RAIRO-Theor. Inf. Appl. 46 (2012) 413–450
DOI: 10.1051/ita/2012012

Available online at:
www.rairo-ita.org

PROBABILISTIC OPERATIONAL SEMANTICS FOR THE LAMBDA CALCULUS

Ugo Dal Lago[1] and Margherita Zorzi[2]

Abstract. Probabilistic operational semantics for a nondeterministic extension of pure λ-calculus is studied. In this semantics, a term evaluates to a (finite or infinite) distribution of values. Small-step and big-step semantics, inductively and coinductively defined, are given. Moreover, small-step and big-step semantics are shown to produce identical outcomes, both in call-by-value and in call-by-name. Plotkin's CPS translation is extended to accommodate the choice operator and shown correct with respect to the operational semantics. Finally, the expressive power of the obtained system is studied: the calculus is shown to be sound and complete with respect to computable probability distributions.

Mathematics Subject Classification. 68Q55, 03B70.

1. Introduction

Randomized computation is central to several areas of theoretical computer science, including computational complexity, cryptography, analysis of computation dealing with uncertainty, incomplete knowledge agent systems. Some works have been devoted also to the design and analysis of programming languages with stochastic aspects. For various reasons, the functional programming paradigm seems appropriate in this context, because of the very thin gap between the realm of programs and the underlying probability world.

Keywords and phrases. Lambda calculus, probabilistic computaion, operational semantics.

[1] Università di Bologna & EPI FOCUS, Dipartimento di Scienze dell'Informazione Mura Anteo Zamboni, 7, 40127 Bologna, Italy. dallago@cs.unibo.it

[2] Laboratoire d'Informatique L.I.P.N., Université Paris-Nord, supported by ANR Complice Project, France. margherita.zorzi@univr.it

Article published by EDP Sciences

The large majority of the literature on probabilistic functional programming view probability as a *monad*, in the sense of Moggi [17, 18]. This is the case for the works by Jones and Plotkin about denotational semantics of probabilistic functional programs [15], or for many of the recently proposed probabilistic functional programming languages: the stochastic λ-calculus [24] and the λ-calculi by Park *et al.* [20, 21]. The monadic structure of probability distributions provides a good denotational model for the calculi and it makes evident how the mathematical foundations of probability can be applied directly in a natural way to the semantics of probabilistic programs. This allows, for example, to apply this approach to the formalization of properties of randomized algorithms in interactive theorem proving [1]. The monadic approach seems particularly appropriate in applications, since some programming languages, like Haskell, directly implement monads.

But there is another, more direct, way to endow the λ-calculus with probabilistic choice, namely by enriching it with a binary choice operator $\oplus$. This way, we can form terms whose behavior is probabilistically chosen as the one of the first or of the second argument. It is not clear, however, whether the operational theory underlying ordinary, deterministic λ-calculus, extends smoothly to this new, probabilistic setting. The aim of this paper is precisely to start an investigation in this direction. The *object* of our study will be the nondeterministic λ-calculus, being it a minimal extension of the ordinary λ-calculus with a choice operator. The *subject* of our study, on the other hand, will be the properties of two notions of probabilistic semantics for it, namely call-by-value and call-by-name evaluation. Big-step and small-step semantics will be defined and proved equivalent, both when defined inductively and when defined coinductively. CPS translations extending the ones in the literature are presented and proved to have the usual properties enjoyed in the deterministic case. Finally, some results about the expressive power of the obtained calculus are proved.

1.1. Related works

A pioneering investigation in the field of stochastic functional languages is Probabilistic LCF (PLCF in the following), by Saheb-Djahromi [26]. PLCF is a typed, higher-order calculus based on Milner's LCF, Plotkin's PCF [23] and Plotkin's probabilistic domains (further developed by Jones and Plotkin in [15]). The syntax of PLCF includes two kinds of abstractions which deal separately with call-by-value and call-by-name evaluation. The author declares the explicit intent of providing "a foundation for the probabilistic study of the computation" and even if a number of important aspects are unexplored, the approach is interesting and related to the present investigation. Saheb-Djahromi provides both denotational and operational semantics for PLCF. Denotational semantics, defined in terms of probabilistic domains, is an extension of Milner's and Plotkin's one. Operational semantics is given as a Markov chain, and an equivalence result between the latter and a denotational model is stated and proved as an extension of Plotkin's results.

In recent years, some λ-calculi with probabilistic features have been introduced, strongly oriented to applications (*e.g.* robotics). The most developed approach is definitely the monadic one [17, 18], based on the idea that probability distributions form a monad [12, 15]. In [24], Ramsey and Pfeffer introduce the *stochastic λ-calculus*, in which the denotation of expressions are distributions and in which the probability monad is shown to be useful in the evaluation of queries about the defined probabilistic model. A measure theory and a simple measure language are introduced and the stochastic λ-calculus is compiled into the measure language. Moreover, a denotational semantics based upon the monad of probability measures is defined. In [19], Park defines the typed calculus λ_γ, an extension of the typed λ-calculus with an *integral abstraction* and a *sampling construct*, which bind probability and sampling variables respectively. A system of simple types for λ_γ is also introduced. The author briefly discusses about the expressive power of λ_γ: the calculus has been shown to be able to express the most relevant probability distributions. λ_γ does not make use of monads, which are however present in [20], in which the idea of a calculus based on the mathematical notion of a sampling function is further developed through the introduction of $\lambda_\circ$. $\lambda_\circ$ is based on the monad of sampling functions and is able to specify probability distributions over infinite discrete domains and continuous domains. The authors also develop a new operational semantics, called *horizontal operational semantics*. The calculus $\lambda_\circ$ is further studied and developed in [21].

Nondeterminism and probability, however, can find their place in the λ-calculus by completely different means. In [7], de'Liguoro and Piperno propose a non deterministic λ-calculus, called $\Lambda_\oplus$. $\Lambda_\oplus$ is nothing more than the usual, untyped, λ-calculus with an additional binary operator $\oplus$ which represents binary, nondeterministic choice: $M \oplus N$ rewrites to either M or N. The authors give a standardization theorem and an algebraic semantics for $\Lambda_\oplus$. The classical definition of a Böhm tree is extended to the non-deterministic case by means of inductively defined "approximating operators". Several relevant properties such as discriminability are studied and, moreover, some suitable models for non-deterministic λ-calculus are proposed and discussed. In [8], Di Pierro *et al.* propose an untyped λ-calculus with probabilistic choice. Its syntax is itself an extension of pure λ-calculus with n-ary probabilistic choice in the form $\bigoplus_{i=1}^{n} p_i : M_i$. The main objective, however, is showing how probabilistic abstract interpretation can be exploited in the context of static analysis of probabilistic programs even in presence of higher-order functions. This is reflected by its operational semantics, which is more directed to program analysis (when two terms can be considered equivalent) than to computation (what is the value obtained by evaluating a program? how can we compute it?). Our choice here consists in endowing $\Lambda_\oplus$ with a probabilistic operational semantics. This has the advantage of simplicity, since the language is a minimal extension of the well-known pure λ-calculus. This does not mean, however, that the expressive power of $\Lambda_\oplus$ is low. As showed in Section 8, all computable probabilistic distributions can be represented in our language, which is expressive at least as systems designed for more practical issue (such as [19]).

1.2. Outline

After some motivating observations about the interplay between rewriting and nondeterministic choice (Sect. 2), the calculus $\Lambda_\oplus$ and its call-by-value probabilistic operational semantics are introduced (Sects. 4 and 5, respectively). Both small-step and big-step semantics are defined and their equivalence is proved in detail. Remarkably, the result holds even when operational semantics is formulated coinductively. The same results hold for call-by-name evaluation and are briefly described in Section 6. Call-by-value and call-by-name can be shown to be able to simulate each other by slight modifications of the well-known CPS translations [22], described in Section 7. The paper ends with a result about the expressive power of $\Lambda_\oplus$, namely the equivalence between representable and computable distributions.

2. Some motivating observations

λ-calculus can be seen both as an equational theory on λ-terms and as an abstract model of computation. In the first case, it is natural (and customary) to allow to apply equations (*e.g.* β or η equivalences) at any position in the term. The obtained calculus enjoys confluence, in the form of the so-called Church-Rosser theorem: equality of terms remains transitive even if equations becomes rewriting rules, *i.e.* if they are given with an orientation. More computationally, on the other hand, the meaning of any λ-term is the *value* it evaluates to in some strategy or machine. In this setting, abstractions are often considered as values, meaning that reduction cannot take place in the scope of a λ-abstraction. What's obtained this way is a calculus with *weak* reduction which is not confluent in the Church-Rosser sense. As an example, take the term $(\lambda x.\lambda y.x)((\lambda z.z)(\lambda z.z))$. In call-by-value, it reduces to $\lambda y.(\lambda z.z)$. In call-by-name, it reduces to $\lambda y.(\lambda z.z)(\lambda z.z)$. A beautiful operational theory has developed since Plotkin's pioneering work [22]. Call-by-value and call-by-name have been shown to be dual to each other [2], and continuation-based translations allowing to simulate one style with the other have been designed and analyzed very carefully [4, 5].

Now, suppose to endow the λ-calculus with nondeterministic sums. Suppose, in other words, to introduce a binary infix term operator $\oplus$ such that $M \oplus N$ can act either as M or N in a nondeterministic flavor. What we obtain is the so-called nondeterministic lambda-calculus, which has been introduced and studied in [7]. We cannot hope to get any confluence results in this setting (at least if we stick to reduction as a binary relation on *terms*): a term like $M = M_\top \oplus M_\bot$. (where $M_\top = \lambda x.\lambda y.x$ and $M_\bot = \lambda x.\lambda y.y$ are the usual representation of truth values in the λ-calculus) reduces to two distinct values (which are different in a very strong sense) in any strategy. The meaning of any λ-term, here, is a *set of values* accounting for all the possible outcomes. The meaning of M, as an example, is the set $\{M_\top, M_\bot\}$. Nontermination, mixed with nondeterministic choice, implies that the meaning of terms can even be an *infinite* set of values. As an example, take the λ-term

$$(Y(\lambda x.\lambda y.(y \oplus x(M_{\mathrm{succ}}y))))V_0$$

where Y is a fixed-point combinator, V_n represents the natural number $n \in \mathbb{N}$ and M_{succ} computes the successor. Evaluating it (*e.g.* in call-by-value) produces the infinite set

$$\{V_0, V_1, V_2, \ldots\}.$$

It's clear that ordinary ways to give an operational semantics to the λ-calculus (*e.g.* a finitary, inductively defined, formal system) do not suffice here, since they intrinsically attribute a finitary meaning to terms. So, how can we define small-step and big-step semantics in a nondeterministic setting?

