Probabilistic Metric Semantics for a Simple Language with Recursion

Marta Kwiatkowska and Gethin Norman

School of Computer Science University of Birmingham,
Edgbaston, Birmingham B15 2TT, UK

Abstract. We consider a simple divergence-free language RP for reactive processes which includes prefixing, deterministic choice, action-guarded probabilistic choice, synchronous parallel and recursion. We show that the probabilistic bisimulation of Larsen & Skou is a congruence for this language. Following the methodology introduced by de Bakker & Zucker we give denotational semantics to this language by means of a complete metric space of (deterministic) probabilistic trees defined in terms of the powerdomain of closed sets. This new metric, although not an ultra-metric, nevertheless specialises to the metric of de Bakker & Zucker. Our semantic domain admits a full abstraction result with respect to probabilistic bisimulation.

1 Introduction

Probabilistic and stochastic phenomena are important in many areas of computing, for example, distributed systems, fault tolerance, communication protocols and performance analysis, and thus formal and automated tools for reasoning about such systems are needed. This paper makes a contribution towards the foundations of languages for specifying probabilistic systems, and thus furthers understanding of the probabilistic phenomena which have so far proved troublesome to handle by conventional techniques, see e.g. the probabilistic powerdomain construction [9].

The recent trend in the semantics of programming languages has been to supply a language with three pairwise "equivalent" semantics: operational, denotational and logical. Each semantics gives a different view of the language – the operational focuses on the transition system, denotational on compositionality, while the logical on the properties and satisfaction – and a statement of their "equivalence" states how closely they are related. The work of Kozen [13] for a *while* language with random assignment is a pre-cursor of this approach in the area of probabilistic languages, but so far no framework encompassing the three semantics has been proposed for a probabilistic extension of a process algebra.

In this paper we consider a probabilistic variant of a process algebra (a "reactive" language in the terminology of [20]) based on CCS [17] and CSP [8]. The calculus contains recursion, deterministic choice and concurrency, but instead of non-deterministic choice it has (action guarded) probabilistic choice.

W. Penczek and A. Szalas (Eds.), Proc. 21st International Symposium on Mathematical Foundations of Computer Science (MFCS'96), volume 2324 of LNCS, pages 419–430, 1996.

The operational semantics of this language is given in terms of the probabilistic transition systems and probabilistic bisimulation of Larsen & Skou [14]. The calculus is provided with a denotational, metric-space semantics derived following the techniques introduced by de Bakker & Zucker [5] for the non-probabilistic case. We show the semantics to be fully abstract with respect to the probabilistic bisimulation. Our result can be seen as complementing the framework of Larsen & Skou who (without considering a calculus) give a logical characterization of probabilistic bisimulation in terms of probabilistic modal logic.

Existing research in this area has focussed mainly on the operational side, see e.g [2,4,6,10,12,15,19,21]. In [2,12,15,19] complete axiomatizations of the constructed probabilistic process calculi are given, with [15] dealing with a reactive model and [2,12] generative models in the terminology of [20]. The probabilistic powerdomain construction [9] has been applied to give domain-theoretic semantics to certain languages, but as yet no fully abstract metric model has been proposed. Fully abstract characterizations for testing equivalences are included in [11,4,21]; denotational semantics is given in [11], but recursion is not considered. [18] introduces denotational semantics for probabilistic CSP in terms of conditional probability measures on the space of infinite traces. A "metric" for ϵ-bisimulation can be found in [6]; in contrast to ours, it does not satisfy the axioms of a metric.

We omit most details of the proofs from this version of the paper.

2 Probabilistic Transition Systems and Bisimulation

We assume the reader has some knowledge of metric spaces and the methodology for metric denotational semantics (see e.g. [5]).

Let D be a set. A *probability distribution with countable support* on D is a function $f : D \longrightarrow [0,1]$ such that the set $s(f) = \{d \in D \mid f(d) > 0\}$ is countable and $\sum_{d \in D} f(d) = 1$. Unless otherwise stated, by a *probability distribution* we shall mean a probability distribution with countable support. Let D be a set, and let $\mu(D)$ denote the family of probability distributions on D. Given any probability distribution f, and a set D such that $s(f) \subseteq D$, f can be extended to f_D such that $f_D \in \mu(D)$. When it is clear from the context what the set D is we write f instead of f_D.

Proposition 1. *The family $\mu(D)$ of probability distributions on D is a metric space with respect to the metric:*

$$d_\mu(f,g) = \frac{1}{2} \sum_{p \in s(f) \cup s(g)} | f(p) - g(p) | \ .$$

Furthermore, for all f and $g \in \mu(D)$, $0 \leq d_\mu(f,g) \leq 1$.

