The powerdomain of indexed valuations

Daniele Varacca
BRICS[*]*- University of Aarhus, Denmark*
Computer Laboratory - University of Cambridge, UK
varacca@brics.dk

Abstract

This paper is about combining nondeterminism and probabilities. We study this phenomenon from a domain theoretic point of view. In domain theory, nondeterminism is modeled using the notion of powerdomain, while probability is modeled using the powerdomain of valuations. Those two functors do not combine well, as they are. We define the notion of powerdomain of indexed valuations, which can be combined nicely with the usual nondeterministic powerdomain. We show an equational characterization of our construction. Finally we discuss the computational meaning of indexed valuations, and we show how they can be used, by giving a denotational semantics of a simple imperative language.

1 Introduction

Nondeterminism and probability are computational effects whose semantics has been thoroughly studied. The combination of the two appears to be essential in giving models for concurrent processes [13, 3, 10]. Denotationally, nondeterminism is handled by the notion of powerdomain functor in a suitable category of domains [9], while probabilistic behavior is handled by the powerdomain of valuations [4, 6]. They happen to be monads, thus fitting the general idea, introduced by Moggi, of monads as models for computational effects [8].

A general way for combining two monads is by defining a *distributive law* [2]. Suppose we have two monads (T, η^T, μ^T), (S, η^S, μ^S) on some category. A *distributive law* of S over T is a natural transformation $d : ST \stackrel{\cdot}{\longrightarrow} TS$ satisfying the following axioms:

- $d \circ \eta^S T = T\eta^S$
- $d \circ S\eta^T = \eta^T S$
- $d \circ \mu^S T = T\mu^S \circ dS \circ Sd$
- $d \circ S\mu^T = \mu^T S \circ Td \circ dT$

With a distributive law we can define a monad on the functor TS: if $d : ST \stackrel{\cdot}{\longrightarrow} TS$ is a distributive law, then $\big(TS, \eta^T\eta^S, (\mu^T\mu^S) \circ TdS\big)$ is a monad.

$$TSTS \xrightarrow[TdS]{\cdot} TTSS \xrightarrow[\mu^T\mu^S]{\cdot} TS$$

Now, let (P, η^P, μ^P) be the finite nonempty powerset monad, and (V, η^V, μ^V) be the finite probability distribution monad in the category $\mathbf{SET}$. If we want to build some kind of "combined" monad, we have to face the following obstacle:

Proposition 1.1. [1] *There is no distributive law of V over P, nor of P over V.*

The same is true for the corresponding monads in the category $\mathbf{CONT}$ of continuous domains. If $\mathcal{P}$ is some powerdomain monad and $\mathcal{V}$ is the powerdomain of valuations monad, then there is no distributive law between them.

Let us look at the reason for this phenomenon. The monads above arise as free algebras for suitable equational theories. In such cases a categorical distributive law amounts to an equational distributive law between the operators of the theories. In our case we would like to distribute the probabilistic choice over the nondeterministic one. Assume that $+_p$ is a probabilistic choice operator: $A +_p B$ is choosing A with probability p and B with probability $(1-p)$. This operator satisfies $A +_p A = A$. Now if A, B, C are sets, we would like that:

$$A +_p (B \cup C) = (A +_p B) \cup (A +_p C)\ .$$

But if this law has to be true, then

$$\{x, y\} = \{x, y\} +_p \{x, y\} = \{x, y, x +_p y, y +_p x\}$$

which is a contradiction.

[*] Basic Research in Computer Science (www.brics.dk), funded by the Danish National Research Foundation.

[1] Personal communication by Gordon Plotkin

To overcome this difficulty, Tix [11] and Mislove [7], independently, define the notion of geometrically convex powerdomain $\mathcal{P}_{TM}$, which cannot be applied to arbitrary continuous domains, but to continuous d-cones only. Morally, $\mathcal{P}_{TM}(X)$ is the set of all *convex* subsets of X. The composition $\mathcal{P}_{TM} \circ \mathcal{V}$ is a monad.

Our solution amounts to defining the notion of *indexed valuation* functor $\mathcal{IV}$. Indexed valuations are similar to usual valuations but they do not satisfy $A +_p A = A$. It is possible to define a distributive law between the usual $\mathcal{P}$ and $\mathcal{IV}$. We perform this construction both in the category of sets and in the category of continuous domains.

As in the case of $\mathcal{P}$ and $\mathcal{V}$, the functor $\mathcal{IV}$ is freely generated by an (in)equational theory.

Besides their categorical justification, indexed valuations have a computational meaning, which we present by giving semantics to an imperative language containing both random assignment and nondeterministic choice. The operational semantics is given in terms of alternating transition systems. Such systems come equipped with a notion of scheduler for resolving the nondeterminism. In the literature there are two notions of scheduler: deterministic and probabilistic. Using indexed valuations, we give a denotational semantics which is adequate with respect to deterministic schedulers. A semantics in terms of the Tix-Mislove construction, instead, is adequate with respect to probabilistic schedulers.

2 Indexed valuations in the category of sets

We perform our construction in the category of sets and functions. We define the indexed valuation functor and the distributive law. The notions introduced in this section will also be necessary when we work in the category of continuous domains.

2.1 Definition

We need some preliminary definitions.

Definition 2.1. A *discrete valuation* (DV) on a set X is a function

$$\nu : X \to \overline{\mathbb{R}^+} := [0, +\infty] .$$

Given a discrete valuation ν on X, we define its *norm* $|\nu|$ as

$$|\nu| := \sum_{x \in X} \nu(x) := \sup_{F \subseteq_{fin} X} \sum_{x \in F} \nu(x) .$$

For $Y \subseteq X$ we also put

$$\nu(Y) := \sum_{y \in Y} \nu(y) .$$

The support $Supp(\nu)$ of a DV ν is defined as

$$Supp(\nu) := \nu^{-1}(]0, +\infty]) = \{x \in X \mid \nu(x) > 0\} .$$

As far as I know this notion, although straightforward, does not have a standard name in the literature. The name "discrete valuations" comes from the fact that they are valuations on the discrete topology (see section 3.1). In [5] discrete valuations taking values in $[0, +\infty[$ are called *weightings*.