Another problematic point is confluence. The situation is even worse than in ordinary, deterministic, λ-calculus. Take the following term [26,28]:

$$M = (\lambda x.M_{xor}xx)(M_\top \oplus M_\bot)$$

where $M_{xor} = \lambda x.\lambda y.(x(\lambda z.zM_\bot M_\top)(\lambda z.zM_\top M_\bot))y$ is a term computing a parity function of the two bits in input. When reducing it call-by-value, we obtain the outcome $\{M_\bot\}$, while reducing it call-by-name, we obtain $\{M_\top, M_\bot\}$. This phenomenon is due to the interaction between nondeterministic choice and copying: in call-by-value we choose before copying, and the final result can only be of a certain form. In call-by-name, we copy before choosing, getting distinct outcomes. What happens to CPS translations in this setting? Is it still possible to define them?

The aim of this paper is precisely to give answers to the questions above. Or, better, to give answers to their natural, quantitative generalizations obtained by considering $\oplus$ as an operator producing any of two possible outcomes with identical probability.

3. A brief introduction to coinduction

The *coinduction principle* is a definitional principle dual to induction, which supports a proof principle. It is a very useful instrument for reasoning about diverging computations and unbounded structures. Coinductive techniques are not yet as popular as inductive techniques; for this reason, in this section we give a short introduction to coinduction, following [13, 16]. A nice introduction to coinduction can be found, *e.g.*, in [27].

Inductive and coinductive definitions are given here by interpretating sets of rules. Let $\mathcal{U}$ be a set of judgements.

Definition 3.1 (inference system)**.** An *inference rule* is a pair (A, c), where $c \in \mathcal{U}$ is the conclusion of the rule and $A \subseteq \mathcal{U}$ is the set of its premises or antecedents. An *inference system* Φ over $\mathcal{U}$ is a set of inference rules over $\mathcal{U}$.

The usual way to give meaning to an inference system Φ is to consider fixed points of the associated inference operator:

Definition 3.2 (operator induced by an inference system Φ)**.** If Φ is an inference system over $\mathcal{U}$, we define the operator $F_\Phi : \wp(\mathcal{U}) \to \wp(\mathcal{U})$ as

$$F_\Phi(A) = \{c \in \mathcal{U} \mid \exists B \subseteq A \text{ such that } (B, c) \in \Phi\}.$$

In other words, $F_\Phi(A)$ is the set of judgments that can be inferred in *one step* from the judgments in A by using the inference rules. The following definitions are the main ingredients for the inductive and coinductive proof principles:

Definition 3.3 (closed and consistent sets)**.** If Φ is a set of inference rules over $\mathcal{U}$, then:

- A set $A \subseteq \mathcal{U}$ is said to be (F_Φ-)*closed* if $F_\Phi(A) \subseteq A$.
- A set $A \subseteq \mathcal{U}$ is said to be (F_Φ-)*consistent* if $A \subseteq F_\Phi(A)$.

Intuitively, a closed set A is a set such that no new judgments can be inferred from A and a consistent set A is a set such that all judgments that cannot be inferred from A are not in A.

It is quite easy to prove that the inference operator is monotone, *i.e.* if $A \subseteq B$, then $F_\Phi(A) \subseteq F_\Phi(B)$. By Tarski's fixed point theorem for complete lattices (taking here the complete lattice $(\wp(\mathcal{U}), \subseteq, \bigcup, \bigcap)$), it follows that the inference operator possesses both a least fixed point and a greatest fixed point, which are the smallest closed set and the largest consistent set, respectively:

$$\begin{aligned} \mathit{lfp}(F_\Phi) &= \bigcap \{A \mid F_\Phi(A) \subseteq A\}; \\ \mathit{gfp}(F_\Phi) &= \bigcup \{A \mid A \subseteq F_\Phi(A)\}. \end{aligned}$$

The least fixed point $\mathit{lfp}(F_\Phi)$ is the *inductive interpretation* of the inference system Φ, and the greatest fixed point $\mathit{gfp}(F_\Phi)$ is its *coinductive interpretation*. These interpretations lead to the following two proof principles:

- **Induction principle**: to prove that all judgments in the inductive interpretation belong to a set A, show that A is F_Φ-closed.
- **Coinduction principle**: to prove that all judgments in a set A belong to the coinductive interpretation, show that A is F_Φ-consistent. Indeed, if A is F_Φ-consistent, then

$$\begin{aligned} A \subseteq F_\Phi(A) &\Longrightarrow A \in \{B \mid B \subseteq F_\Phi(B)\} \\ &\Longrightarrow A \subseteq \bigcup \{B \mid B \subseteq F_\Phi(B)\} = \mathit{gfp}(F_\Phi). \end{aligned}$$

We conclude this section with two examples:

Example 3.4 (finite and infinite trees)**.** Let us consider all those (finite and infinite) trees whose nodes are labelled with a symbol from the alphabet $\{\circ, \bullet\}$.

Examples of finite trees are

$$\circ\langle\bullet,\circ\rangle$$
$$\bullet\langle\circ\langle\circ,\bullet\rangle,\bullet\langle\circ,\bullet\rangle,\circ\rangle$$
$$\bullet$$

while an example of an infinite tree is the (unique) tree t such that $t = \circ\langle t, t, t\rangle$. Let $\mathcal{U}$ the set of judgments in the form $t \Downarrow$, where t is a possibly infinite tree as above. Moreover, let $\varPhi$ be the inference system composed by all the instances of the four rules below:

$$\frac{}{\bullet \Downarrow} \qquad \frac{}{\circ \Downarrow} \qquad \frac{t_1 \Downarrow, \ldots, t_n \Downarrow}{\bullet\langle t_1, \ldots, t_n\rangle \Downarrow} \qquad \frac{t_1 \Downarrow, \ldots, t_n \Downarrow}{\circ\langle t_1, \ldots, t_n\rangle \Downarrow}\,.$$

Let us define $F_\varPhi : \wp(\mathcal{U}) \to (\mathcal{U})$ as

$$\begin{aligned} F_\varPhi(A) = \{\bullet \Downarrow\} \cup \{\circ \Downarrow\} \cup \{\bullet\langle t_1, \ldots, t_n\rangle \Downarrow \ \mid t_1 \Downarrow \in A, \ldots, t_n \Downarrow \in A\} \cup \\ \{\circ\langle t_1, \ldots, t_n\rangle \Downarrow \ \mid t_1 \Downarrow \in A, \ldots, t_n \Downarrow \in A\}. \end{aligned}$$

for each $A \in \mathcal{U}$. The inductive interpretation of $\varPhi$, $\mathit{lfp}(F_\varPhi)$, contains all judgements $t \Downarrow$ where t is *finite* tree, while its coinductive interpretation, $\mathit{gfp}(F_\varPhi)$, contains also all judgments $t \Downarrow$ where t is *infinite*.

Example 3.5 (binary streams)**.** Another example of a coinductively defined set is the set of all binary streams $\{0,1\}^\omega$ [11, 25]. Let $\mathbb{R}^\omega = \{\sigma \mid \sigma : \mathbb{N} \to \mathbb{R}\}$ be the set of all real streams. In this example, judgements are in the form $\sigma \Downarrow$, where σ is a real stream. We give now an inference system whose coinductive interpretation by means of the associated operator yields the set $\{0,1\}^\omega$. First, let us define two functions on $\mathbb{R}^\omega$: $\mathsf{hd} : \mathbb{R}^\omega \to \mathbb{R}$ defined as $\mathsf{hd}(\sigma) = \sigma(0)$, and $\mathsf{tl} : \mathbb{R}^\omega \to \mathbb{R}^\omega$ defined as $\mathsf{tl}(\sigma) = \rho$ where $\rho(n) = \sigma(n+1)$ for every n. Let $\varPhi$ be the inference system consisting of (all the instances of) the two rules below:

$$\frac{\mathsf{hd}(\sigma) = 0 \quad \mathsf{tl}(\sigma) \Downarrow}{\sigma \Downarrow} \qquad\qquad \frac{\mathsf{hd}(\sigma) = 1 \quad \mathsf{tl}(\sigma) \Downarrow}{\sigma \Downarrow}$$

$\varPhi$ induces the monotone operator $F_\varPhi : \wp(\mathbb{R}^\omega) \to \wp(\mathbb{R}^\omega)$ defined as

$$\begin{aligned} F_\varPhi(A) = \{\sigma \Downarrow \ \mid \mathsf{hd}(\sigma) = 0 \text{ and } \mathsf{tl}(\sigma) \Downarrow \in A\} \cup \\ \{\sigma \Downarrow \ \mid \mathsf{hd}(\sigma) = 1 \text{ and } \mathsf{tl}(\sigma) \Downarrow \in A\} \end{aligned}$$

for each set $A \in \wp(\mathbb{R}^\omega)$. $F_\varPhi(A)$ is the set of judgments that can be inferred in one step from the judgments in A by using the inference rules. The coinductive interpretation of $F_\varPhi$ gives the set we are looking for, namely the set of all judgements $\sigma \Downarrow$, where σ is an infinite sequence of 0 and 1.

4. Syntax and preliminary definitions

In this section we introduce the syntax of $\Lambda_\oplus$, a language of λ-terms with binary choice introduced by de'Liguoro and Piperno [7]. This is the language whose probabilistic semantics is the main topic of this paper.

The most important syntactic category is certainly those of terms. Actually, $\Lambda_\oplus$ is nothing more than the usual untyped and pure λ-calculus, endowed with a binary choice operator $\oplus$ which is meant to represent nondeterministic choice.

Definition 4.1 (terms)**.** Let $\mathsf{X} = \{x, y, \ldots\}$ be a denumerable set of variables. The set $\Lambda_\oplus$ of *term expressions*, or *terms* is the smallest set such that:

1. if $x \in \mathsf{X}$ then $x \in \Lambda_\oplus$;
2. if $x \in \mathsf{X}$ and $M \in \Lambda_\oplus$, then $\lambda x.M \in \Lambda_\oplus$;
3. if $M, N \in \Lambda_\oplus$ then $MN \in \Lambda_\oplus$;
4. if $M, N \in \Lambda_\oplus$ then $M \oplus N \in \Lambda_\oplus$.

Terms are ranged over by metavariables like M, N, L.

Terms, as usual, are considered modulo renaming of bound variables. The set of free variables of a term M is indicated as $\mathsf{FV}(M)$ and is defined as usual. A term M is *closed* if $\mathsf{FV}(M) = \emptyset$. The (capture-avoiding) substitution of N for the free occurrences of x in M is denoted $M\{x/N\}$. Unless otherwise stated, all results in this paper hold only for *programs*, that is to say for closed terms. Values are defined in a standard way:

Definition 4.2 (values)**.** A term is a *value* if it is a variable or a λ-abstraction. We will call Val the set of all values. Values are ranged over by metavariables like V, W, X.

The reduction relation $\to$ considered in [7] is obtained by extending usual β-reduction with two new reduction rules, namely $M \oplus N \to M$ and $M \oplus N \to N$, which can be applied in every possible context. In this paper, following Plotkin [22], we concentrate on weak reduction: computation can only take place in applicative (or choice) contexts.