We recall the notions of probabilistic transition systems and probabilistic bisimulation introduced originally by Larsen & Skou [14]. A *probabilistic transition system* is a tuple $S = (P, Act, Can, \mu)$ where P is a set of processes (states),

Act is a set of observable actions, Can is an *Act*-indexed family of sets of processes where Can_a is the set of processes capable of performing the action a as their initial move, μ is a family of probabilistic distributions, $\mu_{p,a} : P \longrightarrow [0,1]$, for $a \in Act$, $p \in Can_a$, indicating the possible next states and their probabilities after p has performed a, i.e. $\mu_{p,a}(q) = \lambda$ means that the probability that p becomes q after performing a is λ.

Note that it is required that $\sum_{p' \in P} \mu_{a,p}(p') = 1$ since $\mu_{p,a}$ is a probability distribution. A probabilistic transition can, for a given state p and action a, be thought of as yielding a probabilistic distribution on the set of all processes P. The notation for probabilistic transitions is as follows:

$$p \xrightarrow{a}_{\lambda} p' \text{ whenever } p \in Can_a \text{ and } \mu_{p,a}(p') = \lambda$$
$$p \xrightarrow{a} p' \text{ whenever } p \xrightarrow{a}_{\lambda} p' \text{ for some } \lambda > 0 \ .$$

It should be noted that Larsen and Skou's definition models *reactive* systems in the terminology of van Glabbeek et al [20]. In the reactive model for probabilistic processes, a button-pressing experiment suceeds with probability 1, or else fails. When successful, the process makes an internal state transition according to a probability distribution associated with the depressed button. More formally, in the reactive model the probabilities are *action guarded*, meaning there is a (single) probability distribution for each process and action it can perform, thus imposing *determinism* at the language level.

Definition 1. *Let (P, Act, Can, μ) be a probabilistic transition system. A* probabilistic bisimulation *R on P is an equivalence relation $R \subseteq P \times P$ such that whenever pRq the following holds:*

$$\forall a \in Act \; \forall S \in P/R \; . \; p \xrightarrow{a}_{\lambda} S \iff q \xrightarrow{a}_{\lambda} S$$

where P/R denotes the quotient of P by R and for any $p \in P$ and $S \in P/R$ $p \xrightarrow{a}_{\lambda} S$ if and only if $\lambda = \sum_{s \in S} \mu_{p,a}(s)$. Two processes p and q are probabilistic bisimilar *(notation $p \sim q$) if they are contained in a probabilistic bisimulation. The largest probabilistic bisimulation is denoted by $\sim$.*

3 Language RP and its Operational Semantics

We consider a divergence-free probabilistic process algebra based on CCS [17] and CSP [8], referred to as RP, which includes recursion and deterministic choice, but instead of the usual non-determinism has action-guarded probabilistic choice. The language derives from the need to model reactive systems. We choose RP instead of PCCS [6] as RP is more intuitive for reactive processes and avoids the need for two types of transitions.

Let *Act* denote the set of actions (ranged over by $a, b \ldots$), and $\mathcal{X}$ the set of process variables (ranged over by $x, y \ldots$), both sets being countable. The syntax of all expressions is as follows:

$$q ::= \underline{0} \mid x \mid \sum_{i \in I} a_{\mu_i}.q_i \mid q_1 \oplus q_2 \text{ where } \mathcal{A}(q_1) \cap \mathcal{A}(q_2) = \emptyset \mid q_1 || q_2 \mid fix\ x.q$$

where $a \in Act$, $x \in \mathcal{X}$, $\underline{0}$ denotes the inactive process, $\sum_{i\in I}\ a_{\mu_i}.q_i$ (where $a \in Act$, I is an index set, and $\mu_i \in (0,1]$ is a countable set of real numbers such that $\sum_{i\in I}\mu_i = 1$) denotes probabilistic choice, $q_1 \oplus q_2$ denotes deterministic choice (we require that the sets of initial actions of q_1 and q_2 are disjoint), $q_1||q_2$ denotes synchronous parallel, and $fixx.q$ denotes recursion. In the case of I finite we also write $a_{\mu_1}.q_1 + a_{\mu_2}.q_2 + ...a_{\mu_n}.q_n$. Formally, the set $\mathcal{A}(q)$ of initial actions of q is defined inductively by setting $\mathcal{A}(\underline{0}) = \mathcal{A}(x) = \emptyset$, $\mathcal{A}(\sum_{i\in I} a_{\mu_i}.q_i) = \{a\}$, $\mathcal{A}(fixx.q) = \mathcal{A}(q)$, and $\mathcal{A}(q_1 \oplus q_2) = \mathcal{A}(q_1||q_2) = \mathcal{A}(q_1) \cup \mathcal{A}(q_2)$. We only consider the subset of the *guarded* expressions $\mathcal{E}$ defined over syntax:

$$p ::= \underline{0} \mid \sum_{i\in I} a_{\mu_i}.q_i \mid p_1 \oplus p_2 \text{ where } \mathcal{A}(p_1) \cap \mathcal{A}(p_2) = \emptyset \mid p_1||p_2 \mid fix\ x.p \ .$$

