}(v_1) \\ \mu_2(v_2,a,v_2') & \textit{otherwise} \end{cases}$$

- *if* $v_2' \neq r_2$ $\quad \mu((r_1,r_2),a,(v_1',v_2')) = 0$

Note the careful treatment of endpoints in the above definition: in a merge, if one process terminates, the other continues with its original, unweighted probability. Also, in a left-merge, special attention is paid to transitions from the root (r_1, r_2) of $g_1 \lfloor\!\lfloor_\sigma g_2$: the first and third clauses collectively define the transition distribution function μ on all transitions from (r_1, r_2), with the third clause giving probability 0 to transitions starting with g_2.

We have that probabilistic bisimulation is a congruence in $pr\mathrm{PA}(A)$.

Proposition 3.4 *If $g_1 \leftrightarrows^{pr} g_2$, then $g \parallel_\sigma g_1 \leftrightarrows^{pr} g \parallel_\sigma g_2$, $g \lfloor\!\lfloor_\sigma g_1 \leftrightarrows^{pr} g \lfloor\!\lfloor_\sigma g_2$, and $g_1 \lfloor\!\lfloor_\sigma g \leftrightarrows^{pr} g_2 \lfloor\!\lfloor_\sigma g$.*

Proof: Let $g = <V, r, \mu>$ such that $V \cap (V_1 \cup V_2) = \emptyset$, assume $g_1 \leftrightarrows^{pr} g_2$, and let $\mathcal{R}$ be a probabilistic bisimulation between g_1 and g_2. We first show that

$$\mathcal{R}' = \{((v, v_1), (v, v_2)) \mid v \in V, (v_1, v_2) \in \mathcal{R}\}$$

is a probabilistic bisimulation between $g \parallel_\sigma g_1$ and $g \parallel_\sigma g_2$. First note that because $\mathcal{R}$ is an equivalence relation, so is $\mathcal{R}'$. Also, for $v \in V$, $w \in (V_1 \cup V_2)$, $[(v, w)]_{\mathcal{R}'} = \{v\} \times [w]_{\mathcal{R}}$. Now consider the pair $((v, v_1), (v, v_2)) \in \mathcal{R}'$ and let $\mu_i^{\parallel}$ be the tdf of $g \parallel_\sigma g_i$, $i = 1, 2$. For $a \in A$, the only a-transitions of (v, v_1) of non-zero probability are of the form:

1. $\mu_1^{\parallel}((v, v_1), a, [(v', v_1)]_{\mathcal{R}'}) = \sigma \cdot \mu(v, a, v')$
2. $\mu_1^{\parallel}((v, v_1), a, [(v, v_1')]_{\mathcal{R}'}) = (1 - \sigma) \cdot \mu_1(v_1, a, [v_1']_{\mathcal{R}})$

Well, we also have that $\mu_2^{\parallel}((v, v_2), a, [(v', v_2)]_{\mathcal{R}'}) = \sigma \cdot \mu(v, a, v')$ and, because $g_1 \leftrightarrows^{pr} g_2$, $\mu_2^{\parallel}((v, v_2), a, [(v, v_1')]_{\mathcal{R}'}) = (1 - \sigma) \cdot \mu_1(v_1, a, [v_1']_{\mathcal{R}})$. The argument is similar in case (1) if v_1 is an endpoint (the value of $\mu_1^{\parallel}$ would not be weighted by σ), and in case (2) if v is an endpoint (the value of $\mu_1^{\parallel}$ would not be weighted by $1 - \sigma$).

An argument similar to the above can be used to show that $\mathcal{R}'$ is also a probabilistic bisimulation between $g \lfloor\!\lfloor_\sigma g_1$ and $g \lfloor\!\lfloor_\sigma g_2$. In particular, there are fewer transitions of non-zero probability from (r, r_1) and (r, r_2) since such transitions can come from g only. Like in the endpoint cases considered just above, the probabilities of these transitions are not weighted by σ.

A nearly symmetric argument establishes that

$$\mathcal{R}'' = \{((v_1, v), (v_2, v)) \mid (v_1, v_2) \in \mathcal{R}, v \in V\}$$

is a probabilistic bisimulation between $g_1 \lfloor\!\lfloor_\sigma g$ and $g_2 \lfloor\!\lfloor_\sigma g$. □

Theorem 3.2

1. *$pr\mathcal{F}/\leftrightarrows^{pr} \models pr\mathrm{PA}(A)$*
2. *For all closed expressions s,t over $\Sigma(pr\mathrm{PA}(A))$:*

$$pr\mathcal{F}/\leftrightarrows^{pr} \models s = t \implies pr\mathrm{PA}(A) \vdash s = t.$$

Proof: For part 1, the soundness of axioms prM1 – prM4 is immediate by probabilistic process graph isomorphism arguments. The following comments, however, are in order. Axiom prM1 is a kind of expansion law for probabilistic merge. In prM2, $a \lfloor\!\lfloor_\sigma y$ behaves like y after performing a as it will have reached a state where y is in a probabilistic merge with an endpoint. In prM3, $(a \cdot x) \lfloor\!\lfloor_\sigma y$ behaves like $x \parallel_\sigma y$ after performing a since left-merge behaves like merge after its root transitions. The left-hand and right-hand side processes of prM4 both represent a probabilistic merge with z, the first step of which must come from x (with probability π) or y (with probability $1-\pi$).

For part 2, the proof is similar to the one given in [BW90] for the completeness of PA(A). We use the following term rewrite system, with rules corresponding to prBPA(A) axioms prA3-5 and prPA(A) axioms prM1 – prM4, to eliminate all occurrences of $\|_\sigma$ and $\lfloor\!\lfloor_\sigma$ in a closed prPA(A) term:

$$
\begin{aligned}
& x +_\pi x \longrightarrow x \\
& (x +_\pi y) \cdot z \longrightarrow x \cdot z +_\pi y \cdot z \\
& (x \cdot y) \cdot z \longrightarrow x \cdot (y \cdot z) \\
& x \,\|_\sigma\, y \longrightarrow x \,\lfloor\!\lfloor_\sigma\, y \;+_\sigma\; y \,\lfloor\!\lfloor_{(1-\sigma)}\, x \\
& a \,\lfloor\!\lfloor_\sigma\, y \longrightarrow a \cdot y \\
& a \cdot x \,\lfloor\!\lfloor_\sigma\, y \longrightarrow a \cdot (x \,\|_\sigma\, y) \\
& (x +_\pi y) \,\lfloor\!\lfloor_\sigma\, z \longrightarrow (x \,\lfloor\!\lfloor_\sigma\, z) +_\pi (y \,\lfloor\!\lfloor_\sigma\, z)
\end{aligned}
$$

It can be proved that this term rewriting system is strongly normalizing and that a normal form of a closed term must be a probabilistic basic term. By part 1 of the theorem (the soundness of prPA(A)) and Theorem 3.1 (the soundness and completeness of prBPA(A)), the result is proven. □

3.4 Probabilistic ACP

3.4.1 Equational Specification

The signature of prACP$_I^-(A)$ also extends that of prBPA(A). Recalling that $A_\delta = A \cup \delta$, we have:

$$
\begin{aligned}
\Sigma(pr\text{ACP}_I^-(A)) = {} & \Sigma(pr\text{BPA}(A)) \;\cup\; \{\delta :\rightarrow pr\mathbf{P}\} \;\cup\; \{I : pr\mathbf{P} \rightarrow 2^{A_\delta}\} \;\cup \\
& \{|_{\sigma,\theta} : pr\mathbf{P} \times pr\mathbf{P} \rightarrow pr\mathbf{P} \mid \sigma,\theta \in (0,1)\} \;\cup\; \{\|_{\sigma,\theta} : pr\mathbf{P} \times pr\mathbf{P} \rightarrow pr\mathbf{P} \mid \sigma,\theta \in (0,1)\} \;\cup \\
& \{\lfloor\!\lfloor_{\sigma,\theta} : pr\mathbf{P} \times pr\mathbf{P} \rightarrow pr\mathbf{P} \mid \sigma,\theta \in (0,1)\} \;\cup\; \{\partial_H : pr\mathbf{P} \rightarrow pr\mathbf{P} \mid H \subseteq A\}
\end{aligned}
$$

Thus, for each of the operators $|$, $\|$, and $\lfloor\!\lfloor$ we have a family of operators, each indexed by two probabilities from the interval $(0,1)$. These operators work intuitively as follows. Consider first the merge operator. In the expression $x \,\|_{\sigma,\theta}\, y$, a communication between x and y occurs with probability $1-\theta$, and an autonomous move by either x or y occurs with probability θ. Given that an autonomous move occurs, it comes from x with probability σ and from y with probability $1-\sigma$. The situation is similar for $x \,\lfloor\!\lfloor_{\sigma,\theta}\, y$ except the first step must (with probability 1) come from x. Likewise, the first step of $x \,|_{\sigma,\theta}\, y$ must result from a communication between x and y.