Notation 4.3. In the following, we sometimes need to work with finite sequences of terms. A sequence $M_1, \ldots, M_n$ is denoted as $\overline{M}$. This notation can be used to denote sequences obtained from other sequences and terms, *e.g.*, $M \oplus \overline{N}$ is $M \oplus N_1, \ldots, M \oplus N_n$ whenever $\overline{N}$ is $N_1, \ldots, N_n$.

4.1. Distributions

In the probabilistic semantics we will endow $\Lambda_\oplus$ with, a program reduces not to a single value but rather to a distribution of possible observables, *i.e.* to a function assigning a probability to any value. This way, all possible outputs of all binary choices are taken into account, each with its own probability. Divergence is indeed a possibility in the untyped setting and our definition reflects it.

Definition 4.4 (distributions)**.**

1. A *probability distribution* is a function $\mathscr{D}:\mathsf{Val} \to \mathbb{R}_{[0,1]}$ such that $\sum_{V\in\mathsf{Val}} \mathscr{D}(V) \leq 1$. $\mathcal{P}$ denotes the set of all probability distributions.
2. A *proper probability distribution* is a probability distribution such that
$$\sum_{V\in\mathsf{Val}} \mathscr{D}(V) = 1.$$
3. Given a probability distribution $\mathscr{D}$, its *support* $\mathsf{S}(\mathscr{D})$ is the subset of Val whose elements are values to which $\mathscr{D}$ attributes positive probability.
4. Given a probability distribution $\mathscr{D}$, its *sum* $\sum \mathscr{D}$ is simply $\sum_{V\in\mathsf{Val}} \mathscr{D}(V)$.

The notion of a probability distribution as we gave it is general enough to capture the semantics of those terms which have some nonnull probability of divergence. We use the expression $\{V_1^{p_1}, \ldots, V_k^{p_k}\}$ to denote the probability distribution $\mathscr{D}$ with finite support defined as $\mathscr{D}(V) = \sum_{V_i = V} p_i$. Please observe that $\sum \mathscr{D} = \sum_{i=1}^{k} p_i$. Sometimes we need to compare distinct distributions. The natural way to do that is just by lifting the canonical order on $\mathbb{R}$ up to distributions, pointwise:

Definition 4.5. $\mathscr{D} \leq \mathscr{E}$ iff $\mathscr{D}(V) \leq \mathscr{E}(V)$ for every value V.

The structure $(\mathcal{P}, \leq)$ is a partial order, but not a lattice [6]: the join of two distributions $\mathscr{D}$ and $\mathscr{E}$ does not necessarily exist in $\mathcal{P}$. However, $(\mathcal{P}, \leq)$ is a complete meet-semilattice, since meets are always guaranteed to exist: for every $A \subseteq \mathcal{P}$, the greatest lower bound of distributions in A is itself a distribution. The least upper bound of $A \subseteq \mathcal{P}$, on the other hand, is indeed a distribution when A is a ω-chain. This turns $(\mathcal{P}, \leq)$ into a ω-complete partial order, for which $\sum(\cdot)$ (as a function from distributions to $[0,1]$) is continuous. Finally, please observe that all functions from Val to $\mathbb{R}_\infty$ actually form a complete lattice. And that $\sum(\cdot)$ as a function from those functions to $\mathbb{R}_\infty$ is a complete lattice homomorphism.

5. Call-by-value

In this section, four ways to give a probabilistic semantics to $\Lambda_\oplus$ are introduced, all of them following the so-called call-by-value discipline. All the relational symbols introduced in this section are summarized for the reader's convenience in Table 1.

A (weak) call-by-value notion of reduction can be obtained from ordinary reduction by restricting it in such a way that only values are passed to functions. Accordingly, choices are only made when both alternatives are themselves values:

Definition 5.1 (call-by-value reduction)**.** Leftmost reduction $\mapsto_{\mathsf{v}}$ is the least binary relation between $\Lambda_\oplus$ and $\Lambda_\oplus^*$ such that:

$$\begin{array}{rlrl}
(\lambda x.M)V \mapsto_{\mathsf{v}} M\{x/V\}; & & V \oplus W \mapsto_{\mathsf{v}} V, W; & \\
MN \mapsto_{\mathsf{v}} \overline{L}N & \text{if } M \mapsto_{\mathsf{v}} \overline{L}; & M \oplus N \mapsto_{\mathsf{v}} \overline{L} \oplus N & \text{if } M \mapsto_{\mathsf{v}} \overline{L}; \\
VM \mapsto_{\mathsf{v}} V\overline{N} & \text{if } M \mapsto_{\mathsf{v}} \overline{N}; & V \oplus M \mapsto_{\mathsf{v}} V \oplus \overline{N} & \text{if } M \mapsto_{\mathsf{v}} \overline{N};
\end{array}$$

where $V, W \in \mathsf{Val}$.

Table 1. Call-by-value probabilistic semantics: symbols.

Symbol	Description
$\mapsto_{\mathsf{v}}$	Leftmost reduction (Sect. 5, Def. 5.1)
$\Rightarrow_{\mathsf{IV}}$	Inductive small-step semantics relation (Sect. 5.1.1)
$\Rightarrow_{\mathsf{CV}}$	Coinductive small-step semantics relation (Sect. 5.1.3)
$\Rightarrow^{\infty}$	Coinductive small-step semantics relation for divergence (Sect. 5.1.2)
$\Downarrow_{\mathsf{IV}}$	Inductive big-step semantics relation (Sect. 5.2)
$\Downarrow_{\mathsf{CV}}$	Coinductive big-step semantics relation (Sect. 5.2)
$\mathcal{S}_{\mathsf{IV}}(M)$	Inductive small-step semantics of M (Sect. 5.1.1, Def. 5.3)
$\mathcal{S}_{\mathsf{CV}}(M)$	Coinductive small-step semantics of M (Sect. 5.1.3)
$\mathcal{D}_{\mathsf{CV}}(M)$	Coinductive small-step semantics of divergence for M (Sect. 5.1.2)
$\mathcal{B}_{\mathsf{IV}}(M)$	Inductive big-step semantics of M (Sect. 5.2, Def. 5.7)
$\mathcal{B}_{\mathsf{CV}}(M)$	Coinductive big-step semantics of M (Sect. 5.2)

Please, notice that reduction *is not probabilistic*. In fact, reduction is a relation between terms and unlabeled sequences of terms without any reference to probability. Informally, if $M \mapsto_{\mathsf{v}} N_1 \dots N_n$ means that M rewrites in one step to every N_i with the *same* probability $1/n$. Clearly, $n \in \{1,2\}$ whenever $M \mapsto_{\mathsf{v}} N_1 \dots N_n$. Notice again how the evaluation of *both* branches of a binary choice is done *before* performing the choice itself. One the one hand, this is very much in the style of call-by-value evaluation. On the other, a more standard notion of choice, which is performed before evaluating the branches can be easily encoded as follows:

$$M + N = ((\lambda x.\lambda y.x) \oplus (\lambda x.\lambda y.y))(\lambda z.M)(\lambda z.N)(\lambda w.w),$$

where z does not appear free in M nor in N.

5.1. CbV small-step semantics

Following the general methodology described in [16], we model separately convergence and divergence. Finite computations are, as usual, inductively defined, while divergence can be captured by interpreting a different set of rules coinductively. Both definitions give some *quantitative* information about the dynamics of any term M: either a distribution of possible outcomes is associated to M, or the probability of divergence of M is derived. Both induction and coinduction can be used to characterize the distribution of values a term evaluates to. In an inductive characterization, one is allowed to underapproximate the target distribution, but then takes an upper bound of all the approximations. In a coinductive characterization, on the other hand, one can naturally overapproximate the distribution, and take a lower bound.

5.1.1. Inductive CbV small-step semantics for convergence

A small-step semantics for convergence is captured here by way of a binary relation $\Rightarrow_{\mathsf{IV}}$ between terms in $\Lambda_\oplus$ and distributions. The relation $\Rightarrow_{\mathsf{IV}}$ is defined as the inductive interpretation of the inference system whose rules follow:

$$\frac{}{M \Rightarrow_{\mathsf{IV}} \emptyset}\,\mathsf{se}_{\mathsf{IV}} \qquad \frac{}{V \Rightarrow_{\mathsf{IV}} \{V^1\}}\,\mathsf{sv}_{\mathsf{IV}} \qquad \frac{M \mapsto_{\mathsf{v}} \overline{N} \quad N_i \Rightarrow_{\mathsf{IV}} \mathscr{D}_i}{M \Rightarrow_{\mathsf{IV}} \sum_{i=1}^{n} \frac{1}{n}\mathscr{D}_i}\,\mathsf{sm}_{\mathsf{IV}}.$$

As usual, $\overline{N}$ stands for the sequence $N_1, \dots N_n$. Since the relation $\Rightarrow_{\mathsf{IV}}$ is inductively defined, any proof of judgments involving $\Rightarrow_{\mathsf{IV}}$ is a *finite* object. If π is such a proof for $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, we write $\pi : M \Rightarrow_{\mathsf{IV}} \mathscr{D}$. The proof π is said to be *structurally smaller or equal to* another proof ρ if the number of rule instances in π is smaller or equal to the number of rule instances in ρ. In this case, we write $\pi \preceq \rho$.

First of all, observe that $\Rightarrow_{\mathsf{IV}}$ is not a function: many different distributions can be put in correspondence with the same term M. Moreover, there is one distribution $\mathscr{D}$ such that $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ always holds, independently on M, namely $\emptyset$. Actually, rule $\mathsf{se}_{\mathsf{IV}}$ allows you to "give up" while looking for a distribution for M and conclude that M is in relation with $\emptyset$. In other words, $\Rightarrow_{\mathsf{IV}}$ is not meant to be a way to attribute *one* distribution to every term, but rather to find all finitary approximants of the (unique) distribution we are looking for (see Def. 5.3).

The set of the distributions a term evaluates to is a directed set:

Lemma 5.2. *For every term M, if $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and $M \Rightarrow_{\mathsf{IV}} \mathscr{E}$, then there exists a distribution $\mathscr{F}$ such that $M \Rightarrow_{\mathsf{IV}} \mathscr{F}$ with $\mathscr{D} \leq \mathscr{F}$ and $\mathscr{E} \leq \mathscr{F}$.*

Proof. By induction on the structure of derivations for $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$.