Observe that prefixing is a special case of probabilistic choice: $a.p$ is equivalent to $a_1.p$, meaning that after a is performed the process becomes p with probability 1. The syntactic restriction on the choice operator is necessary to draw comparisons with Larsen and Skou's formalism [14]. The operational semantics is as follows:

$$\textbf{Act}\ \frac{}{\sum_{i\in I} a_{\mu_i}.p_i \xrightarrow{a}_{\mu}\ p_j}\ j \in I \text{ and } \mu = \sum_{p_i = p_j} \mu_i$$

$$\textbf{Sum}_\textbf{1}\ \frac{p_1 \xrightarrow{a}_{\mu}\ q}{p_1 \oplus p_2 \xrightarrow{a}_{\mu}\ q} \qquad \textbf{Sum}_\textbf{2}\ \frac{p_2 \xrightarrow{a}_{\mu}\ q}{p_1 \oplus p_2 \xrightarrow{a}_{\mu}\ q}$$

$$\textbf{Par}\ \frac{p_1 \xrightarrow{a}_{\mu_1}\ q_1 \text{ and } p_2 \xrightarrow{a}_{\mu_2}\ q_2}{p_1||p_2 \xrightarrow{a}_{\mu_1\mu_2}\ q_1||q_2} \qquad \textbf{Rec}\ \frac{p\,\{fix\ x.p/x\} \xrightarrow{a}_{\mu}\ q}{fix\ x.p \xrightarrow{a}_{\mu}\ q}$$

where $q\{p/x\}$ denotes the result of changing all free occurences of x in q by p, with change of bound variables to avoid clashes. Following the usual convention, we define the set RP of *(guarded) processes* of the language as the set of expressions in $\mathcal{E}$ with no free variables.

Proposition 2. *Probabilistic bisimulation is a congruence for the language RP, i.e. it is preserved by all contexts of the language.*

Furthermore, the equational laws below, derived following Milner [16], characterise RP:

$$\begin{array}{rccl}
(\oplus\mathbf{1}) & p_1 \oplus p_2 & = & p_2 \oplus p_1 \\
(\oplus\mathbf{2}) & p_1 \oplus (p_2 \oplus p_3) & = & (p_1 \oplus p_2) \oplus p_3 \\
(\oplus\mathbf{3}) & p \oplus \underline{0} & = & p \\
(\mathbf{Par1}) & p_1||p_2 & = & p_2||p_1 \\
(\mathbf{Par2}) & p||\underline{0} & = & p \\
(\mathbf{Rec1}) & fix\ x.p & = & p\,\{fix\ x.p/x\} \\
(\mathbf{Rec2}) & q = p\,\{q/x\} & \Rightarrow & q = fix\ x.p \\
(\mathbf{Act}) & \sum_{i\in I} a_{\mu_i}.p_i & = & \sum_{i\in I\setminus J} a_{\mu_i}.p_i + a_{\mu}.p
\end{array}$$

where $p \in \mathrm{RP}$ and $J \subseteq I$ such that for all $j \in J$ we have $p_j = p$ and $\mu = \sum_{j\in J}\mu_j$. It follows from the **Rec** laws that fixed points are unique up to bisimulation.

4 A Metric for Probabilistic Computations

We first turn our attention to probabilistic computations, which should be thought of as suitable generalizations of sequential computations (= sequences of steps) of de Bakker & Zucker [5]. As in the non-probabilistic case, a probabilistic process will be represented by a certain set of such computations.

Intuitively, a probabilistic computation step will be represented by a pair consisting of an action and a probabilistic distribution, i.e. an element of the set $A \times \mu(D)$, where D is assumed to be the set of all probabilistic computations. Thus, each such step $p = (a, f)$, for some $f \in \mu(D)$ and $a \in A$, can be viewed as the process which can perform the action a and become a process $q \in D$ with probability $f(q)$. To allow for termination we also require a distinguished element p_0 to model the inactive process. This gives:

$$D \cong \{p_0\} \cup A \times \mu(D)$$

as the candidate for a domain equation for probabilistic computations.

We proceed by applying the techniques of [5] to derive an inductively defined collection of metric spaces $(D_n, d), n = 0, 1, \ldots$, where the elements of the spaces model *finite* probabilistic computations. Informally, $D_0 \subseteq D_1 \subseteq \ldots \subseteq D_n \ldots$ form a sequence of sets, where as n increases the number of probabilistic processes which are modelled increases, with D_n modelling the processes capable of performing one probabilistic action at a time up to *the depth* n. Formally:

Definition 2 ((Finite probabilistic computations)). *Let $D_n, n = 0, 1, \ldots$, be a collection of carrier sets defined inductively by:*

$$D_0 = \{p_0\} \;\; \textit{and} \;\; D_{n+1} = \{p_0\} \cup A \times \mu(D_n)$$

where A is a set of actions. Let $D_\omega = \bigcup_n D_n$ denote bounded computations.