The treatment of the communication merge is exactly analogous to the situation in the non-probabilistic case (Section 2.3). The "totality" axiom C0 now becomes:

$$
\boxed{\forall a,b \in pr\mathbf{P}\ \overline{A_\delta}(a) \wedge \overline{A_\delta}(b) \implies \exists c \in pr\mathbf{P}\ \forall \sigma,\theta \in (0,1)\ \overline{A_\delta}(c) \wedge a \,|_{\sigma,\theta}\, b = c \quad pr\text{C0}}
$$

The axioms of prACP$_I^-(A)$ are as follows. In this system, a,b,c range over A_δ, $H_\delta = H \cup \{\delta\}$, and I has functionality $I : pr\mathbf{P} \rightarrow 2^{A_\delta}$. Also, $\cap$,$\cup$ are used on 2^{A_δ} without further specification.

prBPA(A) $\quad +$

$\delta \cdot x = \delta$	prA7

$+$

prC0 $\quad +$

$a \mid_{\sigma,\theta} b = b \mid_{(1-\sigma),\theta} a$	prC1
$(a \mid_{\sigma,\theta} b) \mid_{\sigma',\theta'} c = a \mid_{\sigma,\theta} (b \mid_{\sigma',\theta'} c)$	prC2
$\delta \mid_{\sigma,\theta} a = \delta$	prC3

$+$

$x \parallel_{\sigma,\theta} y = ((x \mathbin{⌊\!⌊}_{\sigma,\theta} y) +_{\sigma} (y \mathbin{⌊\!⌊}_{(1-\sigma),\theta} x)) +_{\theta} (x \mid_{\sigma,\theta} y)$	prCM1
$a \mathbin{⌊\!⌊}_{\sigma,\theta} y = a \cdot y$	prCM2
$(a \cdot x) \mathbin{⌊\!⌊}_{\sigma,\theta} y = a \cdot (x \parallel_{\sigma,\theta} y)$	prCM3
$(x +_{\pi} y) \mathbin{⌊\!⌊}_{\sigma,\theta} z = (x \mathbin{⌊\!⌊}_{\sigma,\theta} z) +_{\pi} (y \mathbin{⌊\!⌊}_{\sigma,\theta} z)$	prCM4
$a \mid_{\sigma,\theta} (b \cdot x) = (a \mid_{\sigma,\theta} b) \cdot x$	prCM5
$(a \cdot x) \mid_{\sigma,\theta} b = (a \mid_{\sigma,\theta} b) \cdot x$	prCM6
$(a \cdot x) \mid_{\sigma,\theta} (b \cdot y) = (a \mid_{\sigma,\theta} b) \cdot (x \parallel_{\sigma,\theta} y)$	prCM7
$(x +_{\pi} y) \mid_{\sigma,\theta} z = x \mid_{\sigma,\theta} z \ +_{\pi} \ y \mid_{\sigma,\theta} z$	prCM8
$x \mid_{\sigma,\theta} (y +_{\pi} z) = x \mid_{\sigma,\theta} y \ +_{\pi} \ x \mid_{\sigma,\theta} z$	prCM9

$+$

$I(a) = \{a\}$	prI1
$I(x \cdot y) = I(x)$	prI2
$I(x +_{\pi} y) = I(x) \cup I(y)$	prI3

$+$

$a \in H \implies \partial_H(a) = \delta$	prD1
$a \notin H \implies \partial_H(a) = a$	prD2
$I(x) \subseteq H_{\delta} \implies \partial_H(x +_{\pi} y) = \partial_H(y)$	prD3.1
$I(x +_{\pi} y) \cap H_{\delta} = \emptyset \implies \partial_H(x +_{\pi} y) = \partial_H(x) +_{\pi} \partial_H(y)$	prD3.2
$\partial_H(x \cdot y) = \partial_H(x) \cdot \partial_H(y)$	prD4

3.4.2 Graph Model

As for prBPA(A) and prPA(A), we provide a bisimulation model for prACP$^-_I(A)$ and prove completeness for finite processes. We begin with the definition of the prACP$^-_I(A)$ operators on probabilistic process graphs, and for this purpose we need to introduce a "normalization factor" to be used in computing conditional probabilities in a restricted process.

Definition 3.6 *Let $g = <V, r, \mu>$ be a probabilistic process graph. Then, for $v \in V$, the* normalization factor of v with respect to the set of actions $H \subseteq A$ *is given by*

$$\nu_H(v) = 1 - \sum\{\mu(v, a, v') \mid a \in H_\delta, v' \in V\}$$

Intuitively, $\nu_H(v)$ is the sum of the probabilities of those transitions from v that remain after restricting by the set of actions H. In the following, let $initials(v) = \{a \in A_\delta \mid \exists v' \; \mu(v, a, v') > 0\}$ for v a probabilistic process graph node, and let the empty summation of probabilities be 0.

Definition 3.7 *The operators δ, $\|_{\sigma,\theta}$, $\lfloor\!\lfloor_{\sigma,\theta}$, $|_{\sigma,\theta}$, ∂_H, $H \subseteq A$, and I are defined on $pr\mathcal{F}$ as follows:*

δ: *is given by $<\{r_\delta, v_\delta\}, r_\delta, \mu_\delta>$ where $\mu_\delta(r_\delta, \delta, v_\delta) = 1$ is the only transition with non-zero probability.*

$g_1 \|_{\sigma,\theta} g_2$: *is given by $<V_1 \times V_2, (r_1, r_2), \mu>$ where for all $a \in A_\delta$, $v_1, v_1' \in V_1$, $v_2, v_2' \in V_2$*

$$\mu((v_1, v_2), a, (v_1', v_2)) = \begin{cases} \sigma \cdot \theta \cdot \mu_1(v_1, a, v_1') & \textit{if } \neg endpoint(v_2) \\ \mu_1(v_1, a, v_1') & \textit{otherwise} \end{cases}$$

$$\mu((v_1, v_2), a, (v_1, v_2')) = \begin{cases} (1-\sigma) \cdot \theta \cdot \mu_2(v_2, a, v_2') & \textit{if } \neg endpoint(v_1) \\ \mu_2(v_2, a, v_2') & \textit{otherwise} \end{cases}$$

$$\mu((v_1, v_2), a, (v_1', v_2')) = (1-\theta) \cdot \sum_{b,c:\, b \,|_{\sigma,\theta}\, c = a} \mu_1(v_1, b, v_1') \cdot \mu_2(v_2, c, v_2')$$

$g_1 \lfloor\!\lfloor_{\sigma,\theta} g_2$: *is given by $<V_1 \times V_2, (r_1, r_2), \mu>$ where for all $a \in A_\delta$, $v_1, v_1', \in V_1$, $v_2, v_2', \in V_2$*

- $\mu((r_1, r_2), a, (v_1', r_2)) = \mu_1(r_1, a, v_1')$
- *if $v_1 \neq r_1$ or $v_2 \neq r_2$*

$$\mu((v_1, v_2), a, (v_1', v_2)) = \begin{cases} \sigma \cdot \theta \cdot \mu_1(v_1, a, v_1') & \textit{if } \neg endpoint(v_2) \\ \mu_1(v_1, a, v_1') & \textit{otherwise} \end{cases}$$

$$\mu((v_1, v_2), a, (v_1, v_2')) = \begin{cases} (1-\sigma) \cdot \theta \cdot \mu_2(v_2, a, v_2') & \textit{if } \neg endpoint(v_1) \\ \mu_2(v_2, a, v_2') & \textit{otherwise} \end{cases}$$

$$\mu((v_1, v_2), a, (v_1', v_2')) = (1-\theta) \cdot \sum_{b,c:\, b \,|_{\sigma,\theta}\, c = a} \mu_1(v_1, b, v_1') \cdot \mu_2(v_2, c, v_2')$$