- If $M \Rightarrow_{\mathsf{IV}} \emptyset$ then $\mathscr{F} = \mathscr{E}$;
- If $M \Rightarrow_{\mathsf{IV}} \{V^1\}$ we have that $\mathscr{E} = \emptyset$, or $\mathscr{E} = \{V^1\}$, then $\mathscr{F} = \mathscr{D}$;
- If $M \Rightarrow_{\mathsf{IV}} N_1, \dots, N_k$ and $N_i \Rightarrow_{\mathsf{IV}} \mathscr{D}_i$ for $i = 1, \dots, k$, there are some cases:
  - If $\mathscr{E} = \emptyset$, then $\mathscr{F} = \mathscr{D}$;
  - If $\mathscr{E} > \emptyset$, then $\mathscr{E} = \sum_{i=1}^{k} \frac{1}{k}\mathscr{E}_i$, where $N_i \Rightarrow_{\mathsf{IV}} \mathscr{E}_i$ for $i = 1, \dots, k$. Now, by inductive hypothesis, there exist distributions $\mathscr{F}_i$ such that $N_i \Rightarrow_{\mathsf{IV}} \mathscr{F}_i$ and $\mathscr{D}_i, \mathscr{E}_i \leq \mathscr{F}_i$ for $i = 1, \dots, k$. We have $\mathscr{F} = \sum_{i=1}^{k} \frac{1}{k}\mathscr{F}_i$, and by definition $\mathscr{D} \leq \mathscr{F}$ and $\mathscr{E} \leq \mathscr{F}$.

This concludes the proof. □

We are now ready to define what *the* small-step semantics of any term is:

Definition 5.3. The (call-by-value) small-step semantics of a λ-term $M \in \Lambda_{\oplus}$ is the distribution $\mathcal{S}_{\mathsf{IV}}(M)$ defined as $\sup_{M \Rightarrow_{\mathsf{IV}} \mathscr{D}} \mathscr{D}$.

Please observe that $\mathcal{S}_{\mathsf{IV}}(M)$ is always guaranteed to exist, because distributions form an ω-complete partial order and the set of those distributions such that $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ is a denumerable directed set, by Lemma 5.2.

Example 5.4. The term $(\lambda x.x)(\lambda x.x)$ evaluates to $\mathscr{F} = \emptyset$, by means of rules $\mathsf{se}_{\mathsf{IV}}$, and to $\mathscr{E} = \{\lambda x.x^1\}$, by means of rules $\mathsf{sm}_{\mathsf{IV}}$ and $\mathsf{sv}_{\mathsf{IV}}$. By definition, $\mathcal{S}_{\mathsf{IV}}(M) = \mathscr{E}$.

5.1.2. Coinductive CbV small-step semantics for divergence

Divergence is captured by another, *coinductively* defined binary relation $\Rightarrow^{\infty}$ between $\Lambda_{\oplus}$ and $\mathbb{R}_{[0,1]}$. Rules defining the underlying inference systems are those obtained from

$$\frac{}{V \Rightarrow^{\infty}_{0}}\ \mathsf{dv}_{\mathsf{v}} \qquad \frac{M \mapsto_{\mathsf{v}} \overline{N} \quad N_i \Rightarrow^{\infty}_{p_i}}{M \Rightarrow^{\infty}_{\sum_{i=1}^{n}\frac{1}{n}p_i}}\ \mathsf{dm}_{\mathsf{v}}$$

where $\overline{N}$ is $N_1, \ldots N_n$.

The relation $\Rightarrow^{\infty}$ deals naturally with infinite computations, being defined coinductively. This allows to derive the divergence probability p of a term in $\Lambda_{\oplus}$. Rules dv_{v} and dm_{v} can be read as follows: values diverge with probability 0, while probability of divergence for a term M is equal to the normalized sum of its reducts' probabilities of divergence. The following example stresses a crucial point:

Example 5.5. Let us consider the well-known diverging term $\Omega = \Delta\Delta$, where $\Delta = \lambda x.xx$. We would like to be able to prove that $\Omega \Rightarrow^{\infty}_{1}$, namely that Ω diverges with probability 1. Doing that formally requires proving that some sets of judgments A including $\Omega \Rightarrow^{\infty}_{1}$ is consistent with respect to $\Rightarrow^{\infty}$. Actually, we can choose A as $\{\Omega \Rightarrow^{\infty}_{1}\}$, since $\Omega \Rightarrow^{\infty}_{1}$ can be easily derived from itself by rule dm_{v}. The trouble is that in the same way we can derive $\Omega \Rightarrow^{\infty}_{p}$ for every possible $0 \leq p \leq 1$. So, in a sense, $\Rightarrow^{\infty}$ is inconsistent.

Example 5.5 shows that not all probabilities attributed to terms via $\Rightarrow^{\infty}$ are accurate. Actually, a good definition consists in taking the *divergence probability* of any term M, $\mathcal{D}_{\mathsf{CV}}(M)$ simply as $\sup_{M \Rightarrow^{\infty}_{p}} p$. As an example, $\mathcal{D}_{\mathsf{CV}}(\Omega) = 1$, since $M \Rightarrow^{\infty}_{1}$ and, clearly, one cannot go beyond 1.

5.1.3. Coinductive CbV small-step semantics for convergence

If one takes the inductive semantics from Section 5.1.1, drops $\mathsf{se}_{\mathsf{IV}}$ and interpret everything coinductively, what came out is an alternative semantics for convergence:

$$\frac{}{V \Rightarrow_{\mathsf{CV}} \{V^1\}}\ \mathsf{sv}_{\mathsf{CV}} \qquad \frac{M \mapsto_{\mathsf{v}} \overline{N} \quad N_i \Rightarrow_{\mathsf{CV}} \mathscr{D}_i}{M \Rightarrow_{\mathsf{CV}} \displaystyle\sum_{i=1}^{n} \frac{1}{n}\mathscr{D}_i}\ \mathsf{sm}_{\mathsf{CV}}.$$

Interpreting everything coinductively has the effect of allowing infinite computations to be modeled. But this allows to “promise” to reach a certain distribution without really being able to fulfil this:

Example 5.6. Consider again Ω. One would like to be able to prove that $\Omega \Rightarrow_{\mathsf{CV}} \emptyset$. Unfortunately, the coinductive interpretation of the formal system above contains $\Omega \Rightarrow_{\mathsf{CV}} \mathscr{D}$ *for every* distribution $\mathscr{D}$, as can be easily verified.

The solution to the problem highlighted by the example above is just defining the coinductive semantics of any term M just as $\mathcal{S}_{\mathsf{CV}}(M) = \inf_{M \Rightarrow_{\mathsf{CV}} \mathscr{D}} \mathscr{D}$. Clearly, $\mathcal{S}_{\mathsf{CV}}(\Omega) = \emptyset$. Observe that $\mathcal{S}_{\mathsf{CV}}(M)$ is guaranteed to exist because $(\mathcal{P}, \leq)$ is a meet-semilattice.

5.2. CbV big-step semantics

An alternative style to give semantics to programming languages is the so-called big-step semantics. Big-step semantics is more compositional than small-step: the meaning of a term can be obtained from the meanings of its sub-terms.

Probabilistic big step semantics for $\Lambda_{\oplus}$ can be given by a binary relation $\Downarrow_{\mathsf{CV}}$ between $\Lambda_{\oplus}$ and distributions. It is the coinductive interpretation of (all instances of) the following rules:

$$\frac{M \Downarrow_{\mathsf{CV}} \mathscr{D} \quad N \Downarrow_{\mathsf{CV}} \mathscr{E} \quad \{P\{x/V\} \Downarrow_{\mathsf{CV}} \mathscr{F}_{P,V}\}_{\lambda x.P \in \mathsf{S}(\mathscr{D}), V \in \mathsf{S}(\mathscr{E})}}{MN \Downarrow_{\mathsf{CV}} \sum_{\lambda x.P \in \mathsf{S}(\mathscr{D}), V \in \mathsf{S}(\mathscr{E})} \mathscr{D}(\lambda x.P) \cdot \mathscr{E}(V) \cdot \mathscr{F}_{P,V}} \; \mathsf{ba_v}$$

$$\frac{}{V \Downarrow_{\mathsf{CV}} \{V^1\}} \; \mathsf{bv_v} \qquad \frac{M \Downarrow_{\mathsf{CV}} \mathscr{D} \quad N \Downarrow_{\mathsf{CV}} \mathscr{E}}{M \oplus N \Downarrow_{\mathsf{CV}} \frac{1}{2} \cdot \mathscr{D} \cdot \sum \mathscr{E} + \frac{1}{2} \cdot \mathscr{E} \cdot \sum \mathscr{D}} \; \mathsf{bs_v}.$$

The most interesting rule is definitely $\mathsf{ba_v}$: to give semantics to an application MN, we first give semantics to M and N, obtaining two distributions $\mathscr{D}$ and $\mathscr{E}$, respectively. Then, for every $\lambda x.P$ in $\mathsf{S}(\mathscr{D})$ and for every V in $\mathsf{S}(\mathscr{E})$, we evaluate $P\{x/V\}$, obtaining some distributions $\mathscr{F}_{P,V}$. The meaning of MN is nothing more that the sum of all such distributions $\mathscr{F}_{P,V}$, each weighted by the probability of getting $\lambda x.P$ and V in $\mathscr{D}$ and $\mathscr{E}$, respectively. This way of defining the big-step semantics of applications can be made simpler, at the price of making the task of deriving semantic assertions harder, by replacing the premise $\{P\{x/V\} \Downarrow_{\mathsf{CV}} \mathscr{F}_{P,V}\}_{\lambda x.P \in \mathsf{S}(\mathscr{D}), V \in \mathsf{S}(\mathscr{E})}$ of rule $\mathsf{ba_v}$ with $\{P\{x/V\} \Downarrow_{\mathsf{CV}} \mathscr{F}_{P,V}\}_{P \in \Lambda_{\oplus}, V \in \mathsf{Val}}$.

Another interesting rule is $\mathsf{bs_v}$. Please observe how the distributions $\mathscr{D}$ and $\mathscr{E}$ must be normalized by $\sum \mathscr{E}$ and $\sum \mathscr{D}$ (respectively) when computing the result.

This reflects call-by-value evaluation: if any of M and N diverges, their sum must diverge, too.

Like $\Rightarrow_{\mathsf{IV}}$ and $\Rightarrow_{\mathsf{CV}}$, the relation $\Downarrow_{\mathsf{CV}}$ is not a function: many possible distributions can be assigned to the same term M, in particular when M (possibly) diverges. In particular, distributions which somehow overapproximate the "real one" can always be attributed to M by the rules above. Thus, we define *the* big-step semantics of each λ-term in the following way:

Definition 5.7. The (call-by-value) coinductive big-step semantics of a λ-term $M \in \Lambda_{\oplus}$ is the distribution $\mathcal{B}_{\mathsf{CV}}(M)$ defined as $\inf_{M \Downarrow_{\mathsf{CV}} \mathscr{D}} \mathscr{D}$.