We denote the set of discrete valuations on X by $V(X)$.

Definition 2.2. A discrete valuation ν is a *discrete partial probability distribution* if $|\nu| \leq 1$.

Partial probability distributions are sufficient for semantics of probabilistic processes. Nevertheless, we choose to deal with the more general notion of valuation, because the corresponding equational theory is nicer while the other fundamental properties are the same [6].

We now introduce the main new concept.

Definition 2.3. Let X be a set. An *indexed discrete valuation* (IDV) on X is a pair $(Ind, Prob)$ where $Ind : I \to X$ is a function and $Prob$ is a discrete valuation on I, for some set I.

Notice that we do not require that Ind be injective. This is indeed the main point of this construction: we want to divide the probability of an element among its indexes. One possible interpretation is that indexes in I represent computations, while elements of X represent observations.

We shall also write x_i for $Ind(i)$ and p_i for $Prob(i)$. An indexed discrete valuation $\xi := (Ind, Prob)$ will also be denoted as $(x_i, p_i)_{i \in I}$.

We are now going to define an equivalence relation on the class of IDV's. It is the transitive closure of two simpler equivalence relations.

Definition 2.4. We set

$$(x_i, p_i)_{i \in I} \sim_1 (y_j, q_j)_{j \in J}$$

if and only if there exists a bijection $h : I \to J$ such that

$$\forall i \in I.\, y_{h(i)} = x_i \, ,$$
$$\forall i \in I.\, q_{h(i)} = p_i \, .$$

This says that two IDV's are equivalent up to renaming of the indexes. If we interpret indexes as computations, we may say that we do not care about the identity of a single computation. We only care of how many different computations there are, and how they relate to observations.

Given an IDV $(x_i, p_i)_{i \in I}$, let $I_0 := \{i \in I \mid p_i = 0\}$.

Definition 2.5. We set

$$(x_i, p_i)_{i \in I} \sim_2 (y_j, q_j)_{j \in J}$$

if and only if

$$I \setminus I_0 = J \setminus J_0 \, ,$$
$$\forall i \in I \setminus I_0.\, x_i = y_i \; \& \; p_i = q_i \, .$$

This says that only indexes in the support matter. Intuitively, computations with probability 0 do not happen, so we may as well ignore them.

Definition 2.6. The equivalence relation $\sim$ is the transitive closure of $\sim_1 \circ \sim_2$.

From now on we will use the term "indexed discrete valuations" to denote equivalence classes under $\sim$.

Given a set X and an infinite cardinal number α we define the set $IV_\alpha(X)$ as follows:

$$IV_\alpha(X) := \{[(x_i, p_i)_{i\in I}]_\sim \mid Card(I) < \alpha\} .$$

In particular $IV_{\aleph_0}(X)$ is the set of indexed discrete valuations whose indexing set is finite. Its elements are called indexed finite valuations. For simplicity, we shall write IV for $IV_{\aleph_0}$.

The construction above can be extended to a functor $IV : \mathbf{SET} \to \mathbf{SET}$ as follows. If $f : X \to Y$ then

$$IV(f)([(x_i, p_i)_{i\in I}]_\sim) := [(f(x_i), p_i)_{i\in I}]_\sim .$$

It is easy to check that this construction is well defined (i.e. does not depend on the representative). The functor IV extends to a monad, with the following unit and multiplication (we drop the mention of equivalence classes to simplify the reading):

$$\eta_X : X \to IV(X) ,$$

$$\eta_X(x) := (x, 1)_{*\in\{*\}} ;$$

$$\mu_X : IV(IV(X)) \to IV(X) ,$$

$$\mu_X((x_{i_\lambda}, p_{i_\lambda})_{i_\lambda\in I_\lambda}, \pi_\lambda)_{\lambda\in\Lambda} := (x_j, q_j)_{j\in J}$$

where

$$J = \biguplus_{\lambda\in\Lambda} I_\lambda , \quad q_j = p_j\pi_\lambda \text{ if } j \in I_\lambda .$$

To simplify the definition of μ, recall that an IDV is in fact an equivalence class. We can therefore assume that $I_\lambda = I$ for every $\lambda \in \Lambda$ because we can always reindex and add indexes with probability 0. Therefore

$$((x_{i_\lambda}, p_{i_\lambda})_{i_\lambda\in I_\lambda}, \pi_\lambda)_{\lambda\in\Lambda} \sim ((x_i^\lambda, p_i^\lambda)_{i\in I}, \pi_\lambda)_{\lambda\in\Lambda} .$$

And

$$\mu_X(((x_i^\lambda, p_i^\lambda)_{i\in I}, \pi_\lambda)_{\lambda\in\Lambda}) := (x_i^\lambda, \pi_\lambda p_i^\lambda)_{(i,\lambda)\in I\times\Lambda} .$$

Proposition 2.7. *The triple (IV, η, μ) defined above is a monad.*

We define a distributive law of indexed valuations over powerset.

Theorem 2.8. *Let $IV : \mathbf{SET} \to \mathbf{SET}$ be as above, and $P : \mathbf{SET} \to \mathbf{SET}$ be the covariant nonempty finite powerset monad. Then*

$$d : IV \circ P \to P \circ IV$$

$$d_X(S_i, p_i)_{i\in I} := \{(h(i), p_i)_{i\in I} \mid h : I \to X,\ h(i) \in S_i\}$$

is a distributive law.

It then follows that the functor $P \circ IV$ is endowed with a monad structure.

2.2 Equational characterization

We now define two operations on indexed discrete valuations, which will allow us to characterize IV as a free algebra.

Definition 2.9. Let $\nu := (Ind, Prob) = (x_i, p_i)_{i\in I}$ and $\xi := (Ind', Prob') = (y_j, q_j)_{j\in J}$ be IDV's on X. Assume that $I \cap J = \emptyset$ (this is not restrictive, because we can always reindex).

We define $\nu \oplus \xi$ to be $(Ind \cup Ind', Prob \cup Prob')$. For $p \in \overline{\mathbb{R}^+}$ we define $p\nu$ to be $(x_i, pp_i)_{i\in I}$. With $\underline{0}$ we denote the IDV whose indexing set is empty.