For simplicity, we consider any $f \in \mu(D_n)$ as the extension of f to D_ω, i.e. $f_{D_\omega} \in \mu(D_\omega)$, with the subscript often dropped. We now explain the intuition behind our metric on probabilistic computations D_ω: the distance is set to 1 if the computations differ on the initial action, and to a (possibly infinite) sum derived from the distances between the resulting distributions otherwise. The latter involves the notion of a *truncation* on distributions, which we now define.

Definition 3. *Let $f \in \mu(D_\omega)$. For $k \in \mathbb{N}$ define the kth* truncation *of f, $f[k] \in \mu(D_k)$, as follows. The support of $f[k]$ is given by $s(f[k]) \stackrel{def}{=} \bigcup\{\, p[k] \,|\, p \in s(f) \,\}$, and for any $q \in D_k$,*

$$f[k](q) = \begin{cases} 0 & \textit{if } q \notin s(f[k]) \\ \sum\{f(p) \mid p \in s(f) \textit{ and } p[k] = q\} & \textit{otherwise} \end{cases}$$

where for $p \in D_\omega$ the auxiliary truncation on probabilistic computations, $p[k] \in D_k$, is defined inductively on $k \in \mathbb{N}$ by putting $p[0] = p_0$ for all p and

$$p[k+1] = \begin{cases} p_0 & \textit{if } p = p_0 \\ (a, f[k]) & \textit{if } p = (a, f) \textit{ for some } a \in A \textit{ and } f \in \mu(D_\omega) \end{cases} .$$

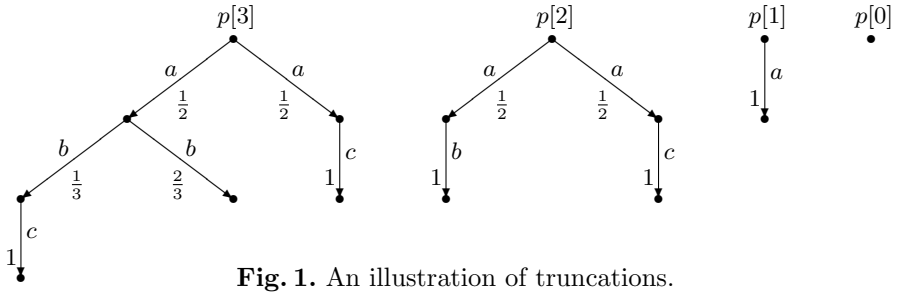

Fig. 1. An illustration of truncations.

The truncation of probabilistic distributions (and respectively of probabilistic computations, which we omit for reasons of space) satisfies the following properties useful in proofs of properties of our metric, as truncations are an integral part of its definition. These properties are, moreover, reminiscent of the properties of projection spaces of Große-Rhode and Ehrig [7].

Proposition 3.

(a) *If* $f \in \mu(D_n)$ *then* $f[k] \in \mu(D_k)$ *when* $0 \leq k \leq n$ *and* $f[k] = f_{D_k}$ *when* $k \geq n$
(b) *If* $f \in \mu(D_\omega)$ *then for all* k, m $(\ f[m]\ )[k] = f[\min\{m, k\}]$
(c) *For all* $f, g \in \mu(D_\omega)$ *and* $k \in \mathbb{N}$ $d_\mu(f[k], g[k]) \leq d_\mu(f, g)$
(d) *For all* $f, g \in \mu(D_\omega)$*if* $f[m] = g[m]$ *then* $f[k] = g[k]$ *for all* $0 \leq k \leq m$.

We now define a metric on probabilistic computations. In the non-trivial case of computations starting with the same action, the distance is set to an infinite sum of distances between the truncations of the two distributions, with each summand weighted by the depth of the truncation in inverse proportion.

Definition 4. *Let* $(D_n)_{n\in\mathbb{N}}$, D_ω *be the carrier sets defined in Definition 2. Define the metric* d *by induction on the structure of elements of* D_n *by putting* $d(p_0, p_0) = 0$, $d(p_0, (a, f)) = 1$, $d((a, f), p_0) = 1$, *and*

$$d((a, f), (\tilde{a}, g)) = \begin{cases} 1 & \text{if } a \neq \tilde{a} \\ \sum_{k=0}^{\infty} 2^{-(k+1)} d_\mu(f[k], g[k]) & \text{otherwise} \ . \end{cases}$$

Lemma 1. *Let* (D_ω, d) *be as above, then* $0 \leq d(p, q) \leq 1$ *for all* $p, q \in D_\omega$.

We now prove the following for D_ω, and simultaneously for each D_n.