- *if $v_2' \neq r_2$* $\mu((r_1, r_2), a, (v_1', v_2')) = 0$

$g_1 |_{\sigma,\theta} g_2$: *is given by $<V_1 \times V_2, (r_1, r_2), \mu>$ where for all $a \in A_\delta$, $v_1, v_1' \in V_1$, $v_2, v_2' \in V_2$*

- $\mu((r_1, r_2), a, (v_1', v_2')) = \sum_{b,c:b \,|_{\sigma,\theta}\, c=a} \mu_1(r_1, b, v_1') \cdot \mu_2(r_2, c, v_2')$
- *if $v_1 \neq r_1$ or $v_2 \neq r_2$*

$$\mu((v_1, v_2), a, (v_1', v_2)) = \begin{cases} \sigma \cdot \theta \cdot \mu_1(v_1, a, v_1') & \textit{if } \neg endpoint(v_2) \\ \mu_1(v_1, a, v_1') & \textit{otherwise} \end{cases}$$

$$\mu((v_1, v_2), a, (v_1, v_2')) = \begin{cases} (1-\sigma) \cdot \theta \cdot \mu_2(v_2, a, v_2') & \textit{if } \neg endpoint(v_1) \\ \mu_2(v_2, a, v_2') & \textit{otherwise} \end{cases}$$

$$\mu((v_1, v_2), a, (v_1', v_2')) = (1-\theta) \cdot \sum_{b,c:\, b \,|_{\sigma,\theta}\, c = a} \mu_1(v_1, b, v_1') \cdot \mu_2(v_2, c, v_2')$$

- *if ($v_1' \neq r_1$ and $v_2' = r_2$) or ($v_1' = r_1$ and $v_2' \neq r_2$)* $\mu((r_1, r_2), a, (v_1', v_2')) = 0$

$\partial_H(g_1)$: *is given by* $< V_1, r_1, \mu >$ *where, for all* $a \in A$, $v, v' \in V_1$,

- *if* $initials(v) \subseteq H_\delta$
  $\mu(v, a, v') = 0$
  $\mu(v, \delta, v') = \sum_{a \in A_\delta} \mu_1(v, a, v')$
- *if* $initials(v) \not\subseteq H_\delta$
  $$\mu(v, a, v') = \begin{cases} 0 & \text{if } a \in H_\delta \\ \mu_1(v, a, v')/\nu_H(v) & \text{otherwise} \end{cases}$$

$I(g_1)$: *gives the set of actions* $initials(r_1)$.

Similar to the case of $pr\mathrm{PA}(A)$, the first and third clauses of the definitions of $g_1 \Vert_{\sigma,\theta} g_2$ and $g_1 |_{\sigma,\theta} g_2$ collectively define the transition distribution function μ on all transitions from the root (r_1, r_2). Also note that in the definition of $\partial_H(g_1)$, division by the normalization factor $\nu_H(v)$ occurs only when $initials(v) \not\subseteq H_\delta$, which ensures that $\nu_H(v) > 0$.

Processes are still stochastic in the graph model of $pr\mathrm{ACP}_I^-(A)$ if the probability of δ-transitions is taken into account. On the other hand, one may prefer the "substochastic" interpretation that a process like $a +_{\frac{1}{2}} \delta$ performs an a-transition (after which it successfully terminates) with probability $\frac{1}{2}$, but may also do nothing (deadlock) with probability $\frac{1}{2}$. However, the process $\partial_\emptyset(a +_{\frac{1}{2}} \delta)$ never deadlocks and is equivalent to a.

The presence of δ-edges requires a new definition of probabilistic bisimulation.

Definition 3.8 *Let* $g_1 = < V_1, r_1, \mu_1 >$, $g_2 = < V_2, r_2, \mu_2 >$ *be probabilistic process graphs. A* probabilistic δ-bisimulation *between* g_1 *and* g_2 *is an equivalence relation* $\mathcal{R} \subseteq (V_1 \cup V_2) \times (V_1 \cup V_2)$ *with the following properties:*

- $\mathcal{R}(r_1, r_2)$
- $\forall v \in V_1,\ w \in V_2$ *such that* $\mathcal{R}(v, w)$:
  - $\forall a \in A,\ S \in (V_1 \cup V_2)/\mathcal{R},\quad \mu_1(v, a, S \cap V_1) = \mu_2(w, a, S \cap V_2)$
  - $\mu_1(v, \delta, V_1) = \mu_2(w, \delta, V_2)$

Graphs g_1 *and* g_2 *are* probabilistically δ-bisimilar, *written* $g_1 \leftrightarrows_\delta^{pr} g_2$, *if there exists a probabilistic* δ*-bisimulation between* g_1 *and* g_2.

The definition is the same as the earlier definition of probabilistic bisimulation except that probabilistically δ-bisimilar nodes must perform the action δ with the same total probability, without regard to where the δ-transitions lead.

In order to prove that $\leftrightarrows_\delta^{pr}$ is a congruence in $pr\mathrm{ACP}_I^-(A)$, we need the following proposition to facilitate our reasoning that $\leftrightarrows_\delta^{pr}$ respects restriction.

Proposition 3.5 *Let* $g_1 \leftrightarrows_\delta^{pr} g_2$ *and let* $\mathcal{R}$ *be a probabilistic* δ*-bisimulation between* g_1 *and* g_2 *with* $(v_1, v_2) \in \mathcal{R}$. *Then:*

1. $\mu_1(v_1, a, V_1) = \mu_2(v_2, a, V_2)$, $a \in A_\delta$

2. $initials(v_1) = initials(v_2)$

3. $\nu_H(v_1) = \nu_H(v_2)$, $H \subseteq A$.

Proof: For $a = \delta$, result (1) is immediate from Definition 3.8. For $a \neq \delta$, (1) is easily deduced from Definition 3.8 as $\mu_1(v_1, a, S \cap V_1) = \mu_2(v_2, a, S \cap V_2)$ for all equivalence classes S of the partition of $V_1 \cup V_2$ induced by $\mathcal{R}$. Results (2) and (3) are simple consequences of (1). □

Proposition 3.6 *If* $g_1 \leftrightarrows_\delta^{pr} g_2$, *then* $g \parallel_{\sigma,\theta} g_1 \leftrightarrows_\delta^{pr} g \parallel_{\sigma,\theta} g_2$, $g \,⫡_{\sigma,\theta}\, g_1 \leftrightarrows_\delta^{pr} g \,⫡_{\sigma,\theta}\, g_2$, $g_1 \,⫡_{\sigma,\theta}\, g \leftrightarrows_\delta^{pr} g_2 \,⫡_{\sigma,\theta}\, g$, $g \,|_{\sigma,\theta} g_1 \leftrightarrows_\delta^{pr} g \,|_{\sigma,\theta}\, g_2$, $\partial_H(g_1) \leftrightarrows_\delta^{pr} \partial_H(g_2)$, *for all* $H \subseteq A$, *and* $I(g_1) = I(g_2)$.

Proof: The proof for $\parallel_{\sigma,\theta}$ is similar to the proof for $\parallel_\sigma$ in Proposition 3.4. Let $a \neq \delta$. The a-transitions of non-zero probability stemming from (v, v_1) are now of the form:

1. $\mu_1^{\parallel}((v, v_1), a, [(v', v_1)]_{\mathcal{R}'}) = \sigma \cdot \theta \cdot \mu(v, a, v')$
2. $\mu_1^{\parallel}((v, v_1), a, [(v, v_1')]_{\mathcal{R}'}) = (1 - \sigma) \cdot \theta \cdot \mu_1(v_1, a, [v_1']_{\mathcal{R}})$
3. $\mu_1^{\parallel}((v, v_1), a, [(v', v_1')]_{\mathcal{R}'}) = (1 - \theta) \cdot \sum_{b,c:\, b \,|_{\sigma,\theta}\, c = a} \mu(v, b, v') \cdot \mu_1(v_1, c, [v_1']_{\mathcal{R}})$
4. $\mu_1^{\parallel}((v, v_1), \delta, V \times V_1) = \sigma \cdot \theta \cdot \mu(v, \delta, V) + (1 - \theta) \cdot \sum_{b,c:\, b \,|_{\sigma,\theta}\, c = \delta} \mu(v, b, V) \cdot \mu_1(v_1, c, V_1) + (1 - \sigma) \cdot \theta \cdot \mu_1(v_1, \delta, V_1)$

The argument for the first two types of transitions is virtually identical to the argument set forth in Proposition 3.4. For the third type, since $g_1 \leftrightarrows_\delta^{pr} g_2$, $\mu_1^{\parallel}((v, v_1), a, [(v', v_1')]_{\mathcal{R}'}) = \mu_2^{\parallel}((v, v_2), a, [(v', v_1')]_{\mathcal{R}'})$. The arguments for the first three cases collectively are sufficient for the fourth case and we are done. As in Proposition 3.4, the argument is similar if v_1 or v is an endpoint.