Note, in particular, that $p\nu \oplus (1-p)\nu \not\sim \nu$.

Consider the following equational theory:

1. $A \oplus B = B \oplus A$;
2. $A \oplus (B \oplus C) = (A \oplus B) \oplus C$;
3. $A \oplus \underline{0} = A$;
4. $0A = \underline{0}$;
5. $1A = A$;
6. $p(A \oplus B) = pA \oplus pB \quad p \in \overline{\mathbb{R}^+}$;
7. $p(qA) = (pq)A \quad p, q \in \overline{\mathbb{R}^+}$;

These axioms are almost the ones defining a *real cone* [11]. The only difference is that we drop the axiom $(p + q)A = (pA \oplus qA)$.

Proposition 2.10. *The indexed finite valuations monad is the free algebra monad for the theory (1)–(7).*

The finite nonempty powerset is the free algebra for the following theory:

8. $A \uplus B = B \uplus A$;
9. $A \uplus (B \uplus C) = (A \uplus B) \uplus C$;
10. $A \uplus A = A$;

The functorial composition of the two monads, together with the described distributive law gives us the free algebra over the theory (1)–(10) augmented with the following axioms.

11. $p(A \uplus B) = pA \uplus pB$.

12. $A \oplus (B \uplus C) = (A \oplus B) \uplus (A \oplus C)$

Notice that (11)–(12) express equationally that the probabilistic operators distribute over the nondeterministic one.

If we see finite indexed valuations and finite sets as (equivalence classes of) terms we can give a syntactic interpretation of the categorical distributive law: it takes a term where there are no probabilistic operators inside a nondeterministic one and transforms it into a term where all the nondeterministic operators have been pushed outside. In other words we interpret the equations (11)–(12) as rewriting rules, from left to right. Formally:

Theorem 2.11. *The monad on $P \circ IV$ obtained via the categorical distributive law defined above is the free algebra for the equational theory (1)–(12).*

3 Indexed valuations in the category of continuous domains

3.1 Definition

To give a semantics to languages with recursion we work in a category of domains. We shall recall briefly the domain theoretic notions we need. For a more detailed treatment, the suggested reference is [1].

In every DCPO an *approximation* relation $\ll$ is definable (also known as *way-below* relation). We say that $x \ll y$ if $y \sqsubseteq \bigsqcup^{\uparrow} Z \Longrightarrow \exists z \in Z. x \sqsubseteq z$. A subset B of a DCPO D is a *basis* if every $x \in D$ is the directed lub of the elements of B that are way-below x. A DCPO with a basis is called a *continuous domain*.

The properties of the way-below relation on B suggest the following definition.

Definition 3.1. A relation $<$ on a set X is an *AB-relation* if it is transitive and satisfies the finite interpolation property: for every $F \subseteq_{fin} X$ and for every $x \in X$, $F < x \Longrightarrow \exists y \in X.\ F < y < x$. The structure $(X, <)$ is called an *abstract basis*.

A preorder is also an abstract basis. For AB-relations we shall use the same terminology as for preorders. We therefore speak of monotonic functions, lower sets, directed sets, and so on. In particular we recall that an *ideal* is a lower directed set. The set of ideals of X is called $Idl(X)$. For any $x \in X$ the set $\iota(x) := x\!\downarrow\ := \{y \mid y < x\}$ is an ideal. The structure $(Idl(X), \subseteq)$ is a continuous domain with basis $\iota(X)$. It is called the *ideal completion* of X. Conversely if D is a continuous domain with basis B, then $(B, \ll)$ is an abstract basis whose ideal completion is isomorphic to D.

Let $(X, <)$ be an abstract basis, $(D, \sqsubseteq)$ be a (not necessarily continuous) DCPO, and $f : X \to D$ be a function mapping $<$ to $\sqsubseteq$ ("monotonic"). The function $f^{\sharp} : Idl(X) \to D$ defined as follows

$$f^{\sharp}(I) = \bigsqcup_{x \in I}^{\uparrow} f(x)$$

is continuous. Notice that in general $f^{\sharp}(\iota(x)) \neq f(x)$. If $(X, <)$, $(Y, <)$ are two abstract bases and if $f : X \to Y$ is a monotonic function, we define the *extension* $\overline{f}$ to be the continuous function $(\iota \circ f)^{\sharp} : Idl(X) \to Idl(Y)$.

We conclude this quick introductory part with the definition of the powerdomain of valuations.

Definition 3.2. A *continuous valuation* on a topological space (X, τ) is a function $\nu : \tau \to \overline{\mathbb{R}^+}$ satisfying:

- (Strictness)
  $\nu(\emptyset) = 0$;
- (Monotonicity)
  $U \subseteq V \Longrightarrow \nu(U) \leq \nu(V)$;
- (Modularity)
  $\nu(U) + \nu(V) = \nu(U \cup V) + \nu(U \cap V)$;
- (Continuity) whenever $\mathcal{J}$ is a directed subset of τ

$$\nu\Big(\bigcup \mathcal{J}\Big) = \sup_{U \in \mathcal{J}} \nu(U)$$

For each $x \in X$, the function η_x such that

$$\eta_x(U) = \begin{cases} 1 & x \in U \\ 0 & x \notin U \end{cases}$$

is a continuous valuation and it is called *point* valuation. A *simple* valuation is a linear combination of point valuations (the addition and scalar multiplication being defined pointwise). A continuous valuation on a DCPO D is a continuous valuation on its Scott topology. The set $\mathcal{V}(D)$ of continuous valuations on D ordered pointwise is again a DCPO. If D is a continuous domain, $\mathcal{V}(D)$ is a continuous domain with basis the set of simple valuations.