Example 5.8. Let us consider again Ω, from Example 5.5. As it can be verified, $\Omega \Downarrow_{\mathsf{CV}} \mathscr{D}$ for all possible distributions $\mathscr{D}$. To formally prove that, we need to find a consistent set A (with respect to $\Downarrow_{\mathsf{CV}}$) containing all judgments in the form $\Omega \Downarrow_{\mathsf{CV}} \mathscr{D}$. A is actually the set

$$\{\Delta \Downarrow_{\mathsf{CV}} \{\Delta^1\}\} \cup \{\Omega \Downarrow_{\mathsf{CV}} \mathscr{D} \mid \mathscr{D} : \mathsf{Val} \to \mathbb{R}_{[0,1]}\}.$$

Clearly, A is consistent, since any judgment in A can be obtained from other judgments in A in one deduction step:

- $\Delta \Downarrow_{\mathsf{CV}} \{\Delta^1\}$ by rule $\mathsf{bv_v}$.
- From $\Delta \Downarrow_{\mathsf{CV}} \{\Delta^1\}$ and $\Omega \Downarrow_{\mathsf{CV}} \mathscr{D}$, one can easily derive $\Omega \Downarrow_{\mathsf{CV}} \mathscr{D}$ by rule $\mathsf{ba_v}$.

As a consequence, $\mathcal{B}_{\mathsf{CV}}(\Omega) = \inf_{\Omega \Downarrow_{\mathsf{CV}} \mathscr{D}} \mathscr{D} = \emptyset$.

One may wonder whether an *inductive* big step semantics can be defined for $\Lambda_{\oplus}$. The answer is positive: one only needs to add a rule attributing the empty distribution to any terms, in the spirit of the small-step inductive semantics from Section 5.1.1. In other words, we obtain the system

$$\frac{M \Downarrow_{\mathsf{CV}} \mathscr{D} \quad N \Downarrow_{\mathsf{IV}} \mathscr{E} \quad \{P\{x/V\} \Downarrow_{\mathsf{IV}} \mathscr{F}_{P,V}\}_{\lambda x.P \in \mathsf{S}(\mathscr{D}), V \in \mathsf{S}(\mathscr{E})}}{MN \Downarrow_{\mathsf{IV}} \displaystyle\sum_{\lambda x.P \in \mathsf{S}(\mathscr{D}), V \in \mathsf{S}(\mathscr{E})} \mathscr{D}(\lambda x.P) \cdot \mathscr{E}(V) \cdot \mathscr{F}_{P,V}}\ \mathsf{ba_{IV}}$$

$$\frac{}{M \Downarrow_{\mathsf{CV}} \{\emptyset\}}\ \mathsf{bv_{IV}} \qquad \frac{}{V \Downarrow_{\mathsf{IV}} \{V^1\}}\ \mathsf{bv_{IV}} \qquad \frac{M \Downarrow_{\mathsf{IV}} \mathscr{D} \quad N \Downarrow_{\mathsf{IV}} \mathscr{E}}{M \oplus N \Downarrow_{\mathsf{IV}} \frac{1}{2} \cdot \mathscr{D} \cdot \sum \mathscr{E} + \frac{1}{2} \cdot \mathscr{E} \cdot \sum \mathscr{D}}\ \mathsf{bs_{IV}}.$$

As can be expected, the inductive big step semantics $\mathcal{B}_{\mathsf{IV}}(M)$ of an term M is simply $\sup_{M \Downarrow_{\mathsf{IV}} \mathscr{D}}$. This is similar to the semantics considered by Jones [14].

In Table 1 we summarize the most important symbols introduced and defined in the previous sections for the call-by-value probabilistic semantics.

5.3. Divergence and convergence in CbV small-step semantics

In the last two sections, various operational semantics for both convergence and divergence have been introduced. Clearly, one would like them to be essentially equivalent, *i.e.*, one would like them to attribute the same meaning to programs.

In this section, divergence and convergence small-step semantics will be compared and proved equivalent: the probability of divergence $\mathcal{D}_{\mathsf{CV}}(M)$ of M obtained through $\Rightarrow^{\infty}$ will be proved to be equal to $1 - \sum \mathcal{S}_{\mathsf{IV}}(M)$. In Section 5.4 small-step and big-step semantics for convergence will be proved to produce identical outcomes.

The first step consists in proving that $1 - \sum \mathcal{S}_{\mathsf{IV}}(M)$ is a lower bound to divergence probability of any term M.

Theorem 5.9. *For every term M, $M \Rightarrow^{\infty}_{1-\sum \mathcal{S}_{\mathsf{IV}}(M)}$.*

Proof. We can prove that all judgments $M \Rightarrow^{\infty}_{1-\sum \mathcal{S}_{\mathsf{IV}}(M)}$ belong to the coinductive interpretation of the underlying formal system Φ. To do that, we need to prove that the set A of all these judgements is consistent, *i.e.* that $A \subseteq F_{\Phi}(A)$. This amounts to show that if $c \in A$, then there is a derivation for c whose immediate premises are themselves in A. Let's distinguish two cases:

- If M is a value V, then $M \Rightarrow^{\infty}_{0}$ and $\mathcal{S}_{\mathsf{IV}}(M) = \{V^1\}$. The thesis easily follows.
- If M is not a value, then $M \mapsto_{\mathsf{v}} \overline{N}$, with $\overline{N} = N_1, \ldots, N_n$. Now, consider the judgements $N_i \Rightarrow^{\infty}_{1-\sum \mathcal{S}_{\mathsf{IV}}(N_i)}$, with $i \in [1, n]$: they are all in the set A. Finally, consider the judgement $M \Rightarrow^{\infty}_{\sum_{i=1}^{n} \frac{1}{n}(1-\sum \mathcal{S}_{\mathsf{IV}}(N_i))}$: it is in $F_{\Phi}(A)$ because of the presence of rule $\mathsf{dm_v}$:
$$\frac{M \mapsto_{\mathsf{v}} \overline{N} \quad N_i \Rightarrow^{\infty}_{p_i}}{M \Rightarrow^{\infty}_{\sum_{i=1}^{n} \frac{1}{n} p_i}}\ \mathsf{dm_v}.$$
It remains to show that $\sum \mathcal{S}_{\mathsf{IV}}(M) = \sum_{i=1}^{n} \frac{1}{n} \sum \mathcal{S}_{\mathsf{IV}}(N_i)$:
$$\begin{aligned}\sum \mathcal{S}_{\mathsf{IV}}(M) &= \sum \sup_{M \Rightarrow_{\mathsf{IV}} \mathscr{D}} \mathscr{D} = \sum \sup_{N_i \Rightarrow_{\mathsf{IV}} \mathscr{E}_i} \left(\sum_{i=1}^{n} \frac{1}{n} \mathscr{E}_i \right) \\ &= \sum \left(\sum_{i=1}^{n} \frac{1}{n} \sup_{N_i \Rightarrow_{\mathsf{IV}} \mathscr{E}_i} \mathscr{E}_i \right) = \sum \left(\sum_{i=1}^{n} \frac{1}{n} \mathcal{S}_{\mathsf{IV}}(N_i) \right) \\ &= \sum_{i=1}^{n} \frac{1}{n} \sum \mathcal{S}_{\mathsf{IV}}(N_i).\end{aligned}$$

This concludes the proof. □

We need something more, however, namely that summing some divergence probability and the convergence probability obtained through convergence semantics, we cannot go beyond 1:

Proposition 5.10. *If $M \Rightarrow^{\infty}_{p}$ and $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, then $p + \sum \mathscr{D} \leq 1$.*

Proof. The proof goes by induction on the structure of the derivation for $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$:

- If the only rule in the derivation is
$$\frac{}{M \Rightarrow_{\mathsf{IV}} \emptyset}$$
then $\mathscr{D} = \emptyset$ and $\sum \mathscr{D} = 0$. As a consequence, $p + \sum \mathscr{D} = p \leq 1$ by definition of the divergence relation.
- If the only rule in the derivation is
$$\frac{}{V \Rightarrow_{\mathsf{IV}} \{V^1\}}$$
then $\sum \mathscr{D} = 1$ and $M \Rightarrow_0^\infty$. As a consequence $p + \sum \mathscr{D} \leq 1$.
- If the derivation has the form
$$\frac{M \mapsto_{\mathsf{v}} \overline{N} \quad N_i \Rightarrow_{\mathsf{IV}} \mathscr{D}_i}{M \Rightarrow_{\mathsf{IV}} \sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i}$$
then we apply the induction hypothesis on each $N_i \Rightarrow_{\mathsf{IV}} \mathscr{D}_i$ for $i \in [1, n]$. For each $i \in [1, n]$, $\sum \mathscr{D}_i \leq 1 - p_i$, where p_i is such that $N_i \Rightarrow_{p_1}^\infty$. Then we have:
$$\begin{aligned}\sum \mathscr{D} = \sum \sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i = \sum_{i=1}^{n} \frac{1}{n} \sum \mathscr{D}_i &\overset{i.h.}{\leq} \sum_{i=1}^{n} \frac{1}{n}(1 - p_i) \\ = 1 - \sum_{i=1}^{n} \frac{1}{n} p_i = 1 - p.\end{aligned}$$

This concludes the proof. □

Everything can be now glued together as follows, exploiting the density of real numbers:

Corollary 5.11. *For every M, $\mathcal{D}_{\mathsf{CV}}(M) + \sum \mathcal{S}_{\mathsf{IV}}(M) = 1$.*

Proof. $\mathcal{D}_{\mathsf{CV}}(M) + \sum \mathcal{S}_{\mathsf{IV}}(M) \geq 1$ by Theorem 5.9. Suppose, by way of contradiction, that $\mathcal{D}_{\mathsf{CV}}(M) + \sum \mathcal{S}_{\mathsf{IV}}(M) > 1$. This implies $M \Rightarrow_p^\infty$ where $p + \sum \mathcal{S}_{\mathsf{IV}}(M) > 1$. This, in turn, implies that $p + \sum \mathscr{D} > 1$ for some $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$. And this is not possible by Proposition 5.10. □

5.4. Relating the various definitions for convergence

Our goal in this section consists in proving that the four distinct definitions for the semantics of a term in $\Lambda_\oplus$ (inductive and coinductive, big-step and small-step semantics) attribute identical meanings to any term. One possibility could be to proceed by "proving" three edges of the following diagram:

$$\begin{tikzcd}
\mathcal{S}_{\mathsf{IV}} \arrow[r, dash] \arrow[d, dash] & \mathcal{S}_{\mathsf{CV}} \arrow[d, dash] \\
\mathcal{B}_{\mathsf{IV}} \arrow[r, dash] & \mathcal{B}_{\mathsf{CV}}
\end{tikzcd}$$

The vertical edges relate two formulations given with identical "inductive flavors" but differing as to whether they are big-step or small-step. Conversely, horizontal edges put in correspondence two formulations which are both big-step or small step, but which differ as to which kind of interpretation is taken over (essentially) the same set of rules. Horizontal edges are definitely interesting, but vertical ones are also important. In order to avoid tedious and long (but not necessarily informative) proofs, we only prove the diagonal edge shown below in this paper:

$$\begin{tikzcd}
\mathcal{S}_{\mathsf{IV}} \arrow[dr, squiggly, dash] & \mathcal{S}_{\mathsf{CV}} \\
\mathcal{B}_{\mathsf{IV}} & \mathcal{B}_{\mathsf{CV}}
\end{tikzcd}$$

This proof shows the difficulties of both "vertical" and "horizontal" edges.