Again, as in Proposition 3.4, the proofs for $⫡_{\sigma,\theta}$ and $|_{\sigma,\theta}$ follow reasoning similar to, if not simpler than, the proof of $\parallel_{\sigma,\theta}$. In particular, there are fewer transitions of non-zero probability from (r, r_1) and (r, r_2) since such transitions can come from g only, in the case of probabilistic left-merge, and from communications between g,g_1 or g,g_2 only, in the case of probabilistic communication merge.

For the case of restriction, assume $g_1 \leftrightarrows_\delta^{pr} g_2$ and let $\mathcal{R}$ be a probabilistic δ-bisimulation between g_1 and g_2. We show that $\mathcal{R}$ is also a probabilistic δ-bisimulation between $\partial_H(g_1)$ and $\partial_H(g_2)$, $H \subseteq A$. Let $(v_1, v_2) \in \mathcal{R}$ and let μ_i^{∂} be the tdf of $\partial_H(g_i)$, $i = 1, 2$. If $initials(v_1) \subseteq H_\delta$ then, by Proposition 3.5, $initials(v_2) \subseteq H_\delta$ and therefore $\mu_1^{\partial}(v_1, \delta, V_1), \mu_2^{\partial}(v_2, \delta, V_2) = 1$. Otherwise, $\mu_1^{\partial}(v_1, a, V_1), \mu_2^{\partial}(v_2, a, V_2) = 0$, if $a \in H_\delta$; and for all $S \in (V_1 \cup V_2)/\mathcal{R}$, $\mu_1^{\partial}((v_1, a, S \cap V_1), \mu_2^{\partial}(v_2, a, S \cap V_2) = \mu_1(v_1, a, S \cap V_1)/\nu_H(v_1)$, if $a \notin H_\delta$. This last step is a consequence of the fact that $(v_1, v_2) \in \mathcal{R}$ and Proposition 3.5, part (3).

That $\leftrightarrows_\delta^{pr}$ respects operator I follows directly from part (3) of Proposition 3.5. □

Theorem 3.3

1. $pr\mathcal{F}/\leftrightarrows_\delta^{pr} \models pr\mathrm{ACP}_I^-(A)$

2. *For all closed expressions p,q over* $\Sigma(pr\mathrm{ACP}_I^-(A))$:

$$pr\mathcal{F}/\leftrightarrows_\delta^{pr} \models p = q \implies pr\mathrm{ACP}_I^-(A) \vdash p = q.$$

Proof: For part 1, the proof of soundness of axiom prA7 is a simple extension of the soundness argument for A7 (Theorem 2.2). Axioms prC1–3 are merely postulated about the communication merge $|_{\sigma,\theta}$. The soundness of the rest of the axioms of $pr\mathrm{ACP}_I^-(A)$ rests on probabilistic process graph isomorphism arguments (the remarks given in the soundness part of the proofs of Theorems 2.2 and 3.3 are relevant with the obvious extensions).

Note that the condition to prD3.1 implies that $\nu_H(r_x) = 0$ and the condition to prD3.2 implies that $\nu_H(r_{x+_\pi y}) = 1$ and $\nu_H(r_x), \nu_H(r_y) = 1$. The soundness of these axioms now easily follows. As alluded to in Section 2.3, unlike D3.2, prD3.2 is not sound under the weaker condition

$$I(x) - H_\delta \neq \emptyset \text{ and } I(y) - H_\delta \neq \emptyset$$

(for example, consider $x = a +_{\frac{1}{2}} b$, $y = c$, $H = \{a\}$, and $\pi = \frac{2}{3}$). This situation is closely related to the fact that the equivalence induced on the stratified model of probabilistic processes via abstraction to the generative model is not a congruence; in particular, it fails to respect restriction [vGSST90].

For part 2, the proof is analogous to the completeness proof of $\mathrm{ACP}_I^-(A)$.

- The definition of a probabilistic basic term uses $+_\pi$ instead of $+$.
- The term rewriting system $pr\mathrm{RACP}_I^-(A)$ uses the probabilistic counterparts of the rules in $\mathrm{RACP}_I^-(A)$ and the normal form is defined analogously as well. For example, $pr\mathrm{RACP}_I^-(A)$ contains the rule prC0′

$$a \,|_{\sigma,\theta}\, b = c \implies a \,|_{\sigma,\theta}\, b \longrightarrow c$$

- The proof that a probabilistic normal form is also a probabilistic basic term proceeds as before – no rule in $pr\mathrm{RACP}_I^-(A)$ is conditional with respect to any probability.
- $pr\mathrm{RACP}_I^-(A)$ is strongly normalizing modulo prA1, prA2, prA2′: take a $pr\mathrm{RACP}_I^-(A)$ reduction and erase all probability subscripts. One obtains a valid $\mathrm{RACP}_I^-(A)$ reduction.
- The "elimination theorem" for $pr\mathrm{ACP}_I^-(A)$ is also similar. Let p be a closed $pr\mathrm{ACP}_I^-(A)$ term and let $\bar{p}$ be the closed $\mathrm{ACP}_I^-(A)$ term obtained by erasing all probability subscripts. Now let
$$p = t_0 \longrightarrow t_1 \longrightarrow \cdots \longrightarrow t_n$$
be a normalizing reduction of $\bar{p}$. This reduction can be decorated appropriately with probabilities to obtain a $pr\mathrm{RACP}_I^-(A)$ normalization of p.

□

4 $\mathbf{ACP_I^-}$ as an Abstraction of $pr\mathbf{ACP_I^-}$

In this section we demonstrate that $\mathrm{ACP}_I^-(A)$ can be considered an abstraction of $pr\mathrm{ACP}_I^-(A)$ at both the level of the graph model and at the level of the equational theory. For the former, we exhibit a homomorphism ϕ from probabilistic process graphs to non-probabilistic process graphs that preserves the structure of the bisimulation congruence classes. For the latter, we exhibit a homomorphism Φ from $pr\mathrm{ACP}_I^-(A)$ terms to $\mathrm{ACP}_I^-(A)$ terms that preserves the validity of equational reasoning.

4.1 Graph Model Homomorphism

The homomorphism $\phi : pr\mathcal{G} \longrightarrow \mathcal{G}$, from probabilistic process graphs to non-probabilistic process graphs, simply "forgets" probabilities.

Definition 4.1 *Let $g = \langle V, r, \mu \rangle$ be a probabilistic process graph. Then $\phi(g) = \langle V, r, \longrightarrow \rangle$ has the same states and start state as g and $\longrightarrow$ is such that*

$$v_1 \xrightarrow{a} v_2 \iff \mu(v_1, a, v_2) > 0$$

Proposition 4.1 *Let g_1, g_2 be probabilistic process graphs.*

$$\begin{aligned}
&\phi(a) = a, a \in A_\delta \\
&\phi(g_1 \cdot g_2) = \phi(g_1) \cdot \phi(g_2) \\
&\phi(g_1 +_\pi g_2) = \phi(g_1) + \phi(g_2) \\
&\phi(g_1 \,|_{\sigma,\theta}\, g_2) = \phi(g_1) \,|\, \phi(g_2) \\
&\phi(g_1 \,\|_{\sigma,\theta}\, g_2) = \phi(g_1) \,\|\, \phi(g_2) \\
&\phi(g_1 \,\lfloor\!\lfloor_{\sigma,\theta}\, g_2) = \phi(g_1) \,\lfloor\!\lfloor\, \phi(g_2) \\
&\phi(\partial_H(g_1)) = \partial_H(\phi(g_1))
\end{aligned}$$

Proposition 4.2 *The homomorphism ϕ preserves the structure of the bisimulation congruence classes. That is,*

$$g_1 \underline{\leftrightarrow}_\delta^{pr} g_2 \implies \phi(g_1) \underline{\leftrightarrow}_\delta \phi(g_2)$$