Now we introduce the new concepts. If $(X, <)$ is an abstract basis then we can define a relation $\prec$ on $IV(X)$ in such a way that $(IV(X), \prec)$ is an abstract basis:

Definition 3.3. For $(x_i, p_i)_{i \in I}, (y_j, q_j)_{j \in J} \in IV(X)$

$$(x_i, p_i)_{i \in I} \prec (y_j, q_j)_{j \in J}$$

if and only if there exists a partial surjective function $f : J \to I$, s.t.:

$$x_{f(j)} < y_j \ ,$$

$$p_i < \sum_{f(j)=i} q_j \ \text{ or } \ p_i = 0.$$

There are three reasons for this choice. Firstly, this definition is an "indexed" version of the splitting lemma of [4]. Secondly, it has an interesting computational interpretation. To explain this we need the notion of scheduler for a probabilistic-nondeterministic operational model, which we shall present in section 4.1. Finally, this definition corresponds to an inequational theory that allows us to match equational and categorical distributive laws, as we did in the category of sets.

Proposition 3.4. *$(IV(X), \prec)$ is an abstract basis.*

Definition 3.5. If D is a continuous domain with basis B, let $\mathcal{IV}(D)$ be the ideal completion of $(IV(B), \prec)$. Its elements are called *indexed valuations*.

For every indexed finite valuation $\nu := (b_i, p_i)_{i \in I}$ over B, we can define a simple valuation ν^* as follows. Consider $X := \{b_i \,|\, i \in I\}$. Then define $\nu^* := \sum_{i \in I} p_i \eta_{b_i}$. For every open subset U of D we define $\nu(U) := \nu^*(U)$

Every indexed valuation $\nu \in \mathcal{IV}(D)$ defines a continuous valuation ν^*: if L is a directed set of indexed finite valuations, and if $\nu = \bigsqcup^{\uparrow} L$, then for every open $O \subseteq D$, we define $\nu^*(O) := \sup_{\nu' \in L} \nu'(O)$.

Moreover, all valuations can be obtained this way.

Proposition 3.6. *[12] Let $\mathcal{V}$ be Jones' powerdomain of valuations functor. The function $(-)^* : \mathcal{IV}(D) \to \mathcal{V}(D)$ is a continuous surjection.*

3.2 Equational characterization

We are going to characterize indexed valuations as a free construction. Recall the equational theory (1)–(7) of section 2.2. We add only one more axiom, which corresponds to definition 3.3 of the AB-relation on indexed finite valuations.

- HV: $(p + q)A \sqsubseteq (pA \oplus qA)$

Definition 3.7. A *continuous quasi-cone* is a structure $(D, \sqsubseteq, \oplus, \odot)$ such that

- $(D, \sqsubseteq)$ is a continuous domain;
- $\oplus : D \times D \to D$ is continuous;
- $\odot : [0, +\infty] \times D \to D$ is continuous[2];
- axioms (1)–(7) + (HV) are satisfied.

[2]The symbol $\odot$ is used here for clarity. Everywhere else the scalar multiplication is denoted simply by juxtaposition of its arguments

Let $\mathbf{CONT}$ be the category of continuous domains, and $\mathbf{QCONT}$ be the category of continuous quasi-cones and continuous homomorphisms.

Proposition 3.8. *The operator $\mathcal{IV}$ defined in the previous section extends to a functor $\mathbf{CONT} \to \mathbf{QCONT}$ which is left adjoint to the forgetful functor.*

Omitting the mention of the forgetful functor, we can say that $\mathcal{IV}$ is a monad in $\mathbf{CONT}$. The unit $\eta^{\mathcal{IV}}_D : D \to \mathcal{IV}(D)$ is defined as $\eta^{\mathcal{IV}}_D(d) = \{(b, p)_{* \in \{*\}} \,|\, b \ll d, \, p < 1\}$. The multiplication $\mu^{\mathcal{IV}}_D : \mathcal{IV}^2(D) \to \mathcal{IV}(D)$ is just the extension of $\mu^{IV}_B : IV^2(B) \to IV(B)$ defined as in the category **SET**.

3.3 A distributive law (categorically)

Given a continuous domain D with basis B, consider the set $P(B)$ of nonempty finite subsets of B, endowed with the Hoare order:

$$X \prec Y \iff \forall x \in X. \exists y \in Y. x \ll y \ .$$

It is known that $(P(B), \prec)$ is a basis for the Hoare powerdomain $\mathcal{P}_H(D)$.

With the usual abuse of notation, we say that $\mathcal{P}_H$ has a monad structure in the category $\mathbf{CONT}$ (in the sequel we write $\mathcal{P}$ for $\mathcal{P}_H$).

To define a distributive law we need to give a family α_D of continuous functions $\mathcal{IV} \circ \mathcal{P}(D) \to \mathcal{P} \circ \mathcal{IV}(D)$. Remember that these functors are defined as ideal completions of some abstract bases. Our approach is to define a (monotonic) function between the bases and take the extension as our candidate. Consider the function $a_B : IV(P(B)) \to P(IV(B))$:

$$a_B\big((S_i, p_i)_{i \in I}\big) = \big\{(h(i), p_i)_{i \in I} \,|\, h : I \to B, \ h(i) \in S_i\big\}$$

We define α_D to be the extension of a_B.

Theorem 3.9. *The natural transformation $\alpha : \mathcal{IV} \circ \mathcal{P} \stackrel{\cdot}{\longrightarrow} \mathcal{P} \circ \mathcal{IV}$ defined above is a distributive law.*

See the appendix for a sketch of the proof.

We therefore have a monad structure for the functor $\mathcal{P} \circ \mathcal{IV}$.

3.4 A distributive law (equationally)

Following [1] we introduce the notion of *Domain-algebra*. Let $\mathcal{E}$ be an inequational theory on some signature. A continuous domain-algebra for $\mathcal{E}$ is a continuous domain together with a Scott-continuous operation for every symbol in the signature satisfying the inequalities in $\mathcal{E}$.

The continuous quasi-cones defined above can be seen as domain-algebras. We have to see the scalar multiplication as a collection of unary operations, one for each positive real (and $+\infty$), and to require explicitly that the scalar multiplication be continuous in the first argument. (Another possibility would be to define multi-sorted domain algebras.)

An interesting theory on the signature $\{\uplus\}$ is the theory (8)–(10) augmented with:

- HP: $A \sqsubseteq A \uplus B$;

Its continuous domain-algebras are also called continuous join semi-lattices. It is known that $\mathcal{P}_H$ is the free continuous join semi-lattice. Similarly to what we have observed in the category **SET**, we have the following result.