Before embarking in the proof of this result, a brief explanation of the architecture of the proof is useful. Consider any term M, and define two sets of distributions $M^{\mathcal{S}}_{\mathsf{IV}}$ and $M^{\mathcal{B}}_{\mathsf{CV}}$ as, respectively, the sets of probability distributions which can be attributed to M in small-step semantics and big-step semantics:

$$M^{\mathcal{S}}_{\mathsf{IV}} = \{\mathscr{D} \mid M \Rightarrow_{\mathsf{IV}} \mathscr{D}\};$$
$$M^{\mathcal{B}}_{\mathsf{CV}} = \{\mathscr{D} \mid M \Downarrow_{\mathsf{CV}} \mathscr{D}\}.$$

We proceed in two steps:

- First of all, we prove that big-step semantics *dominates* small-step semantics, namely that $\mathscr{D} \geq \mathscr{E}$ whenever $\mathscr{D} \in M^{\mathcal{B}}_{\mathsf{CV}}$ and $\mathscr{E} \in M^{\mathcal{S}}_{\mathsf{IV}}$. This way, we are sure that $\mathcal{B}_{\mathsf{CV}}(M) \geq \mathcal{S}_{\mathsf{IV}}(M)$. Details are in Section 5.4.1 below.
- Then, we prove that small step semantics can itself be derived using big-step semantics, namely that $\mathcal{S}_{\mathsf{IV}}(M) \in M^{\mathcal{B}}_{\mathsf{CV}}$. This way, we immediately obtain that $\mathcal{B}_{\mathsf{CV}}(M) \leq \mathcal{S}_{\mathsf{IV}}(M)$, since $\mathcal{B}_{\mathsf{CV}}(M)$ is a minorizer of distributions in $M^{\mathcal{B}}_{\mathsf{CV}}$. Details are in Section 5.4.2.

As a consequence, $\mathcal{S}_{\mathsf{IV}}(M) = \mathcal{B}_{\mathsf{CV}}(M)$. Figure 1 illustrates the architecture of the proof.

5.4.1. Big-step dominates small-step

The fact any distribution obtained through big-step semantics is bigger than any distribution obtained through small-step semantics can be proved by induction on the structure of (finite!) derivations for the latter. For that purpose, given any

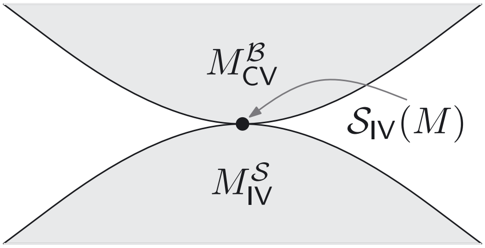
$M^{\mathcal{B}}_{\mathsf{CV}}$
$\mathcal{S}_{\mathsf{IV}}(M)$
$M^{\mathcal{S}}_{\mathsf{IV}}$

Figure 1. The overall picture.

derivation π of a statement in the form $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, its *complexity* is defined as follows, by induction on the structure of π:

- If the only rule used to build π is either $\mathsf{se}_{\mathsf{IV}}$ or $\mathsf{sv}_{\mathsf{IV}}$, then the complexity of π is 0;
- If the last rule in π is $\mathsf{sm}_{\mathsf{IV}}$ and the immediate subderivations of π are $\rho_1, \ldots, \rho_n$, then the complexity of π is defined as $1 + \max\{c_1, \ldots, c_n\}$, where c_i is the complexity of ρ_i for every $i \in \{1, \ldots, n\}$.

Given two derivations π and ρ, we write $\pi \preceq \rho$ when the complexity of π is smaller or equal to the complexity of ρ.

Before proceeding, however, it is necessary to prove that whenever $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and M is, say, a sum $N \oplus L$, then appropriate judgments $N \Rightarrow_{\mathsf{IV}} \mathscr{E}$ and $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ can be derived. Similarly for applications. The following lemma formalizes these ideas and it is a technical tool for Proposition 5.14.

Lemma 5.12. *If $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, then at least one of the following hold:*

1. $\mathscr{D} = \emptyset$;
2. *M is a value V and $\mathscr{D} = \{V^1\}$;*
3. *M is an application NL and there are (finite) distributions $\mathscr{E}$ and $\mathscr{F}$ and for every $\lambda x.P \in \mathsf{S}(\mathscr{E})$ and $V \in \mathsf{S}(\mathscr{F})$ a distribution $\mathscr{G}_{P,V}$ such that:*
   1. *$\rho : N \Rightarrow_{\mathsf{IV}} \mathscr{E}$, $\xi : L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ and $\mu_{P,V} : P\{x/V\} \Rightarrow_{\mathsf{IV}} \mathscr{G}_{P,V}$;*
   2. $\rho, \xi, \mu_{P,V} \prec \pi$;
   3. $\mathscr{D} \leq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}(\lambda x.P) \cdot \mathscr{F}(V) \cdot \mathscr{G}_{P,V}$.
4. *M is a sum $N \oplus L$ and there are (finite) distributions $\mathscr{E}$ and $\mathscr{F}$ where:*
   1. *$\rho : N \Rightarrow_{\mathsf{IV}} \mathscr{E}$ and $\xi : L \Rightarrow_{\mathsf{IV}} \mathscr{F}$;*
   2. $\rho, \xi \prec \pi$;
   3. $\mathscr{D} \leq \frac{1}{2} \cdot \mathscr{E} \cdot (\sum \mathscr{F}) + \frac{1}{2} \cdot \mathscr{F} \cdot (\sum \mathscr{E})$.

Proof. First of all, let us prove the following auxiliary lemma:

Lemma 5.13. *If I is finite and nonempty, and $\pi_i : M \Rightarrow_{\mathsf{IV}} \mathscr{D}_i$ for every $i \in I$, then there is $\rho : M \Rightarrow_{\mathsf{IV}} \sup_{i \in I} \mathscr{D}_i$. Moreover $\rho \preceq \pi_i$ for some $i \in I$.*

The proof of Lemma 5.13 goes by induction on the structure of the proofs in the family $\{\pi_i\}_{i \in I}$ (which can actually be done, since I is finite). Let's now go back to Lemma 5.12. This is an induction on derivations for $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$. The cases where

π is obtained by the rules without premises are trivial. So, we can assume that π has the form

$$\frac{M \mapsto_{\mathsf{v}} \overline{Q} \quad \pi_i : Q_i \Rightarrow_{\mathsf{IV}} \mathscr{D}_i}{M \Rightarrow_{\mathsf{IV}} \displaystyle\sum_{i=1}^{n} \frac{1}{n}\mathscr{D}_i}\;\mathsf{sm}_{\mathsf{IV}}$$

where $\overline{Q}$ is $Q_1, \ldots, Q_n$. Let's distinguish some cases depending on how the premise $M \mapsto_{\mathsf{v}} \overline{Q}$ is derived:

- Suppose $M = NL$ and that $N \mapsto_{\mathsf{v}} \overline{R}$. Then $\overline{Q} = \overline{R}L$. From the induction hypothesis applied to the derivations $\pi_i : R_i L \Rightarrow_{\mathsf{IV}} \mathscr{D}_i$, we obtain that
  - Either all of the $\mathscr{D}_i$ is $\emptyset$. In this case $\mathscr{D}$ is itself $\emptyset$, and a derivation for $M \Rightarrow_{\mathsf{IV}} \emptyset$ can be defined.
  - Or there is at least one among the derivation π_i to which case 3 applies. Suppose that $k_1, \ldots, k_m$ are the indices in $[1, n]$ to which case 3 can be applied. This means that for every in i in $\{1, \ldots, m\}$, there is a distribution $\mathscr{E}_i$, $\mathscr{F}_i$ and for every $\lambda x.P \in \mathsf{S}(\mathscr{E}_i)$ and $V \in \mathsf{S}(\mathscr{F}_i)$ a distribution $\mathscr{H}^i_{P^i,V^i}$ such that:

    $$\begin{aligned} \rho_i &: R_{k_i} \Rightarrow_{\mathsf{IV}} \mathscr{E}_i; \\ \xi_i &: L \Rightarrow_{\mathsf{IV}} \mathscr{F}_i; \\ \phi^i_{P,V} &: P\{V/x\} \Rightarrow_{\mathsf{IV}} \mathscr{H}^i_{P,V}. \end{aligned}$$

    Moreover, $\rho_i, \xi_i, \phi^i_{g,h} \prec \pi_t$, and

    $$\mathscr{D}_{k_i} \leq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}_i), V \in \mathsf{S}(\mathscr{F}_i)} \mathscr{E}_i(P) \cdot \mathscr{F}_i(V) \cdot \mathscr{H}^i_{P,V}.$$

    Now, define $\mathscr{E}$ as the distribution

    $$\sum_{i=1}^{m} \frac{1}{n}\mathscr{E}_i.$$

    Note that $\mathsf{S}(\mathscr{E}) = \bigcup_{i=1}^{m} \mathsf{S}(\mathscr{E}_i)$. Clearly, a derivation ρ for $N \Rightarrow_{\mathsf{IV}} \mathscr{E}$ can be defined such that $\rho \prec \pi$: simply construct it from the derivations ρ_i and some derivations for $R_j \Rightarrow_{\mathsf{IV}} \emptyset$. Moreover, a derivation ξ of $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ where $\mathscr{F} = \sup_{i=1}^{m} \mathscr{F}_i$ can be defined such that $\xi \prec \pi$: use Lemma 5.13. Observe that $\mathsf{S}(\mathscr{F}) = \bigcup_{i=1}^{m} \mathsf{S}(\mathscr{F}_i)$. Similarly, derivations $\mu_{P,V}$ can be defined for every $\lambda x.P \in \mathsf{S}(\mathscr{E})$ and for every $V \in \mathsf{S}(\mathscr{F})$ in such a way that $\mu_{P,V} \prec \pi$,

$\mu_{P,V} : P\{V/x\} \Rightarrow_{\mathsf{IV}} \mathscr{G}_{P,V}$ and $\mathscr{G}_{P,V} = \sup_{i=1}^{m} \mathscr{H}^{i}_{P,V}$. Now:

$$\begin{aligned}
\mathscr{D} &= \sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i \leq \sum_{i=1}^{m} \frac{1}{n} \mathscr{D}_{k_i} \\
&\leq \sum_{i=1}^{m} \frac{1}{n} \left(\sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}_i), V \in \mathsf{S}(\mathscr{F}_i)} \mathscr{E}_i(\lambda x.P) \cdot \mathscr{F}_i(V) \cdot \mathscr{H}^{i}_{P,V} \right) \\
&\leq \sum_{i=1}^{m} \frac{1}{n} \left(\sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}_i(\lambda x.P) \cdot \mathscr{F}(V) \cdot \mathscr{H}^{i}_{P,V} \right) \\
&= \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})} \sum_{i=1}^{m} \frac{1}{n} \left(\mathscr{E}_i(\lambda x.P) \cdot \mathscr{F}(V) \cdot \mathscr{H}^{i}_{P,V} \right) \\
&\leq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}_i), V \in \mathsf{S}(\mathscr{F})} \sum_{i=1}^{m} \frac{1}{n} \left(\mathscr{E}_i(\lambda x.P) \cdot \mathscr{F}(V) \cdot \mathscr{G}_{P,V} \right) \\
&= \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}(\lambda x.P) \cdot \mathscr{F}(V) \cdot \mathscr{G}_{P,V}.
\end{aligned}$$