Theorem 1. (D_ω, d) *is a metric space.*

Proof. 1. We show $d(p, q) = 0$ if and only if $p = q$. In the non-trivial case of $p \neq q$ the result follows by definition of d except when $p = (a, f)$ and $q = (a, g)$: since $p \neq q$ we must have $f \neq g$, and thus from d_μ being a metric and Proposition 3(a) we have that $d_\mu(f[m], g[m]) = d_\mu(f, g) \neq 0$ for $m = \min_n\{s(f), s(g) \subseteq D_n\}$, and thus $d(p, q) \neq 0$ as required.
2. $d(p, q) = d(q, p)$ by definition of d and d_μ a metric on all D_n, $n \in \mathbb{N}$.
3. The inequality $d(p, q) + d(q, r) \geq d(p, r)$ follows from Lemma 1 in all cases except $p = (a, f)$, $q = (a, g)$ and $r = (a, h)$, in which case it holds since d_μ is a metric. □

It should be noted that our metric is not an ultra-metric. An ultrametric can be defined in terms of truncations in the standard way, see [3], but it results in different convergence as demonstrated in the example below.

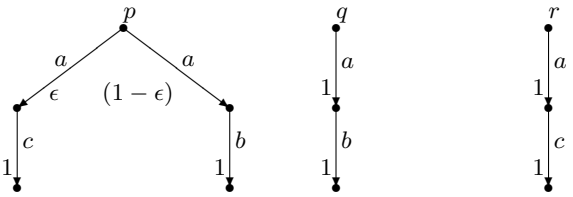

Fig. 2. 'Smooth' metric d (this paper) vs 'discrete' ultrametric of [3]

Example 1. Consider the processes in Figure 2. We have that:

$$d(p,q) = \frac{\epsilon}{2} \ , \quad d(p,r) = \frac{(1-\epsilon)}{2} \quad \text{and} \quad d(q,r) = \frac{1}{2}$$

and hence as $\epsilon \to 0$ the distance $d(p,q) \to 0$ while $d(p,r) \to \frac{1}{2}$. On the other hand, in the metric of [3], the distances between p and q, and between p and r, are $\frac{1}{2}$ for any $\epsilon \in (0,1)$.

Our metric nevertheless specialises to the metric of de Bakker & Zucker [5]. To see this consider a restriction, for each $n \in \mathbb{N}$, of the set $\mu(D_n)$ to the set of *point distributions* of D_n, i.e. the set $\{\, \eta_p \,|\, p \in D_n \,\}$ where

$$\eta_p(q) = \begin{cases} 1 \text{ if } p = q \\ 0 \text{ otherwise} \end{cases}$$

and inductively we denote $\{p_0\} \cup A \times \{\eta_p \,|\, p \in D_n^\eta\}$ by D_{n+1}^η. Intuitively, if $p = (a, \eta_q) \in D_n^\eta$ then the probability of p performing the action a and becoming q is 1, and the probability of p becoming any other process is 0. This can be compared with de Bakker and Zucker's construction of simple processes, where the elements are of the form $p = p_0$ or $p = (a, q)$, for a action and q process. We have the following.

Proposition 4. *The metric d coincides with the metric of de Bakker & Zucker on the subspace D_ω^η of D_ω, i.e. for all p, q in D_ω^η:*

$$d(p,q) = \begin{cases} 0 & \text{if } p = q \\ 2^{1-m} & \text{otherwise where } m = \min_k \{\, p[k] \neq q[k] \,\} \ . \end{cases}$$

We now apply the standard completion technique to derive the domain D of probabilistic computations as consisting of D_ω together with all limit points $p = \lim_{n\to\infty} p_n$, with $\langle p_n \rangle_n$ a Cauchy sequence in D_ω, such that $p_n \in D_n \ \forall n \in \mathbb{N}$.

Definition 5. *Define the space (D, d) of* probabilistic computations *as the metric completion of (D_ω, d).*

We show that d satisfies the required domain equation by constructing isometric embeddings. Categorical techniques of [1] have not been used as it is unclear how to define a functor to represent this construction; this is due to the fact that our metric is not defined inductively in correspondence with the inductively defined metric spaces.

Theorem 2. *D satisfies the domain equation* $D \cong \{p_0\} \cup A \times \mu(D)$.

Proof. Let $\grave{D} \stackrel{def}{=} \{p_0\} \cup A \times \mu(D)$.

1. First define $\psi : \grave{D} \to D$ by

$$\psi(\grave{p}) = \begin{cases} p_0 & \text{if } \grave{p} = p_0 \\ \lim_{n\to\infty} p_n & \text{otherwise} \end{cases}$$

   where, assuming $\grave{p} = (a, g)$ for some $a \in A$ and $g \in \mu(D)$, $p_n = (a, f_n)$ with $f_{n+1} = g[n]$ for $n \in \mathbb{N}$. This is well-defined as $p_n \in D_n$ and the sequence $(p_n)_n$ can be shown to be Cauchy with respect to d. Finally, we demonstrate that that ψ is an isometry.
2. For the opposite direction, we define the map $\phi : D \to \grave{D}$ by

$$\phi(p) = \begin{cases} p_0 & \text{if } p = p_0 \\ (a, g) & \text{otherwise} \end{cases}$$

   where, assuming wlog $p = \lim_{n\to\infty} p_n$ with $\langle p_n \rangle_n$ Cauchy, $p_n = (a, f_n)$ for some $a \in A$ and $f_n \in \mu(D_{n-1})$ for all $n \geq 1$, $g : D \to [0, 1]$ is defined by $g(q) = \lim_{n\to\infty} f_n(q)$ for $q \in D$. To show that this is well-defined, i.e. $\grave{p} \in \grave{D}$, we show $\lim_{n\to\infty} f_n(q)$ exists for all $q \in D$ and $g \in \mu(D)$, i.e. $\sum_{q\in D} g(q) = 1$ and g has countable support. Finally, we show that ϕ is an isometry. □