Theorem 3.10. *The monad on $\mathcal{P} \circ \mathcal{IV}$ obtained from the distributive law is the free continuous domain-algebra over the theory (1)–(12) + (HV) + (HP), with the extra requirement that the scalar multiplication be continuous in the first argument.*

If instead of the axiom (HV) we had added the more standard $(p + q)A = (pA \oplus qA)$, the resulting free construction would be the one of Tix and Mislove, which still includes the equational distributive laws, but without any corresponding categorical distributive law.

4 Semantics of programs

We give an example of how to use the above construction by giving a semantics to a simple imperative language with probabilistic and nondeterministic primitives. First we introduce the operational model. We then introduce the language. We give operational and denotational semantics to the language, and we state an adequacy theorem. Finally we compare briefly the computational meaning of our semantics with the one obtained using Tix-Mislove.

In the sequel we shall write $(x_i, p_i)_{i\in I}$ also to denote $\iota\big((x_i, p_i)_{i\in I}\big) \in \mathcal{IV}(X)$. We will use this notation even when I is not finite, to denote the lub of all its finite "truncations".

4.1 The operational model

Our operational model is a version of the probabilistic transition systems of [5]. Our presentation is inspired by the alternating model of [3].

Definition 4.1. A probabilistic (unlabeled) synchronisation tree (PST) on a set of "states" S is a structure $(Nodes, Arcs, label_N, label_P)$ where

- $(Nodes, Arcs)$ is a tree;
- $Nodes = N \uplus P$: there are *Nondeterministic* nodes and *Probabilistic* nodes;
- $Arcs \subseteq (N \times P) \uplus (P \times N)$: an arc connects only nodes of different kind;
- $label_N : N \to S$: nondeterministic nodes are labeled by states;
- $label_P : Arcs \cap (P \times N) \to \overline{\mathbb{R}^+}$: arcs leaving probabilistic nodes are labeled with probabilities;

Definition 4.2. A deterministic scheduler $\mathcal{S}$ for a PST is a function $f_\mathcal{S} : N \to P$ such that if $f_\mathcal{S}(n) = p$ then $(n, p) \in Arcs$.

A deterministic[3] scheduler chooses for every nondeterministic node one of its probabilistic sons.

Definition 4.3. A probabilistic scheduler $\mathcal{PS}$ for a PST is a function $f_{\mathcal{PS}} : N \to P \to \overline{\mathbb{R}^+}$ such that if $f_{\mathcal{PS}}(n)(p) \neq 0$ then $(n, p) \in Arcs$.

A probabilistic scheduler chooses for every deterministic node a discrete valuation over its sons. Notice that a deterministic scheduler can always be seen as a probabilistic scheduler.

The effect of a scheduler is to give a fully probabilistic model. A probabilistic scheduler is, in practice, labeling with probabilities the arcs going out of a nondeterministic node. Once every arc is endowed with probabilities we can talk of the probability of a path as the (possibly infinite) product of the probabilities of all its components.

Using probabilistic automata and schedulers we can give another motivation for our axiom (HV) and corresponding definition 3.3, which imply

$$\delta_x \sqsubseteq \tfrac{1}{2}\delta_x + \tfrac{1}{2}\delta_x$$

Consider the example in figure 1. The figure represents two

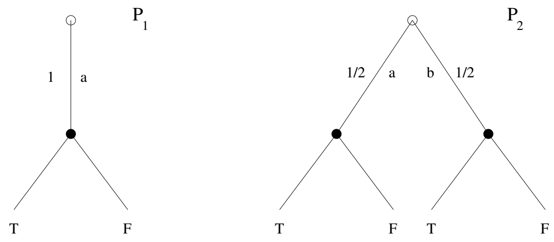

Figure 1. Processes

processes as probabilistic trees. The process P_1 allows two

[3]This is the standard term used in literature. It is not a fortunate choice because what we mean here is absence of probabilities. A better term could be *certain*.

different ways of resolving the nondeterminism. The corresponding probability distributions are δ_T and δ_F. The process P_2 allows four different ways of resolving the nondeterminism, two of which give the probability distribution $\frac{1}{2}\delta_T + \frac{1}{2}\delta_F$. The process P_2 offers more opportunities, so in a Hoare fashion, we consider it better than P_1. Formally this is implied by $1\delta_x \sqsubseteq \frac{1}{2}\delta_x + \frac{1}{2}\delta_x$.

4.2 A simple imperative language

We present a small imperative language **L**. It has the following (abstract) syntactic categories:

- integers **Num**, ranged over by n;
- locations **Loc**, ranged over by X;
- finite probability distributions over integers **Prob**, ranged over by χ;
- arithmetical expressions **Aexp**, ranged over by a;
- boolean expressions **Bexp**, ranged over by b;
- commands **Comm**, ranged over by c.

The (abstract) BNF for the last three syntactic categories are as follows:

$$a ::= n \mid X \mid a + a \mid a - a \mid a * a$$

$$b ::= \mathbf{true} \mid \mathbf{false} \mid a = a \mid a \leq a \mid \neg b \mid b \wedge b \mid b \vee b$$

$$\begin{array}{rl} c ::= & \mathbf{skip} \mid X := a \mid X := \chi \mid c; c \mid \\ & \mathbf{if}\ b\ \mathbf{then}\ c\ \mathbf{else}\ c \mid \mathbf{while}\ b\ \mathbf{do}\ c \mid c\ \mathbf{or}\ c. \end{array}$$

We also need the notion of *state*. A state is a function $\sigma : \mathbf{Loc} \to \mathbf{Num}$. We call Σ the set of states. We call any pair $\langle c, \sigma \rangle$ a *configuration*. We denote the set of all configurations by Γ. The set Γ is ranged over by γ. To make the notation more uniform we introduce (at the metalevel) the empty command ϵ. We use it with the following meaning:

$$\langle \epsilon, \sigma \rangle \equiv \sigma \ ,$$

$$\epsilon; c \equiv c; \epsilon \equiv c \ .$$

We extend consequently the notion of configuration so that a state σ is a configuration $\langle c, \sigma \rangle$ where $c = \epsilon$.