- Suppose $M = NL$ and that $L \mapsto_{\mathsf{v}} \overline{R}$. Then $\overline{Q} = N\overline{R}$. This case is very similar to the previous one. Note that since we reduce in a call by value setting, then $N \in \mathsf{Val}$ and by means of small-step semantics rules, $N \Rightarrow_{\mathsf{IV}} \{N^1\}$.
- Suppose $M = NL$ and that we are in presence of a redex, *i.e.* N is in the form $\lambda x.R$ and L is a value and $L \Rightarrow_{\mathsf{IV}} \{L^1\}$. Then $\overline{Q}$ is the unary sequence $R\{L/x\}$. The thesis easily follows by induction, taking $\overline{\pi} : R\{L/x\} \Rightarrow_{\mathsf{IV}} \overline{\mathscr{D}}$ as premise, where $\overline{\mathscr{D}} = \sup_{R\{L/x\} \Rightarrow_{\mathsf{IV}} \mathscr{D}_k} \mathscr{D}_k$.
- Suppose $M = N \oplus L$ and that $N \mapsto_{\mathsf{v}} \overline{R}$. Then $\overline{Q} = \overline{R} \oplus L$. From the induction hypothesis applied to the derivations $\pi_i : R_i \oplus L \Rightarrow_{\mathsf{IV}} \mathscr{D}_i$, we have that:
  - Either all of the distributions $\mathscr{D}_i$ is $\emptyset$, then $\mathscr{D}$ is $\emptyset$ itself and a derivation for $M \Rightarrow_{\mathsf{IV}} \emptyset$ can be defined.
  - Or there is at least one derivation among π to which case 4 applies. Suppose that $k_1, \ldots, k_m$ are the indices in $[1, n]$ to which case 3 can be applied. For every in i in $\{1, \ldots, m\}$, there are two distributions $\mathscr{E}_i$ and $\mathscr{F}_i$. Moreover, again by induction hypothesis, we have

$$\begin{aligned}
\rho_i &: R_{k_i} \Rightarrow_{\mathsf{IV}} \mathscr{E}_i \\
\xi_i &: L \Rightarrow_{\mathsf{IV}} \mathscr{F}_i.
\end{aligned}$$

  Moreover, $\rho_i, \xi_i \prec \pi_i$ and $\mathscr{D}_{k_i} \leq \frac{1}{2} \cdot \mathscr{E}_i \cdot (\sum \mathscr{F}_i) + \frac{1}{2} \cdot \mathscr{F}_i \cdot (\sum \mathscr{E}_i)$. Let us define $\mathscr{E}$ as the distribution $\sum_{i=1}^{m} \frac{1}{n} \mathscr{E}_i$ and observe that a derivation $\rho : N \Rightarrow_{\mathsf{IV}} \mathscr{E}$ such that $\rho \prec \pi$ can be defined from the derivation ρ_i. Moreover, a derivation $\xi : L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ can be defined such that $\xi \prec \pi$, taking

$\mathscr{F} = \sup_{i=1}^{m} \mathscr{F}_i$ and applying Lemma 5.13. Finally we have:

$$\begin{aligned}
\mathscr{D} &= \sum_{i=1}^{m} \frac{1}{n} \mathscr{D}_{k_i} \\
&\leq \sum_{i=1}^{m} \frac{1}{n} \left(\frac{1}{2} \cdot \mathscr{E}_i \cdot \left(\sum \mathscr{F}_i \right) + \frac{1}{2} \cdot \mathscr{F}_i \cdot \left(\sum \mathscr{E}_i \right) \right) \\
&= \sum_{i=1}^{m} \frac{1}{n} \left(\frac{1}{2} \cdot \mathscr{E}_i \cdot \left(\sum \mathscr{F}_i \right) \right) + \sum_{i=1}^{m} \frac{1}{n} \left(\frac{1}{2} \cdot \mathscr{F}_i \cdot \left(\sum \mathscr{E}_i \right) \right) \\
&\leq \sum_{i=1}^{m} \frac{1}{n} \left(\frac{1}{2} \cdot \mathscr{E}_i \cdot \left(\sum \mathscr{F} \right) \right) + \sum_{i=1}^{m} \frac{1}{n} \left(\frac{1}{2} \cdot \mathscr{F} \cdot \left(\sum \mathscr{E}_i \right) \right) \\
&= \frac{1}{2} \cdot \sum_{i=1}^{m} \frac{1}{n} \mathscr{E}_i \cdot \left(\sum \mathscr{F} \right) + \frac{1}{2} \cdot \mathscr{F} \cdot \left(\sum_{i=1}^{m} \frac{1}{n} \left(\sum \mathscr{E}_i \right) \right) \\
&= \frac{1}{2} \cdot \sum_{i=1}^{m} \frac{1}{n} \mathscr{E}_i \cdot \left(\sum \mathscr{F} \right) + \frac{1}{2} \cdot \mathscr{F} \cdot \left(\sum \left(\sum_{i=1}^{m} \frac{1}{n} \mathscr{E}_i \right) \right) \\
&= \frac{1}{2} \cdot \mathscr{E} \cdot \sum \mathscr{F} + \frac{1}{2} \cdot \mathscr{F} \cdot \sum \mathscr{E}
\end{aligned}$$

where for each $i \in [1, m]$, $\sum \mathscr{F}_i \leq \sum \mathscr{F}$ holds because $\mathscr{F}$ is the least upper bound of the $\mathscr{F}_i$.

- Suppose $M = Z \oplus L$ ($Z \in \mathsf{Val}$) and that $L \mapsto_{\mathsf{v}} \overline{R}$. Then $\overline{Q} = Z \oplus \overline{R}$. Similar to the previous case.
- Suppose $M = V \oplus W$ and that $M \mapsto_{\mathsf{v}} V, W$. In this case the subderivations which we are looking for are $\pi : V \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and $\rho : W \Rightarrow_{\mathsf{IV}} \mathscr{E}$.

This concludes the proof. □

Now, suppose that $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and $M \Downarrow_{\mathsf{CV}} \mathscr{E}$. Lemma 5.12 provides all what is needed to "unfold" the hypothesis $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and obtain judgments matching exactly those coming from $M \Downarrow_{\mathsf{CV}} \mathscr{E}$. We easily get:

Proposition 5.14. *If $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and $M \Downarrow_{\mathsf{CV}} \mathscr{E}$, then $\mathscr{D} \leq \mathscr{E}$.*

Proof. By induction on the structure of a proof for $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, applying Lemma 5.12 and doing some case analysis based on its outcome:

- If $\mathscr{D} = \emptyset$, then $\mathscr{D} \leq \mathscr{E}$ trivially.
- If $M = V$ and $\mathscr{D} = \{V^1\}$, then $\mathscr{E} = \mathscr{D}$, because the only rule for values in the big step semantic is
$$\frac{}{V \Downarrow_{\mathsf{CV}} \{V\}}\ \mathsf{bv_v}.$$
- If M is an application NL and distributions $\mathscr{F}, \mathscr{G}, \mathscr{H}_{P,V}$ and derivations $\rho, \xi, \mu_{P,V}$ exist as in Lemma 5.12, we can observe, by induction hypothesis, that there exist distributions $\mathscr{I}$ and $\mathscr{J}$ such that $N \Downarrow_{\mathsf{CV}} \mathscr{I}$, $L \Downarrow_{\mathsf{CV}} \mathscr{J}$ where $\mathscr{I} \geq \mathscr{F}$ and $\mathscr{J} \geq \mathscr{G}$. Then we have that $\mathsf{S}(\mathscr{I}) \supseteq \mathsf{S}(\mathscr{F})$ and

$\mathsf{S}(\mathscr{J}) \supseteq \mathsf{S}(\mathscr{G})$. Now, for every $\lambda x.P \in \mathsf{S}(\mathscr{F})$ and $V \in \mathsf{S}(\mathscr{G})$, suppose $\mathscr{K}_{P,V}$ is such that $P\{V/x\} \Downarrow_{\mathsf{CV}} \mathscr{K}_{P,V}$. Again by induction hypothesis, we obtain that $\mathscr{K}_{P,V} \geq \mathscr{H}_{P,V}$. Finally:

$$\begin{aligned}
\mathscr{D} &\leq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{F}), V \in \mathsf{S}(\mathscr{G})} \mathscr{F}(\lambda x.P) \cdot \mathscr{G}(V) \cdot \mathscr{H}_{P,V} \\
&\leq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{I}), V \in \mathsf{S}(\mathscr{J})} \mathscr{F}(\lambda x.P) \cdot \mathscr{G}(V) \cdot \mathscr{H}_{P,V} \\
&\leq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{I}), V \in \mathsf{S}(\mathscr{J})} \mathscr{I}(\lambda x.P) \cdot \mathscr{J}(V) \cdot \mathscr{K}_{P,V} \\
&= \mathscr{E}.
\end{aligned}$$

- If M is $N \oplus L$, we can proceed exactly as in the previous case. In fact, there exist distributions $\mathscr{F}, \mathscr{G}$ and derivations ρ, ξ as in Lemma 5.12, and by induction hypothesis we can observe that for any distributions $\mathscr{I}$ and $\mathscr{J}$ such that $N \Downarrow_{\mathsf{CV}} \mathscr{I}$ and $L \Downarrow_{\mathsf{CV}} \mathscr{J}$, $\mathscr{I} \geq \mathscr{F}$ and $\mathscr{J} \geq \mathscr{G}$ hold.
  We can take $\mathscr{I}$ such that $\mathsf{S}(\mathscr{I}) \supseteq \mathsf{S}(\mathscr{F})$ and $\mathscr{I}(V) \geq \mathscr{F}(V)$ for each $V \in \mathsf{S}(\mathscr{F})$, and we can take $\mathscr{J}$ such that $\mathsf{S}(\mathscr{J}) \supseteq \mathsf{S}(\mathscr{G})$ and $\mathscr{J}(V) \geq \mathscr{G}(V)$ for each $V \in \mathsf{S}(\mathscr{G})$. Then we have:

$$\begin{aligned}
\mathscr{D} &\leq \frac{1}{2} \cdot \mathscr{F} \cdot \left(\sum \mathscr{G}\right) + \frac{1}{2} \cdot \mathscr{G} \cdot \left(\sum \mathscr{F}\right) \\
&\leq \frac{1}{2} \cdot \mathscr{I} \cdot \left(\sum \mathscr{J}\right) + \frac{1}{2} \cdot \mathscr{J} \cdot \left(\sum \mathscr{I}\right) \\
&= \mathscr{E}.
\end{aligned}$$