Proof: Let $g_1 = \langle V_1, r_1, \mu_1 \rangle$, $g_2 = \langle V_2, r_2, \mu_2 \rangle$ be probabilistic process graphs, and let $\phi(g_1) = \langle V_1, r_1, \longrightarrow_1 \rangle$ and $\phi(g_2) = \langle V_2, r_2, \longrightarrow_2 \rangle$ be their homomorphic images under ϕ. Further, let $\mathcal{R} \subseteq V_1 \times V_2$ be a δ-probabilistic bisimulation containing (r_1, r_2). That is, $g_1 \underline{\leftrightarrow}_\delta^{pr} g_2$. Now let (v, w) be an arbitrary pair in $\mathcal{R}$ and assume for some $v' \in V_1$, $a \in A$ that $\mu_1(v, a, v') > 0$. By Definition 4.1, $v \xrightarrow{a}_1 v'$. Then $\mu_1(v, a, [v']) > 0$ where $[v'] = \{u \in V_1 \cup V_2 \mid (u, v') \in \mathcal{R}\} \in (V_1 \cup V_2)/\mathcal{R}$. Since $(v, w) \in \mathcal{R}$, then there exists a $w' \in [v']$ with $\mu_2(r_2, a, w') > 0$; i.e., $\mathcal{R}(v', w')$ and, by Definition 4.1 again, $r_2 \xrightarrow{a}_2 w'$. By a symmetric argument and by considering the case $a = \delta$ (which is simpler), we have as desired that $g_1 \underline{\leftrightarrow}_\delta^{pr} g_2 \implies \phi(g_1) \underline{\leftrightarrow}_\delta \phi(g_2)$. □

The converse of this result is clearly not true, e.g., $a + b \underline{\leftrightarrow}_\delta b + a$ but $a +_{\frac{1}{3}} b \not\underline{\leftrightarrow}_\delta^{pr} b +_{\frac{1}{3}} a$. Thus, the graph model $\mathcal{F}/\underline{\leftrightarrow}_\delta$ of $\mathrm{ACP}_I^-(A)$ is strictly more abstract than the probabilistic graph model $pr\mathcal{F}/\underline{\leftrightarrow}_\delta^{pr}$ of $pr\mathrm{ACP}_I^-(A)$.

4.2 Equational Theory Homomorphism

Let $\mathcal{L}(E)$ be the language of all terms, open and closed, generated by the signature of the equational specification E. The homomorphism $\Phi : \mathcal{L}(pr\mathrm{ACP}_I^-(A)) \longrightarrow \mathcal{L}(\mathrm{ACP}_I^-(A))$ from $pr\mathrm{ACP}_I^-(A)$ terms to $\mathrm{ACP}_I^-(A)$ terms, is defined as follows:

$$\Phi(a) = a, a \in A_\delta$$
$$\Phi(x) = x$$
$$\Phi(x \cdot y) = \Phi(x) \cdot \Phi(y)$$
$$\Phi(x +_\pi y) = \Phi(x) + \Phi(y)$$
$$\Phi(x \mid_{\sigma,\theta} y) = \Phi(x) \mid \Phi(y)$$
$$\Phi(x \parallel_{\sigma,\theta} y) = \Phi(x) \parallel \Phi(y)$$
$$\Phi(x \mathbin{\lfloor\!\lfloor}_{\sigma,\theta} y) = \Phi(x) \mathbin{\lfloor\!\lfloor} \Phi(y)$$
$$\Phi(\partial_H(x)) = \partial_H(\Phi(x))$$

The following proposition states that any valid proof of $pr\mathrm{ACP}_I^-(A)$ can be mapped into a valid proof of $\mathrm{ACP}_I^-(A)$ using the homomorphism Φ.

Proposition 4.3 *Let t_1, t_2 be terms of $pr\mathrm{ACP}_I^-(A)$, i.e., $t_1, t_2 \in \mathcal{L}(pr\mathrm{ACP}_I^-(A))$.*

$$\frac{pr\mathrm{ACP}_I^-(A) \vdash t_1 = t_2}{\mathrm{ACP}_I^-(A) \vdash \Phi(t_1) = \Phi(t_2)}$$

Proof: The proof is by induction on the length of the $pr\mathrm{ACP}_I^-(A)$ proof, using the observation that, for every $pr\mathrm{ACP}_I^-(A)$ axiom of the form $c \implies t_1 = t_2$, its homomorphic image $\Phi(c) \implies \Phi(t_1) = \Phi(t_2)$ is an $\mathrm{ACP}_I^-(A)$ axiom. Here c is a possibly empty condition on the validity of the $pr\mathrm{ACP}_I^-(A)$ axiom, and the fact that $\Phi(c)$ is equal to the condition of the corresponding $\mathrm{ACP}_I^-(A)$ axiom means that no axiom of $pr\mathrm{ACP}_I^-(A)$ is conditional on a probability appearing within an $pr\mathrm{ACP}_I^-(A)$ term. □

Note that the converse of the result does not hold, e.g., $a+b = b+a$ but $a +_{\frac{1}{2}} b \neq b +_{\frac{1}{3}} a$. Thus, $\mathrm{ACP}_I^-(A)$ is a strictly more abstract theory than $pr\mathrm{ACP}_I^-(A)$.

5 Comments on an Internal Probabilistic Choice Operator

In this section we consider the question whether it is possible to add a probabilistic internal choice operator to $pr\mathrm{ACP}_I^-(A)$. Such an operator $\vee_\pi : pr\mathbf{P} \times pr\mathbf{P} \to pr\mathbf{P}$ should have the following properties (similar to $\sqcap$ of CSP [Hoa85]):

1. $x \vee_\pi y$ denotes a process that equals x with probability π and equals y with probability $1-\pi$.
2. $\vee_\pi$ distributes over the operators of $pr\mathrm{ACP}_I^-(A)$, e.g., for all $\Box \in \{\cdot, +_\pi, \parallel_{\sigma,\theta}, \mathbin{\lfloor\!\lfloor}_{\sigma,\theta}, \mid_{\sigma,\theta}\}$:
$$x\Box(y \vee_\pi z) = (x\Box y) \vee_\pi (x \Box z)$$
$$(x \vee_\pi y)\Box z = (x\Box z) \vee_\pi (y \Box z)$$
3. $x \vee_\pi y = y \vee_{1-\pi} x$ and $x \vee_\pi (y \vee_\rho z) = (x \vee_{\pi/(\pi+\rho-\pi\rho)} y) \vee_{\pi+\rho-\pi\rho} z$

Each of these properties is very plausible. Nevertheless, we observe a difficulty that suggests that the setup with $\vee_\pi$ must be flawed. It follows that if an internal probabilistic choice is to be added, at least one of properties (1) – (3) must be removed. But, as stated before, these requirements are needed to simplify any setting simultaneously involving $+_\pi$ and $\vee_\pi$.

The difficulty with $\vee_\pi$ comes about as follows.

Proposition 5.1 $pr\mathrm{ACP}_I^-(A) + (1) - (3) \vdash\ a \vee_{\frac{1}{2}} b = a \vee_{\frac{1}{4}} (b \vee_{\frac{1}{3}} (a +_{\frac{1}{2}} b))$

Proof:

$$\begin{aligned} a \vee_{\frac{1}{2}} b &= (a \vee_{\frac{1}{2}} b) +_{\frac{1}{2}} (a \vee_{\frac{1}{2}} b) \\ &= \cdots \\ &= a \vee_{\frac{1}{4}} (b \vee_{\frac{1}{3}} (a +_{\frac{1}{2}} b)) \end{aligned}$$

□

Next we introduce a probability measure on traces.