The operational semantics for arithmetic and boolean expressions is as usual. The operational semantics of configurations is given by a PST built using the rules in table 1. It is intuitive how, using these rules, one can build a PST on the set of (extended) configurations.

Now given a configuration $\gamma := \langle c, \sigma \rangle$ and a scheduler $\mathcal{S}$ for the PST $\mathcal{T}(\gamma)$ with initial state labeled by (γ), we consider the set of finite non-extensible labeled paths, $\mathcal{B}(\gamma, \mathcal{S})$ for the fully probabilistic tree obtained from $\mathcal{T}(\gamma)$ via the action of the scheduler $\mathcal{S}$. For every $s \in \mathcal{B}(\gamma, \mathcal{S})$ we define the probability of s, $\Pi(s)$, to be the product of the probability labels in s. We define $l(s)$ to be the label of the last node of s.

Definition 4.4. With $c, \sigma, \mathcal{S}$ as above, we define $Ev(\mathcal{S}, c, \sigma)$ to be the indexed valuation

$$Ev(\mathcal{S}, c, \sigma) := (l(s), \Pi(s))_{s \in \mathcal{B}(c, \sigma, \mathcal{S})} \ .$$

This definition is in accordance with the intuitive interpretation of indexed valuations: here the indexes are the paths, and the indexing function is assigning to every path its final state.

4.3 Adequate denotational semantics

We now give a denotational semantics in terms of indexed valuations. This semantics is adequate with respect to deterministic schedulers: an indexed valuation is in the denotation of a configuration if there is a deterministic scheduler which realizes it.

For the finite fragment of the language we can work in the category **SET**. The language $\mathbf{L}^-$ has the same syntax as $\mathbf{L}$ except that it does not include the constructor $\mathtt{while}\ b\ \mathbf{do}\ c$. The operational semantics of $\mathbf{L}^-$ is defined as for $\mathbf{L}$.

The denotational semantics

$$\mathcal{F}[\![c]\!] : \Sigma \to P(IV(\Sigma))$$

is defined in table 2. The IDV $(\sigma, p)_{* \in \{*\}}$ is denoted as (σ, p). The Kleisli extension for the monad $P \circ IV$ is denoted by $(-)^\dagger$. For $f : X \to P(IV(Y))$, we have $f^\dagger := \mu \circ P(IV(f)) : P(IV(X)) \to P(IV(Y))$.

Theorem 4.5 (Adequacy). *Let c be a command of* $\mathbf{L}^-$ *and* ν *be an indexed finite valuation in* $IV(\Sigma)$. *Then* $\nu \in \mathcal{F}[\![c]\!]\sigma$ *if and only if there exists a deterministic scheduler* $\mathcal{S}$ *for* $\mathcal{T}(\langle c, \sigma \rangle)$ *s.t.* $\nu \sim Ev(\mathcal{S}, c, \sigma)$

For the full language we work in **CONT**. We give a denotational semantics to **L** in terms of $\mathcal{P}(\mathcal{IV}(\Sigma))$. Given a command of **L**, we define its denotation

$$[\![c]\!] : \Sigma \to \mathcal{P}(\mathcal{IV}(\Sigma)) \ .$$

The definition is shown in table 3. Recall the behaviour of $\eta : \Sigma \to \mathcal{P}(\mathcal{IV}(\Sigma))$:

$$\eta(\sigma) = \{(\sigma, 1)\} \downarrow \ .$$

Again the symbol $(-)^\dagger$ denotes the Kleisli extension for the monad $\mathcal{P} \circ \mathcal{IV}$.

The adequacy theorem for the full language suffers the usual limitations due to the structure of the Hoare powerdomain, which involves only Scott closed sets.

$$
\begin{array}{cc}
\langle \mathbf{skip}, \sigma\rangle \xrightarrow{1} \sigma & \dfrac{\langle a, \sigma\rangle \to n}{\langle X := a, \sigma\rangle \xrightarrow{1} \sigma[n/X]} \\[2ex]
\langle X := \chi, \sigma\rangle (\xrightarrow{\chi(n)} \sigma[n/X]\ )_{n \in Supp(\chi)} & \dfrac{\langle c, \sigma\rangle (\xrightarrow{p_i} \langle c_i, \sigma_i\rangle)_{i \in I}}{\langle c; c', \sigma\rangle (\xrightarrow{p_i} \langle c_i; c', \sigma_i\rangle)_{i \in I}} \\[2ex]
\dfrac{\langle b, \sigma\rangle \to \mathbf{false}}{\langle \mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1, \sigma\rangle \xrightarrow{1} \langle c_1, \sigma\rangle} & \dfrac{\langle b, \sigma\rangle \to \mathbf{true}}{\langle \mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1, \sigma\rangle \xrightarrow{1} \langle c_0, \sigma\rangle} \\[2ex]
\dfrac{\langle c, \sigma\rangle (\xrightarrow{p_i} \gamma_i)_{i \in I}}{\langle c\ \mathbf{or}\ c', \sigma\rangle (\xrightarrow{p_i} \gamma_i)_{i \in I}} & \dfrac{\langle c', \sigma\rangle (\xrightarrow{p_j} \gamma_j)_{j \in J}}{\langle c\ \mathbf{or}\ c', \sigma\rangle (\xrightarrow{p_j} \gamma_j)_{j \in J}} \\[2ex]
\dfrac{\langle b, \sigma\rangle \to \mathbf{false}}{\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle \xrightarrow{1} \sigma} & \dfrac{\langle b, \sigma\rangle \to \mathbf{true}}{\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle \xrightarrow{1} \langle c; \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle}
\end{array}
$$