5 Domain Equation for Reactive Processes

Observe that the probabilistic computations (the elements of D) are represented either by p_0 (termination), or are limits $\lim_{n\to\infty} p_n$ of Cauchy sequences of (finite) computations, where the limit is of the form $(a, \lim_n f_n)$, and thus initially can only perform the action a. To allow choice it is necessary to use *sets* of elements of D as denotations for probabilitic processes. As we wish to maintain consistency with Larsen & Skou's approach, we mimic the syntactic restrictions in the semantic domain by requiring that such sets must satisfy the following *reactiveness* condition.

Definition 6. *Let $X \subseteq D_\omega$. X is said to satisfy the* reactiveness *condition if, for any $p, q \in D_\omega$ where $p = (a, f)$ and $q = (\tilde{a}, g)$, if $p, q \in X$ then it must be the case that either $a \neq \tilde{a}$ or $p = q$.*

The above guarantees, for any $a \in A$, the existence of at most one element of the form (a, f) in the set X, and so the probability of performing an a transition for any one of these sets is either 1 or 0.

To extend our metric to sets of probabilistic computations we use the Hausdorff distance. As before, we introduce a sequence of metric spaces $(P_n, d)_n$ $n \in \mathbb{N}$.

Definition 7. *Let (P_n, d) $n = 0, 1, \ldots$ be a collection of metric spaces defined inductively by*

$$P_0 = \{p_0\} \text{ and } P_{n+1} = \{p_0\} \cup \mathcal{P}_r\left(A \times \mu(P_n)\right)$$

where A is a set of actions and $\mathcal{P}_r$ denotes the powerset operator restricted to the subsets which satisfy the reactiveness condition. Put $P_\omega = \bigcup_n P_n$ and then define d on P_ω (or on any P_n where $n \in \mathbb{N}$) to be the Hausdorff distance with respect to d as defined on D_ω. Let (P, d) denote the completion of (P_ω, d).

Observe that for any $X \in P_\omega$ we have that $X \in \mathcal{P}_r(A \times \mu(P_n))$ for some $n \in \mathbb{N}$. Then for any distinct elements $p, q \in X$ such that $p = (a, f)$ and $q = (\hat{a}, g)$ with $a \neq \hat{a}$, we have by definiton of the metric d that $d(p, q) = 1$. It follows that X is closed, since the only Cauchy sequences included in X are the trivial ones. Thus, $P_\omega \subseteq \mathcal{P}_c(D_\omega)$, and from the completion techniques it follows that $P \subseteq \mathcal{P}_c(D)$, and hence d is indeed a metric on P_ω.

Moreover, Hahn's Theorem can be used in the proof of the following.

Theorem 3. *Let $\mathcal{P}_{rc}(A \times \mu(P))$ denote the closed subsets of $(A \times \mu(P))$ satisfying the reactiveness condition. Then*

$$P \cong \{p_0\} \cup \mathcal{P}_{rc}\left(A \times \mu(P)\right) \quad .$$

The theorem is proved by an adaptation of a similar result in [5] for the non-probabilistic case. We note that truncations on $f \in \mu(P_\omega)$ are defined as for D, and we define the truncation function on P inductively by putting $X[n] = \{\, p[n] \,|\, p \in X \,\}$ for any $X \subseteq A \times \mu(P)$.

6 Denotational Semantics

We have obtained P as a solution of a domain equation (assuming $A = Act$), and can now give denotational semantics for our language RP. The next step is to define the semantic operators on P.

Definition 8. *The* degree *of a process $p \in P$ is defined inductively by putting $deg(p_0) = 0$, $deg(p) = n$ if $p \in P_n \setminus P_{n-1}$ for some $n \geq 1$, and $deg(p) = \infty$ otherwise. We then say a process p is* finite *if $deg(p) = n$ for some $n \in \mathbb{N}$ and* infinite *otherwise.*

Thus, each $p \in P$ is either finite, or it is infinite, in which case $p = \lim p_n$, $(p_n)_n$ Cauchy, with each p_n of degree n. We now define the operators "$\cup$" and "$||$" on P to model deterministic choice and synchronous parallel; this is achieved by first defining the operators on finite processes and then extending the definition to limits of Cauchy sequences.