Probabilities of Traces

We define $Pr : pr\mathbf{P} \times A^* \to (0,1]$ as follows:

$$\begin{aligned} &Pr(x \to \varepsilon) = 1 \\ &Pr(a \to b) = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{if } a \neq b \end{cases} \\ &Pr(a \to b * c * \sigma) = 0 \\ &Pr(a \cdot x \to b * \sigma) = Pr(a \to b) \cdot Pr(x \to \sigma) \\ &Pr(x +_\pi y \to \sigma) = \pi \cdot Pr(x \to \sigma) + (1-\pi) \cdot Pr(y \to \sigma) \\ &Pr(x \vee_\pi y \to \sigma) = \pi \cdot Pr(x \to \sigma) + (1-\pi) \cdot Pr(y \to \sigma). \end{aligned}$$

Given this meaning, it seems clear that one must require:

$$pr\mathrm{ACP}_I^-(A) + (1) - (3) \vdash p = q \quad \Longrightarrow \quad \text{for all } \sigma \in A^*\ Pr(p \to \sigma) = Pr(q \to \sigma)$$

Now consider the following example:

$$A = \{a, b, guess(a), guess(b), success(a), success(b), fail\}$$

$$\begin{aligned} &a|_{\sigma,\theta} guess(a) = success(a), \forall \sigma, \theta \in (0,1) \\ &b|_{\sigma,\theta} guess(b) = success(b), \forall \sigma, \theta \in (0,1) \\ &a|_{\sigma,\theta} guess(b) = b|_{\sigma,\theta} guess(a) = fail, \forall \sigma, \theta \in (0,1) \end{aligned}$$

All other communications are δ. Let $H = \{a, b, guess(a), guess(b)\}$, and let us write $\|$ for $|_{\frac{1}{2},\frac{1}{2}}$. Now, using Proposition 5.1, we find

$$pr\mathrm{ACP}_I^-(A) + (1) - (3) \vdash\ \partial_H(guess(a) \parallel (a \vee_{\frac{1}{2}} b)) \ = \ \partial_H(guess(a) \parallel a \vee_{\frac{1}{4}} (b \vee_{\frac{1}{3}} (a +_{\frac{1}{2}})))$$

But

$$\begin{aligned} &Pr(\partial_H(guess(a) \parallel a \vee_{\frac{1}{2}} b) \to success(a)) \\ &\qquad = Pr(\partial_H(guess(a) \parallel a) \vee_{\frac{1}{2}} \partial_H(guess(a) \parallel b) \to success(a)) \\ &\qquad = Pr(success(a) \vee_{\frac{1}{2}} fail \to success(a)) \\ &\qquad = \frac{1}{2} \end{aligned}$$

and

$$
\begin{aligned}
& Pr(\partial_H(guess(a) \parallel a \vee_{\frac{1}{4}} (b \vee_{\frac{1}{3}} (a +_{\frac{1}{2}} b))) \to success(a)) \\
&= Pr(\partial_H(guess(a) \parallel a) \vee_{\frac{1}{4}} (\partial_H(guess(a) \parallel b) \vee_{\frac{1}{3}} \partial_H(guess(a) \parallel (a +_{\frac{1}{2}} b))) \to success(a)) \\
&= Pr(success(a) \vee_{\frac{1}{4}} (fail \vee_{\frac{1}{3}} success(a)) \to success(a)) \\
&= \frac{1}{4} + \frac{3}{4} \cdot \frac{2}{3} = \frac{3}{4}
\end{aligned}
$$

This calculation indicates a definite problem for combining a probabilistic alternative composition $+_\pi$ with probabilistic internal choice $\vee_\pi$.

It follows that a generalization to a probabilistic setting of CSP that features both composition mechanisms ($\Box$ and $\sqcap$) cannot be done along the same lines.

If an internal choice must be added, the authors feel that the mentioned difficulty is best remedied by:

1. adding a sort of state distribution SD and an embedding $i : pr\mathbf{P} \to SD$ turning a process into a state distribution.
2. Then, $\vee_\pi$ can have functionality $SD \times SD \to SD$.

6 Conclusions

In this paper, we have presented complete axiomatizations of probabilistic processes within the context of the process algebra ACP. Given that axiom A6 of ACP ($x + \delta = x$) does not have a plausible interpretation in the generative model of probabilistic computation, we introduced the somewhat weaker theory ACP_I^-, in which $A6$ is rejected. ACP_I^- is, in essence, a minor alteration of ACP expressing almost the same process identities on finite processes.

Our end-result is the axiom system $pr\mathrm{ACP}_I^-$, which can be seen as a probabilistic extension of ACP_I^- for generative probabilistic processes. In particular, ACP_I^- is homomorphically derivable from $pr\mathrm{ACP}_I^-$. As desired, we showed that $pr\mathrm{ACP}_I^-$ constitutes a complete axiomatization of Larsen and Skou's probabilistic bisimulation for finite processes.

Several directions for future work can be identified. First, we are interested in adding certain important features to the model, such as recursion and unobservable τ actions. Secondly, we desire also to completely axiomatize the *reactive* and *stratified* models of probabilistic processes [vGSST90]. In the stratified model, which is well-suited for reasoning about probabilistic "fair" scheduling, distinctions are made between processes based on the branching structure of their purely probabilistic choices. We conjecture that by eliminating axiom prA2 (probabilistic alternative composition is not associative in the stratified model!) and weakening the condition to prD3.2 as discussed in the soundness part of the proof of Theorem 3.3, the desired axiomatization can be obtained.

Acknowledgements

The authors gratefully acknowledge Rob van Glabbeek, Chi-Chang Jou, and Bernhard Steffen for valuable discussions.

References

[BK84] J. A. Bergstra and J. W. Klop. Process algebra for synchronous communication. *Information and Computation*, 60:109–137, 1984.

[BM89] B. Bloom and A. R. Meyer. A remark on bisimulation between probabilistic processes. In Meyer and Tsailin, editors, *Logik at Botik*, Springer-Verlag, 1989.

[BW90] J. C. M. Baeten and W. P. Weijland. *Process Algebra. Cambridge Tracts in Computer Science 18*, Cambridge University Press, 1990.

[Chr90] I. Christoff. *Testing Equivalences for Probabilistic Processes.* Technical Report DoCS 90/22, Ph.D. Thesis, Department of Computer Science, Uppsala University, Uppsala, Sweden, 1990.

[CSZ92] R. Cleaveland, S. A. Smolka, and A. E. Zwarico. Testing preorders for probabilistic processes. In *Proceedings of the 19th ICALP*, July 1992.

[GJS90] A. Giacalone, C.-C. Jou, and S. A. Smolka. Algebraic reasoning for probabilistic concurrent systems. In *Proceedings of Working Conference on Programming Concepts and Methods*, IFIP TC 2, Sea of Gallilee, Israel, April 1990.

[Hoa85] C. A. R. Hoare. *Communicating Sequential Processes.* Prentice-Hall, London, 1985.

[JL91] B. Jonsson and K. G. Larsen. Specification and refinement of probabilistic processes. In *Proceedings of the 6th IEEE Symposium on Logic in Computer Science*, Amsterdam, July 1991.

[JS90] C.-C. Jou and S. A. Smolka. Equivalences, congruences, and complete axiomatizations for probabilistic processes. In J. C. M. Baeten and J. W. Klop, editors, *Proceedings of CONCUR '90*, pages 367–383, Springer-Verlag, Berlin, 1990.

[LS89] K. G. Larsen and A. Skou. Bisimulation through probabilistic testing. In *Proceedings of 16th Annual ACM Symposium on Principles of Programming Languages*, 1989.

[LS92] K. G. Larsen and A. Skou. Compositional verification of probabilistic processes. In *Proceedings of CONCUR '92*, Springer-Verlag Lecture Notes in Computer Science, 1992.

[Mil80] R. Milner. *A Calculus of Communicating Systems.* Volume 92 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, 1980.

[Par81] D. M. R. Park. Concurrency and automata on infinite sequences. In *Proceedings of 5th G.I. Conference on Theoretical Computer Science*, pages 167–183, Springer-Verlag, 1981.

[Tof90] C. M. N. Tofts. A synchronous calculus of relative frequency. In J. C. M. Baeten and J. W. Klop, editors, *Proceedings of CONCUR '90*, pages 467–480, Springer-Verlag, Berlin, 1990.

[vGSST90] R. J. van Glabbeek, S. A. Smolka, B. Steffen, and C. M. N. Tofts. Reactive, generative, and stratified models of probabilistic processes. In *Proceedings of the 5th IEEE Symposium on Logic in Computer Science*, pages 130–141, Philadelphia, PA, 1990.

A Proofs of Lemmas 3.1 and 3.2

Lemma 3.1 For any permutation ξ of $\{1, \cdots, n\}$, $n \geq 2$,

$$prBPA(A) \vdash \sum_{i=1}^{n} [\pi_i] x_i = \sum_{i=1}^{n} [\pi_{\xi(i)}] x_{\xi(i)}$$

Proof: The proof is by induction on n. All non-annotated steps are assumed to follow directly from the definition of summation form notation.