Table 1. Operational semantics

$$
\begin{array}{rcl rcl}
\mathcal{F}[\![\mathbf{skip}]\!]\sigma & = & \{(\sigma, 1)\} & \mathcal{F}[\![X := a]\!]\sigma & = & \{(\sigma[n/X], 1)\}\ \text{where } n = [\![a]\!]\sigma \\
\mathcal{F}[\![c_0\ \mathbf{or}\ c_1]\!]\sigma & = & \mathcal{F}[\![c_1]\!]\sigma \cup \mathcal{F}[\![c_0]\!]\sigma & \mathcal{F}[\![X := \chi]\!]\sigma & = & \{(\sigma[n/X], \chi(n))_{n \in Supp(\chi)}\} \\
\mathcal{F}[\![c_0; c_1]\!] & = & (\mathcal{F}[\![c_1]\!])^\dagger \circ \mathcal{F}[\![c_0]\!] & \mathcal{F}[\![\mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1]\!]\sigma & = & \begin{cases} \mathcal{F}[\![c_0]\!](\sigma) & \text{if } [\![b]\!]\sigma = \mathbf{true} \\ \mathcal{F}[\![c_1]\!](\sigma) & \text{if } [\![b]\!]\sigma = \mathbf{false} \end{cases}
\end{array}
$$

Table 2. Denotational semantics for the finite fragment

$$
\begin{array}{rcl rcl}
[\![\mathbf{skip}]\!] & = & \eta & [\![X := a]\!]\sigma & = & \eta(\sigma[n/X])\ \text{ where } n = [\![a]\!]\sigma \\
[\![c_0\ \mathbf{or}\ c_1]\!]\sigma & = & [\![c_1]\!]\sigma \cup [\![c_0]\!]\sigma & [\![X := \chi]\!]\sigma & = & \iota\big(\{(\sigma[n/X], \chi(n))_{n \in Supp(\chi)}\}\big) \\
[\![c_0; c_1]\!] & = & [\![c_1]\!]^\dagger \circ [\![c_0]\!] & [\![\mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1]\!]\sigma & = & \begin{cases} [\![c_0]\!](\sigma) & \text{if } [\![b]\!]\sigma = \mathbf{true} \\ [\![c_1]\!](\sigma) & \text{if } [\![b]\!]\sigma = \mathbf{false} \end{cases} \\
[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!] & = & \mathbf{Fix}(\Phi) \qquad \text{where} & \Phi(f)(\sigma) & = & \begin{cases} \eta(\sigma) & \text{if } [\![b]\!]\sigma = \mathbf{false}\ , \\ (f^\dagger \circ [\![c]\!])\sigma & \text{if } [\![b]\!]\sigma = \mathbf{true}\ . \end{cases}
\end{array}
$$

Table 3. Denotational semantics

Theorem 4.6. *Let c be a command of* $\mathbf{L}$ *and let* $\nu \in \mathcal{IV}(\Sigma)$. *Then* $\nu \in [\![c]\!]\sigma$ *iff for every* ξ *such that* $\xi \ll \nu$ *there exists a scheduler* $\mathcal{S}$ *for* (c,σ) *s.t.* $Ev(\mathcal{S},c,\sigma) \sqsupseteq \xi$.

There are examples showing that this formulation is in some sense optimal: even when an indexed valuation is in the denotation of a command there does not always exist a deterministic scheduler which attains it.

4.4 Comparison with Tix-Mislove

The main feature of our theorems is that they involve deterministic schedulers. We could have given a similar semantics using the Mislove-Tix functor.

Definition 4.7. [11] A *continuous d-cone* is a continuous quasi-cone satisfying $(p+q)A = pA \oplus qA$. A subset X of a continuous d-cone is *convex* if for every $x, y \in X$ and for every $p \in [0,1]$, we have that $px \oplus (1-p)y \in X$. If H is a continuous d-cone, we define

$$\mathcal{P}_{TM}(H) := \{X \subseteq H \mid X \neq \emptyset, \text{convex, Scott-closed}\}$$

With the sets ordered by inclusion, and the operations defined (essentially) pointwise, we have that $\mathcal{P}_{TM}(H)$ is a continuous d-cone.

Notice that when D is a continuous domain, the powerdomain of valuations $\mathcal{V}(D)$ is a continuous d-cone.

The functor $\mathcal{P}_{TM} \circ \mathcal{V}$ has a universal property, so that, when combined with the forgetful functor, it gives rise to a monad.

We can interpret the semantics in table 3 as being of the form

$$[\![c]\!] : \Sigma \to \mathcal{P}_{TM}(\mathcal{V}(\Sigma))\ .$$

The adequacy result is similar to the previous one, but it will speak of probabilistic schedulers. To state this theorem we make use of the "flattening" function $(-)^* : \mathcal{IV}(\Sigma) \to \mathcal{V}(\Sigma)$.

Theorem 4.8. *Let c be a command of* $\mathbf{L}$ *and let* $\zeta \in \mathcal{V}(\Sigma)$. *Then* $\zeta \in [\![c]\!]\sigma$ *iff for every* $\epsilon > 0$ *there exists a probabilistic scheduler* $\mathcal{S}$ *for* (c,σ) *s.t.* $\big(Ev(\mathcal{S},c,\sigma)\big)^* \sqsupseteq (1-\epsilon)\zeta$.

5 Future Work

We have shown that a distributive law exists between the Hoare powerdomain and the Indexed Valuations. The question arises whether similar results hold for the Plotkin and the Smyth powerdomain. We believe that each nondeterministic powerdomain has its corresponding Indexed Valuation functor. The key is the axiom (HV). If we omit it we get a theory whose free domain-algebra distributes over the Plotkin powerdomain. If we replace it with

$$pA \oplus qA \sqsubseteq (p+q)A \quad (SV)$$

we get a theory whose free domain-algebras distribute over the Smyth powerdomain. These ideas have to be studied and a computational meaning has to be found for them.

It would be also interesting to find a concrete characterization of Indexed Valuations. In the category of sets they are but finite random variables. It seems promising to generalize this setting to random variables on suitable measure spaces.

Acknowledgments

Glynn Winskel contributed with essential suggestions to this work. In particular he had the idea of using indexes, while I was still using the more awkward multisets. Special thanks to Achim Jung for the very useful conversations during my visit to Birmingham. In particular he convinced me that the distributive law in the category of domains could be defined via the bases, without the need for a concrete representation. Thanks to Andrzej Filinski who showed me how the nondeterministic and probabilistic monads are combined in ML. Thanks to Alex Berg, Michael Mislove, Gordon Plotkin, Luigi Santocanale and Zhe Yang for useful discussions. Thanks to Marcelo Fiore and Zhe Yang for useful comments on a draft version of the paper.