Definition 9. *Let $p \in P$, $X, Y \in \mathcal{P}_{rc}(A \times \mu(P))$ with finite degree, $(p_i)_i, (q_i)_i$ Cauchy sequences of finite processes.*
(a) (union) Put $p \cup p_0 = p_0 \cup p = p$, $X \cup Y$ is the set theoretic union of X and Y, and define $(\lim_i p_i) \cup (\lim_j q_j) = \lim_k (p_k \cup q_k)$.
(b) (parallel) Put $p||p_0 = p_0||p = p$, and define

$$X||Y = \begin{cases} \{x||y \,|\, x \in X,\ y \in Y \ \&\ d(x,y) < 1\} & \text{if there exists } x \in X \text{ and } y \in Y \\ & \text{such that } d(x,y) < 1 \\ p_0 & \text{otherwise} \end{cases}$$

where for $x = (a, f)$ and $y = (a, g)$ put $x||y \stackrel{def}{=} (a, f||g)$ with

$$(f||g)(p) \stackrel{def}{=} \begin{cases} f(p_1)g(p_2) & \text{if } p = p_1||p_2 \\ 0 & \text{otherwise} \end{cases}$$

for any $p \in P$, and define $(\lim p_i)||(\lim q_j) \stackrel{def}{=} \lim_k (p_k||q_k)$.

Lemma 2. *For all X, $\tilde{X}$ and $Y \in P$ with finite degree*

$$d(X \cup Y, \tilde{X} \cup Y) \leq d(X, \tilde{X}) \quad \text{and} \quad d(X||Y, \tilde{X}||Y) \leq d(X, \tilde{X}) \ .$$

Theorem 4. *$\cup$ and $||$ are well defined and continuous operators on P subject to the restriction that $X \cup Y$ satisfies the reactiveness condition.*

Recall that $\mathcal{E}$ denotes the (guarded) expression with free variables, while RP is the set of closed (guarded) expressions. As usual, in order to handle the variables x of $\mathcal{E}$, we introduce the semantic map $\mathcal{M} : \mathcal{E} \to (E \to P)$ parametrised by environments E, ranged over by ρ, defined by $E = \mathcal{X} \to P$. In addition, we shall require an auxiliary function $\Phi : ([0,1] \times P)^\infty \to (P \to [0, \infty))$, defined as follows: for any $p = \langle (\mu_i)_i, (p_i)_i \rangle_{i \in I} \in ([0,1] \times P)^\infty$, $\Phi(p) = f_p$ where for any $q \in P$

$$f_p(q) = \begin{cases} 0 & \text{if } q \neq p_i \text{ for all } i \in I \\ \sum_{j \in J} \mu_j & \text{otherwise where } J = \{\, j \,|\, j \in I \text{ and } q = p_j \,\} \ . \end{cases}$$

We now define denotational metric semantics for RP expressions $\mathcal{E}$. Recursive processes are defined as limits of Cauchy chains of unfoldings of the map $\mathcal{M}$.

Definition 10 ((Denotational semantics)). *Define $\mathcal{M} : \mathcal{E} \to (E \to P)$ inductively on the structure of elements of $\mathcal{E}$ as follows:*

$$\begin{aligned} \mathcal{M}(\underline{0})(\rho) &= \{p_0\} \\ \mathcal{M}(\textstyle\sum_{i \in I} a_{\mu_i}.p_i)(\rho) &= \{(a, \Phi(\langle (\mu_i)_i, (\,\mathcal{M}(p_i)(\rho)\,)_i \rangle_{i \in I}))\} \\ \mathcal{M}(p_1 \oplus p_2)(\rho) &= \mathcal{M}(p_1)(\rho) \cup \mathcal{M}(p_2)(\rho) \\ \mathcal{M}(p_1||p_2)(\rho) &= \mathcal{M}(p_1)(\rho)||\mathcal{M}(p_2)(\rho) \\ \mathcal{M}(fix\ x.p)(\rho) &= \lim_{k \to \infty} \mathcal{M}^k(p)(\rho) \end{aligned}$$

where $\mathcal{M}^0(p)(\rho) = p_0$ and $\mathcal{M}^{k+1}(p)(\rho) = \mathcal{M}(p)(\rho\{\mathcal{M}^k(p)(\rho)/x\})[k+1]$.

The well-definedness of the semantic map follows from the lemma below.

Lemma 3. *Let $fix\ x.p \in \mathcal{E}$, and let the sequence q_k denote $\mathcal{M}^k(p)(\rho)$, $k \in \mathbb{N}$. Then $q_{k+1}[k] = q_k$ for all $k \in \mathbb{N}$.*

7 Full Abstraction

Finally, we obtain that P is a fully abstract model of the language RP with respect to probabilistic bisimulation. The result follows from Lemma 4 below.