- Basis: $n = 2$
  We prove the non-trivial case where $\xi(1) = 2, \xi(2) = 1$.

$$\begin{aligned}
\sum_{i=1}^{2} [\pi_i]\, x_i = \quad & x_1 +_{\pi_1} \sum_{i=1}^{1} \left[\frac{\pi_{i+1}}{1-\pi_1}\right] x_{i+1} \\
= \quad & x_1 +_{\pi_1} x_2 \\
= \quad & x_2 +_{\pi_2} x_1 && (prA1) \\
= \quad & x_2 +_{\pi_2} \sum_{i=1}^{1} \left[\frac{\pi_i}{1-\pi_2}\right] x_i \\
= \quad & \sum_{i=1}^{2} [\pi_{\xi(i)}] x_{\xi(i)}
\end{aligned}$$

- Hypothesis: supoose the lemma holds for $n \leq k$.
- Induction: $n = k + 1$
  If $\xi(1) = 1$, then we have

$$\begin{aligned}
\sum_{i=1}^{k+1} [\pi_i]\, x_i = \quad & x_1 +_{\pi_1} \sum_{i=1}^{k} \left[\frac{\pi_{i+1}}{1-\pi_1}\right] x_{i+1} \\
= \quad & x_1 +_{\pi_1} \sum_{i=1}^{k} \left[\frac{\pi_{\xi(i+1)}}{1-\pi_1}\right] x_{\xi(i+1)} && \text{(induction)} \\
= \quad & \sum_{i=1}^{k+1} [\pi_{\xi(i)}] x_{\xi(i)}
\end{aligned}$$

  If $\xi(1) = j \neq 1$, then

$$\begin{aligned}
\sum_{i=1}^{k+1} [\pi_i]\, x_i = \quad & x_1 +_{\pi_1} \sum_{i=1}^{k} \left[\frac{\pi_{i+1}}{1-\pi_1}\right] x_{i+1} \\
= \quad & x_1 +_{\pi_1} \sum_{i=1}^{k} \left[\frac{\pi_{\xi'(i+1)}}{1-\pi_1}\right] x_{\xi'(i+1)} && \text{(induction)}
\end{aligned}$$

where ξ' is any permutation from 2 to $n + 1$
with $\xi'(2) = j$

$$= \quad x_1 +_{\pi_1} \left(x_j +_{\frac{\pi_j}{1-\pi_1}} \sum_{i=1}^{k-1} \left[\frac{\pi_{\xi'(i+2)}}{1-\pi_1-\pi_j} \right] x_{\xi'(i+2)} \right)$$

$$= \quad x_1 +_{\pi_1} \left(\sum_{i=1}^{k-1} \left[\frac{\pi_{\xi'(i+2)}}{1-\pi_1-\pi_j} \right] x_{\xi'(i+2)} +_{\frac{1-\pi_1-\pi_j}{1-\pi_1}} x_j \right) \qquad (pr\mathrm{A1})$$

$$= \quad \left(x_1 +_{\frac{\pi_1}{1-\pi_j}} \sum_{i=1}^{k-1} \left[\frac{\pi_{\xi'(i+2)}}{1-\pi_1-\pi_j} \right] x_{\xi'(i+2)} \right) +_{1-\pi_j} x_j \qquad (pr\mathrm{A2})$$

$$= \quad x_j +_{\pi_j} \left(x_1 +_{\frac{\pi_1}{1-\pi_j}} \sum_{i=1}^{k-1} \left[\frac{\pi_{\xi'(i+2)}}{1-\pi_1-\pi_j} \right] x_{\xi'(i+2)} \right) \qquad (pr\mathrm{A1})$$

$$= \quad x_j +_{\pi_j} \left(y_1 +_{\rho_1} \sum_{i=1}^{k-1} \left[\frac{\rho_{i+1}}{1-\rho_1} \right] y_{i+1} \right)$$

where $y_1 = x_1$, $\rho_1 = \frac{\pi_1}{1-\pi_j}$,
for $1 \le i \le k-1$, $y_{i+1} = x_{\xi'(i+2)}$, $\rho_{i+1} = \frac{\pi_{\xi'(i+2)}}{1-\pi_j}$

$$= \quad x_j +_{\pi_j} \sum_{i=1}^{k} [\rho_i] \, y_i$$

$$= \quad x_j +_{\pi_j} \sum_{i=1}^{k} \left[\rho_{\xi''(i)} \right] y_{\xi''(i)} \qquad \text{(induction)}$$

where ξ'' is the permutation of 1 to k with
$y_{\xi''(i)} = x_{\xi(i+1)}$ and $\rho_{\xi''(i)} = \frac{\pi_{\xi(i+1)}}{1-\pi_j}$

$$= \quad x_j +_{\pi_j} \sum_{i=1}^{k} \left[\frac{\pi_{\xi(i+1)}}{1-\pi_j} \right] x_{\xi(i+1)}$$

$$= \quad \sum_{i=1}^{k+1} [\pi_{\xi(i)}] x_{\xi(i)}$$

□

Lemma 3.2 In the summation form $\sum_{i=1}^{n+1}[\pi_i]x_i$, let x_1 and x_2 be syntactically identical. Then

$$pr\text{BPA}(A) \vdash \sum_{i=1}^{n+1}[\pi_i]x_i = \sum_{i=1}^{n}[\rho_i]y_i$$

where $\rho_1 = \pi_1 + \pi_2$, $y_1 = x_1$, and $\rho_i = \pi_{i+1}$, $y_i = x_{i+1}$, $2 \leq i \leq n$.

Proof: There are two cases; all non-annotated steps are assumed to follow directly from the definition of summation form notation. If $n = 1$, then we have:

$$\begin{aligned}
\sum_{i=1}^{2}[\pi_i]\,x_i &= x_1 +_{\pi_1} \sum_{i=1}^{1}\left[\frac{\pi_{i+1}}{1-\pi_1}\right] x_{i+1} \\
&= x_1 +_{\pi_1} \sum_{i=1}^{1}\left[\frac{1-\pi_1}{1-\pi_1}\right] x_1 \\
&= x_1 +_{\pi_1} x_1 \\
&= x_1 && (pr\text{A3}) \\
&= \sum_{i=1}^{1}[\rho_i]\,y_i
\end{aligned}$$

If $n \geq 2$, then we have:

$$\begin{aligned}
\sum_{i=1}^{n+1}[\pi_i]\,x_i &= x_1 +_{\pi_1} \sum_{i=1}^{n}\left[\frac{\pi_{i+1}}{1-\pi_1}\right] x_{i+1} \\
&= x_1 +_{\pi_1} \left(x_2 +_{\frac{\pi_2}{1-\pi_1}} \sum_{i=1}^{n-1}\left[\frac{\pi_{i+2}}{1-\pi_1-\pi_2}\right] x_{i+2}\right) \\
&= (x_1 +_{\frac{\pi_1}{\pi_1+\pi_2}} x_1) +_{\pi_1+\pi_2} \sum_{i=1}^{n-1}\left[\frac{\pi_{i+2}}{1-\pi_1-\pi_2}\right] x_{i+2} && (pr\text{A2}) \\
&= x_1 +_{\pi_1+\pi_2} \sum_{i=1}^{n-1}\left[\frac{\pi_{i+2}}{1-\pi_1-\pi_2}\right] x_{i+2} && (pr\text{A3}) \\
&= y_1 +_{\rho_1} \sum_{i=1}^{n-1}\left[\frac{\rho_{i+1}}{1-\rho_1}\right] y_{i+1} && (\text{Given condition}) \\
&= \sum_{i=1}^{n}[\rho_i]\,y_i
\end{aligned}$$

□

In this series appeared:

90/1	W.P.de Roever- H.Barringer- C.Courcoubetis-D.Gabbay R.Gerth-B.Jonsson-A.Pnueli M.Reed-J.Sifakis-J.Vytopil P.Wolper	Formal methods and tools for the development of distributed and real time systems, p. 17.
90/2	K.M. van Hee P.M.P. Rambags	Dynamic process creation in high-level Petri nets, pp. 19.
90/3	R. Gerth	Foundations of Compositional Program Refinement - safety properties - , p. 38.
90/4	A. Peeters	Decomposition of delay-insensitive circuits, p. 25.
90/5	J.A. Brzozowski J.C. Ebergen	On the delay-sensitivity of gate networks, p. 23.
90/6	A.J.J.M. Marcelis	Typed inference systems : a reference document, p. 17.
90/7	A.J.J.M. Marcelis	A logic for one-pass, one-attributed grammars, p. 14.
90/8	M.B. Josephs	Receptive Process Theory, p. 16.
90/9	A.T.M. Aerts P.M.E. De Bra K.M. van Hee	Combining the functional and the relational model, p. 15.
90/10	M.J. van Diepen K.M. van Hee	A formal semantics for Z and the link between Z and the relational algebra, p. 30. (Revised version of CSNotes 89/17).
90/11	P. America F.S. de Boer	A proof system for process creation, p. 84.
90/12	P.America F.S. de Boer	A proof theory for a sequential version of POOL, p. 110.
90/13	K.R. Apt F.S. de Boer E.R. Olderog	Proving termination of Parallel Programs, p. 7.
90/14	F.S. de Boer	A proof system for the language POOL, p. 70.
90/15	F.S. de Boer	Compositionality in the temporal logic of concurrent systems, p. 17.
90/16	F.S. de Boer C. Palamidessi	A fully abstract model for concurrent logic languages, p. p. 23.
90/17	F.S. de Boer C. Palamidessi	On the asynchronous nature of communication in logic languages: a fully abstract model based on sequences, p. 29.

90/18	J.Coenen E.v.d.Sluis E.v.d.Velden	Design and implementation aspects of remote procedure calls, p. 15.
90/19	M.M. de Brouwer P.A.C. Verkoulen	Two Case Studies in ExSpect, p. 24.
90/20	M.Rem	The Nature of Delay-Insensitive Computing, p.18.
90/21	K.M. van Hee P.A.C. Verkoulen	Data, Process and Behaviour Modelling in an integrated specification framework, p. 37.
91/01	D. Alstein	Dynamic Reconfiguration in Distributed Hard Real-Time Systems, p. 14.
91/02	R.P. Nederpelt H.C.M. de Swart	Implication. A survey of the different logical analyses "if...,then...", p. 26.
91/03	J.P. Katoen L.A.M. Schoenmakers	Parallel Programs for the Recognition of *P*-invariant Segments, p. 16.
91/04	E. v.d. Sluis A.F. v.d. Stappen	Performance Analysis of VLSI Programs, p. 31.
91/05	D. de Reus	An Implementation Model for GOOD, p. 18.
91/06	K.M. van Hee	SPECIFICATIEMETHODEN, een overzicht, p. 20.
91/07	E.Poll	CPO-models for second order lambda calculus with recursive types and subtyping, p. 49.
91/08	H. Schepers	Terminology and Paradigms for Fault Tolerance, p. 25.
91/09	W.M.P.v.d.Aalst	Interval Timed Petri Nets and their analysis, p.53.
91/10	R.C.Backhouse P.J. de Bruin P. Hoogendijk G. Malcolm E. Voermans J. v.d. Woude	POLYNOMIAL RELATORS, p. 52.
91/11	R.C. Backhouse P.J. de Bruin G.Malcolm E.Voermans J. van der Woude	Relational Catamorphism, p. 31.
91/12	E. van der Sluis	A parallel local search algorithm for the travelling salesman problem, p. 12.
91/13	F. Rietman	A note on Extensionality, p. 21.
91/14	P. Lemmens	The PDB Hypermedia Package. Why and how it was built, p. 63.

91/15	A.T.M. Aerts K.M. van Hee	Eldorado: Architecture of a Functional Database Management System, p. 19.
91/16	A.J.J.M. Marcelis	An example of proving attribute grammars correct: the representation of arithmetical expressions by DAGs, p. 25.
91/17	A.T.M. Aerts P.M.E. de Bra K.M. van Hee	Transforming Functional Database Schemes to Relational Representations, p. 21.
91/18	Rik van Geldrop	Transformational Query Solving, p. 35.
91/19	Erik Poll	Some categorical properties for a model for second order lambda calculus with subtyping, p. 21.
91/20	A.E. Eiben R.V. Schuwer	Knowledge Base Systems, a Formal Model, p. 21.
91/21	J. Coenen W.-P. de Roever J.Zwiers	Assertional Data Reification Proofs: Survey and Perspective, p. 18.
91/22	G. Wolf	Schedule Management: an Object Oriented Approach, p. 26.
91/23	K.M. van Hee L.J. Somers M. Voorhoeve	Z and high level Petri nets, p. 16.
91/24	A.T.M. Aerts D. de Reus	Formal semantics for BRM with examples, p. 25.
91/25	P. Zhou J. Hooman R. Kuiper	A compositional proof system for real-time systems based on explicit clock temporal logic: soundness and complete ness, p. 52.
91/26	P. de Bra G.J. Houben J. Paredaens	The GOOD based hypertext reference model, p. 12.
91/27	F. de Boer C. Palamidessi	Embedding as a tool for language comparison: On the CSP hierarchy, p. 17.
91/28	F. de Boer	A compositional proof system for dynamic proces creation, p. 24.
91/29	H. Ten Eikelder R. van Geldrop	Correctness of Acceptor Schemes for Regular Languages, p. 31.
91/30	J.C.M. Baeten F.W. Vaandrager	An Algebra for Process Creation, p. 29.
91/31	H. ten Eikelder	Some algorithms to decide the equivalence of recursive types, p. 26.

91/32	P. Struik	Techniques for designing efficient parallel programs, p. 14.
91/33	W. v.d. Aalst	The modelling and analysis of queueing systems with QNM-ExSpect, p. 23.
91/34	J. Coenen	Specifying fault tolerant programs in deontic logic, p. 15.
91/35	F.S. de Boer J.W. Klop C. Palamidessi	Asynchronous communication in process algebra, p. 20.
92/01	J. Coenen J. Zwiers W.-P. de Roever	A note on compositional refinement, p. 27.
92/02	J. Coenen J. Hooman	A compositional semantics for fault tolerant real-time systems, p. 18.
92/03	J.C.M. Baeten J.A. Bergstra	Real space process algebra, p. 42.
92/04	J.P.H.W.v.d.Eijnde	Program derivation in acyclic graphs and related problems, p. 90.
92/05	J.P.H.W.v.d.Eijnde	Conservative fixpoint functions on a graph, p. 25.
92/06	J.C.M. Baeten J.A. Bergstra	Discrete time process algebra, p.45.
92/07	R.P. Nederpelt	The fine-structure of lambda calculus, p. 110.
92/08	R.P. Nederpelt F. Kamareddine	On stepwise explicit substitution, p. 30.
92/09	R.C. Backhouse	Calculating the Warshall/Floyd path algorithm, p. 14.
92/10	P.M.P. Rambags	Composition and decomposition in a CPN model, p. 55.
92/11	R.C. Backhouse J.S.C.P.v.d.Woude	Demonic operators and monotype factors, p. 29.
92/12	F. Kamareddine	Set theory and nominalisation, Part I, p.26.
92/13	F. Kamareddine	Set theory and nominalisation, Part II, p.22.
92/14	J.C.M. Baeten	The total order assumption, p. 10.
92/15	F. Kamareddine	A system at the cross-roads of functional and logic programming, p.36.
92/16	R.R. Seljée	Integrity checking in deductive databases; an exposition, p.32.
92/17	W.M.P. van der Aalst	Interval timed coloured Petri nets and their analysis, p. 20.

92/18	R.Nederpelt F. Kamareddine	A unified approach to Type Theory through a refined lambda-calculus, p. 30.
92/19	J.C.M.Baeten J.A.Bergstra S.A.Smolka	Axiomatizing Probabilistic Processes: ACP with Generative Probabilities, p. 36.
92/20	F.Kamareddine	Are Types for Natural Language? P. 32.
92/21	F.Kamareddine	Non well-foundedness and type freeness can unify the interpretation of functional application, p. 16.
92/22	R. Nederpelt F.Kamareddine	A useful lambda notation, p. 17.
92/23	F.Kamareddine E.Klein	Nominalization, Predication and Type Containment, p. 40.
92/24	M.Codish D.Dams Eyal Yardeni	Bottum-up Abstract Interpretation of Logic Programs, p. 33.
92/25	E.Poll	A Programming Logic for Fω, p. 15.