References

[1] S. Abramsky and A. Jung. Domain theory. In *Handbook of Logic in Computer Science*, volume 3. Oxford, 1994.

[2] J. Beck. Distributive laws. In *Seminar on Triples and Categorical Homology Theory*, pages 119–140, 1969.

[3] H. Hansson. *Time and Probability in Formal Design of Distributed systems*. PhD thesis, Uppsala University, 1991.

[4] C. Jones. *Probabilistic Non-determinism*. PhD thesis, University of Edinburgh, 1990.

[5] B. Jonsson, K. Larsen, and W. Yi. Probabilistic extensions of process algebras. In *Handbook of Process Algebras*. Elsevier, 2001.

[6] O. Kirch. Bereiche und Bewertungen. Master's thesis, Technische Hochschule Darmstadt, 1993.

[7] M. Mislove. Nondeterminism and probabilistic choice: Obeying the law. In *Proc. 11th CONCUR*, volume 1877 of *LNCS*, pages 350–364. Springer, 2000.

[8] E. Moggi. Notions of computation and monads. *Information and Computation*, 93(1):55–92, July 1991.

[9] G. Plotkin. Domains. University of Edinburgh, 1983.

[10] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. *Nordic Journal of Computing*, 2(2):250–273, 1995. An extended abstract appears in *Proc. 5th CONCUR*, Uppsala, Sweden, LNCS 836, pages 481–496, August 1994.

[11] R. Tix. *Continuous D-Cones: Convexity and Powerdomain Constructions*. PhD thesis, Technische Universität Darmstadt, 1999.

[12] D. Varacca. Issues in probabilistic semantics. Progress report, BRICS - Aarhus University, 2001.

[13] M. Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *Proc. 26th FOCS*, pages 327–338, 1985.

A Sketch of the proof of theorem 3.9

Take a domain D with basis B. Consider the function $a_B : IV(P(B)) \to P(IV(B))$:

$$a_B\big((S_i, p_i)_{i\in I}\big) = \big\{(h(i), p_i)_{i\in I} \,|\, h : I \to B,\ h(i) \in S_i\big\}$$

Lemma A.1. *The function a_B preserves the abstract basis relation.*

Define α_D to be the extension of a_B.

Theorem A.2. *The family $\alpha_D : \mathcal{IV}(\mathcal{P}(D)) \to \mathcal{P}(\mathcal{IV}(D))$ defined above is a distributive law.*

We use the following fact:

Theorem A.3. *Consider domains D, D', D'' with bases B, B', B''. Take functions $f : B \to B'$, $g : B' \to B''$ preserving the AB-relation. Then*

$$\overline{g} \circ \overline{f} = \overline{g \circ f}.$$

Thus if we prove the commutativity of a diagram in the category of abstract basis and monotonic functions, this carries over to the diagram constituted by their extensions.

Now we have to prove the commutativity of four diagrams. We start with

Proposition A.4.

$$\begin{tikzcd}
\mathcal{IV} \circ \mathcal{IV} \circ \mathcal{P} \arrow[r, "\mathcal{IV}\alpha"] \arrow[d, "\mu^{\mathcal{IV}}\mathcal{P}"'] & \mathcal{IV} \circ \mathcal{P} \circ \mathcal{IV} \arrow[r, "\alpha\mathcal{IV}"] & \mathcal{P} \circ \mathcal{IV} \circ \mathcal{IV} \arrow[d, "\mathcal{P}\mu^{\mathcal{IV}}"] \\
\mathcal{IV} \circ \mathcal{P} \arrow[rr, "\alpha"'] & & \mathcal{P} \circ \mathcal{IV}
\end{tikzcd}$$

We know that for every (abstract) basis B the following diagram commutes (because it is essentially the diagram for the distributive law in the category **SET**):

$$\begin{tikzcd}
IV(IV(P(B))) \arrow[r, "\mu^{IV}_{P(B)}"] \arrow[d, "IV(a_B)"'] & IV(P(B)) \arrow[dd, "a_B"] \\
IV(P(IV(B))) \arrow[d, "a_{IV(B)}"'] & \\
P(IV(IV(B))) \arrow[r, "P(\mu^{IV}_B)"'] & P(IV(B))
\end{tikzcd}$$

Lemma A.5. *Using the above notation we have*

- $\alpha_D = \overline{a_B}$
- $\alpha_{\mathcal{IV}(D)} = \overline{a_{IV(B)}}$
- $\mu^{\mathcal{IV}}_{\mathcal{P}(D)} = \overline{\mu^{IV}_{P(B)}}$
- $\mathcal{P}(\mu^{\mathcal{IV}}_D) = \overline{P(\mu^{IV}_B)}$
- $\mathcal{IV}(\alpha_D) = \overline{IV(a_B)}$

The first three equalities are true by definition, while some work is needed to show the other two. It can be helpful to state a more general result

Proposition A.6. *Let D, D' be continuous domains with basis B, B'. Let $f : B \to B'$ be a function preserving $\ll$. Then, using the above notation*

$$\overline{P(f)} = \mathcal{P}(\overline{f}) \quad \overline{IV(f)} = \mathcal{IV}(\overline{f})$$

Lemma A.5 together with lemma A.1 and theorem A.3 give us the proof of proposition A.4.

The proof of two of the other diagrams is similar. A bit different is the following.

Proposition A.7.

$$\begin{tikzcd}
& \mathcal{P} \arrow[dl, "\eta^{\mathcal{IV}}\mathcal{P}"'] \arrow[dr, "\mathcal{P}(\eta^{\mathcal{IV}})"] & \\
\mathcal{IV} \circ \mathcal{P} \arrow[rr, "\alpha"'] & & \mathcal{P} \circ \mathcal{IV}
\end{tikzcd}$$

The difference is that η^{IV} does not preserve the AB-relation. But it is not difficult to check directly the commutativity of the diagram, using the definition of the domains as ideal completions.

Naturality of α_D is shown directly in a similar way, relying on the naturality of a_B in the category of sets.