Lemma 4. *For all $a \in A$ and $p \in \mathrm{RP}$:*

1. *$p \xrightarrow{a}$ if and only if there exists $(a, f) \in \mathcal{M}(p)$.*
2. *For any $q \in \mathrm{RP}$ if $S_{\mathcal{M}(q)} = \{\tilde{q} \mid \tilde{q} \in \mathrm{RP}$ and $\mathcal{M}(\tilde{q}) = \mathcal{M}(q)\}$ then we have $p \xrightarrow{a}_{\mu} S_{\mathcal{M}(q)}$ if and only if $f(\mathcal{M}(q)) \geq \mu$.*
3. *If $f(r) > 0$ for some $r \in P$ then if $S_r = \{q \mid q \in \mathrm{RP}$ and $\mathcal{M}(q) = r\}$ then $p \xrightarrow{a}_{\mu} S_q$ if and only if $f(q) \geq \mu$.*

Theorem 5. *Let $\mathcal{M} : \mathcal{E} \to (E \to P)$ be the semantic map of Definition 10. Then for all $p, q \in \mathrm{RP}$,*

$$p \sim q \text{ if and only if } \mathcal{M}(p) = \mathcal{M}(q) \ .$$

8 Conclusions and Further Work

We have derived a metric space model for a probabilistic extension of a process calculus, which can be further extended with an asynchronous concurrency operator by following, to a large extent, the techniques introduced by de Bakker & Zucker [5].

Although the continuity of prefixing (and also of the asynchronous concurrency operator) fails in our model, our metric is 'smooth' (as apposed to the 'discrete' metric of [3]), and hence closer in spirit to the probabilistic powerdomain construction. It remains to be seen if a suitable combination of our metric and the standard metric which yields continuity can be found. Finally, we intend to consider the addition of non-deterministic choice and apply our results to existing probabilistic process calculi, e.g. PCCS [6].

Acknowledgements: We would like to thank Achim Jung, Michael Huth, Christel Baier and Reinhold Heckmann for discussions and suggestions.

References

1. P.H.M.America and J.J.M.M.Rutten. Solving reflexive domain equations in a category of complete metric spaces, *JCSS, 39, no.3, 1989.*
2. J.C.M.Baeten, J.A.Bergstra and S.A.Smolka. Axiomatising probabilistic processes: ACP with generative probability, *Proc. Concur'92, LNCS, 630, Springer, 1992.*
3. C.Baier and M.Kwiatkowska. Domain equations for probabilistic processes, *preprint.*
4. I.Christoff. Testing equivalences and fully abstract models for probabilistic processes, *Proc. Concur'90, LNCS, 458, Springer, 1990.*
5. J.W.de Bakker and J.I.Zucker. Processes and the denotational semantics of concurrency, *Information and Control, 1/2, 1984.*

6. A.Giacalone, C.-C.Jou and S.A.Smolka. Algebraic reasoning for probabilistic concurrent systems, *In Proc. Programming Concepts and Methods, IFIP, 1990.*
7. M.Große-Rhode and H.Ehrig. Transformation of combined data type and process specifications using projection algebras, *LNCS, 430, Springer, 1989.*
8. C.A.Hoare. Communicating sequential processes, *Prentice Hall, 1985.*
9. C.Jones. Probabilistic non-determinism, *PhD Thesis, University of Edinburgh, 1990.*
10. B.Jonsson and K.G.Larsen. Specification and refinement of probabilistic processes, *Proc. IEEE Logic in Computer Science (LICS), 1991.*
11. B.Jonsson and Wang Yi. Compositional testing preorders for probabilistic processes, *Proc. IEEE Logic in Computer Science (LICS), 1995.*
12. C.-C.Jou and S.Smolka. Equivalences, congruences and complete axiomatizations for probabilistic processes, *Proc. Concur'90, LNCS, 458, Springer, 1990.*
13. D.Kozen. Semantics of probabilistic programs, *Proc. IEEE Symposium on Foundations of Computer Science (FOCS), 1979.*
14. K.G.Larsen and A.Skou. Bisimulation through probabilistic testing, *Information and Computation, 94, 1991.*
15. K.G.Larsen and A.Skou. Compositional verification of probabilistic processes, *Proc. Concur'92, LNCS, 630, Springer, 1992.*
16. R.Milner. Calculi for synchrony and asynchrony, *TCS, 25(3), 1983.*
17. R.Milner. Communication and concurrency, *Prentice Hall, 1989.*
18. K.Seidel. Probabilistic communicating processes, *TCS, 152, 1995.*
19. C.Tofts. A synchronous calculus of relative frequency, *Proc. Concur'90, LNCS, 458, Springer, 1990.*
20. R.J.van Glabbeek, S.A.Smolka, B.Steffen and C.Tofts. Reactive, generative and stratified models of probabilistic processes, *Proc. Concur'92, LNCS, 630, Springer, 1992.*
21. S.Yuen, R.Cleaveland, Z.Dayar and S.A.Smolka. Fully abstract characterizations of testing preorders for probabilistic processes, *Proc. Concur'94, LNCS, 836, Springer, 1994.*