This concludes the proof. □

5.4.2. Small-step is in big-step

Whenever $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and M is not a value, one can always "decompose" $\mathscr{D}$ and find some judgments about the immediate subterms of M. This is Lemma 5.12. If we want to prove that *the* small-step semantics $\mathcal{S}_{\mathsf{IV}}(M)$ of M can be itself attributed to M in the big-step case, we should somehow prove the converse, namely that judgments about the immediate subterms of M can be packaged into an analogous judgment for M:

Lemma 5.15. *Let $M \in \Lambda_{\oplus}$ be any term. Then:*

1. *If M is a value V and $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, then $\mathscr{D} \leq \{V^1\}$.*
2. *If M is an application NL, $N \Rightarrow_{\mathsf{IV}} \mathscr{E}$, $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ and*

$$\{P\{V/x\} \Rightarrow_{\mathsf{IV}} \mathscr{G}_{P,V}\}_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})},$$

then there exists a distribution $\mathscr{D}$ such that $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and

$$\mathscr{D} \geq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}(\lambda x.P) \cdot \mathscr{F}(V) \cdot \mathscr{G}_{P,V}.$$

3. *If M is a sum $N \oplus L$, $N \Rightarrow_{\mathsf{IV}} \mathscr{E}$, $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ then there exist a distribution $\mathscr{D}$ such that $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and $\mathscr{D} \geq \frac{1}{2} \cdot \mathscr{E} \cdot \sum \mathscr{F} + \frac{1}{2} \cdot \mathscr{F} \cdot \sum \mathscr{E}$.*

Proof. Let us prove the three statements separately:

1. If M is a value V, then the only possible judgments involving M are $M \Rightarrow_{\mathsf{IV}} \{V^1\}$ and $M \Rightarrow_{\mathsf{IV}} \emptyset$. The thesis trivially holds.
2. If M is an application NL, then we prove statement 2 by induction on the derivations $N \Rightarrow_{\mathsf{IV}} \mathscr{E}$ and $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$. Let's distinguish some cases:
   - If $N \Rightarrow_{\mathsf{IV}} \emptyset$, then $\sum \mathscr{E}(\lambda x.P) \cdot \mathscr{F}(V) \cdot \mathscr{G}_{P,V} = \emptyset$. We derive $M \Rightarrow_{\mathsf{IV}} \emptyset$ by means of the rule $\mathsf{se}_{\mathsf{IV}}$ and the thesis holds.
   - If $N \Rightarrow_{\mathsf{IV}} \mathscr{E} \neq \emptyset$, suppose that the least rule applied in the derivation is

$$\frac{N \mapsto_{\mathsf{v}} \overline{R} \quad R_i \Rightarrow_{\mathsf{IV}} \mathscr{E}_i}{N \Rightarrow_{\mathsf{IV}} \displaystyle\sum_{i=1}^{n} \frac{1}{n} \mathscr{E}_i}.$$

   Note that $\mathscr{E} = \sum_{i=1}^{n} \frac{1}{n} \mathscr{E}_i$. It is possible to apply the induction hypothesis on every $R_i L$: if $R_i \Rightarrow_{\mathsf{IV}} \mathscr{E}_i$, $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ and $P\{V/x\} \Rightarrow_{\mathsf{IV}} \mathscr{G}_{P,V}$ (for $\lambda x.P \in \mathsf{S}(\mathscr{E}_i) \subseteq \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})$) follows that $R_i L \Rightarrow_{\mathsf{IV}} \mathscr{H}_i$ with $\mathscr{H}_i \geq \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}_i), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}_i(\lambda x.P) \mathscr{F}(V) \mathscr{G}_{P,V}$. We are able to construct a derivation $\mathscr{D}$ in the following way:

$$\frac{NL \mapsto_{\mathsf{v}} \overline{R}L \quad R_i L \Rightarrow_{\mathsf{IV}} \mathscr{H}_i}{NL \Rightarrow_{\mathsf{IV}} \displaystyle\sum_{i=1}^{n} \frac{1}{n} \mathscr{H}_i}.$$

   And finally we have:

$$\begin{aligned}
\mathscr{D} = \sum_{i=1}^{n} \frac{1}{n} \mathscr{H}_i &\geq \sum_{i=1}^{n} \frac{1}{n} \left(\sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}_i), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}_i(\lambda x.P) \mathscr{F}(V) \mathscr{G}_{P,V} \right) \\
&= \sum_{i=1}^{n} \frac{1}{n} \left(\sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}_i(\lambda x.P) \mathscr{F}(V) \mathscr{G}_{P,V} \right) \\
&= \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F}_i)} \left(\sum_{i=1}^{n} \frac{1}{n} \left(\mathscr{E}_i(\lambda x.P) \right) \mathscr{F}_i(V) \mathscr{G}_{P,V} \right) \\
&= \sum_{\lambda x.P \in \mathsf{S}(\mathscr{E}), V \in \mathsf{S}(\mathscr{F})} \mathscr{E}(\lambda x.P) \mathscr{F}(V) \mathscr{G}_{P,V}.
\end{aligned}$$

   - Other cases follows can be handled similarly to the previous one.
3. M is a sum $N \oplus L$. We prove the result by induction on the derivations $N \Rightarrow_{\mathsf{IV}} \mathscr{E}$ and $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$.

- If $N \Rightarrow_{\mathsf{IV}} \emptyset$, then $\frac{1}{2} \cdot \emptyset \cdot \sum \mathscr{F} + \frac{1}{2} \cdot \mathscr{F} \cdot \sum \emptyset = \emptyset$. We derive $M \Rightarrow_{\mathsf{IV}} \emptyset$ by means of the rule $\mathsf{se}_{\mathsf{IV}}$ and the thesis holds.
- If N is a value V, the only interesting case is the one in which L is not a value and also $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ with $\mathscr{F} \neq \emptyset$ (in fact if $L \Rightarrow_{\mathsf{IV}} \emptyset$ we are in the previous case and if L is a value the proof is trivial). Then $M = V \oplus L$. Let us consider the derivation of the judgment $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$: the last rule applied in the derivation is
$$\frac{L \mapsto_{\mathsf{v}} \overline{R} \quad R_i \Rightarrow_{\mathsf{IV}} \mathscr{F}_i}{L \Rightarrow_{\mathsf{IV}} \sum_{i=1}^{n} \frac{1}{n} \mathscr{F}_i}$$
where $\mathscr{F} = \sum_{i=1}^{n} \frac{1}{n} \mathscr{F}_i$. Observe that $M \mapsto_{\mathsf{v}} V \oplus \overline{R}$. We can apply the induction hypothesis to each $L \Rightarrow_{\mathsf{IV}} \sum_{i=1}^{n} \frac{1}{n} \mathscr{F}_i$ and to $V \Rightarrow_{\mathsf{IV}} \{V^1\}$, and we obtain
$$\mathscr{D}_i \geq \frac{1}{2} \cdot \{V^1\} \cdot \sum \mathscr{F}_i + \frac{1}{2} \cdot \mathscr{F}_i.$$
We are able to construct a derivation in the following way:
$$\frac{V \oplus L \mapsto_{\mathsf{v}} V \oplus \overline{R} \quad V \oplus R_i \Rightarrow_{\mathsf{IV}} \mathscr{D}_i}{M \Rightarrow_{\mathsf{IV}} \sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i}.$$
Finally we have:
$$\begin{aligned}\sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i &\geq \sum_{i=1}^{n} \frac{1}{n} \left(\frac{1}{2} \cdot \{V^1\} \cdot \sum \mathscr{F}_i + \frac{1}{2} \cdot \mathscr{F}_i \right) \\ &= \sum_{i=1}^{n} \frac{1}{n} \left(\frac{1}{2} \cdot \{V^1\} \cdot \sum \mathscr{F}_i \right) + \sum_{i=1}^{n} \frac{1}{n} \left(\frac{1}{2} \cdot \mathscr{F}_i \right) \\ &= \frac{1}{2} \cdot \{V^1\} \cdot \left(\sum \sum_{i=1}^{n} \frac{1}{n} \mathscr{F}_i \right) + \frac{1}{2} \cdot \sum_{i=1}^{n} \frac{1}{n} \mathscr{F}_i \\ &= \frac{1}{2} \cdot \{V^1\} \cdot \left(\sum \mathscr{F} \right) + \frac{1}{2} \cdot \mathscr{F}.\end{aligned}$$
- If $N \Rightarrow_{\mathsf{IV}} \mathscr{E}$, suppose that the least rule applied in the derivation is
$$\frac{N \mapsto_{\mathsf{v}} \overline{R} \quad R_i \Rightarrow_{\mathsf{IV}} \mathscr{E}_i}{N \Rightarrow_{\mathsf{IV}} \sum_{i=1}^{n} \frac{1}{n} \mathscr{E}_i}.$$
Note that $\mathscr{E} = \sum_{i=1}^{n} \frac{1}{n} \mathscr{E}_i$. It is possible to apply the induction hypothesis on the single $R_i \oplus L$; then, if $R_i \Rightarrow_{\mathsf{IV}} \mathscr{E}_i$, $L \Rightarrow_{\mathsf{IV}} \mathscr{F}$ we have that $R_i \oplus L \Rightarrow_{\mathsf{IV}} \mathscr{D}_i$

with $\mathscr{D}_i \geq \frac{1}{2} \cdot \mathscr{E}_i \cdot \sum \mathscr{F}_i + \frac{1}{2} \cdot \mathscr{F}_i \cdot \sum \mathscr{E}_i$. We construct a derivation in the following way:

$$\frac{N \oplus L \mapsto_{\mathsf{v}} \overline{R} \oplus L \quad R_i \oplus L \Rightarrow_{\mathsf{IV}} \mathscr{D}_i}{M \Rightarrow_{\mathsf{IV}} \sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i}.$$

Finally we have:

$$\begin{aligned}
\sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i &\geq \sum_{i=1}^{n} \frac{1}{n} \left(\frac{1}{2} \mathscr{E}_i \cdot \sum \mathscr{F}_i + \frac{1}{2} \mathscr{F}_i \cdot \sum \mathscr{E}_i \right) \\
&= \sum_{i=1}^{n} \frac{1}{n} \left(\frac{1}{2} \mathscr{E}_i \cdot \sum \mathscr{F}_i \right) + \sum_{i=1}^{n} \frac{1}{n} \left(\frac{1}{2} \mathscr{F}_i \cdot \sum \mathscr{E}_i \right) \\
&= \frac{1}{2} \sum_{i=1}^{n} \frac{1}{n} \left(\mathscr{E}_i \cdot \sum \mathscr{F}_i \right) + \frac{1}{2} \sum_{i=1}^{n} \frac{1}{n} \left(\mathscr{F}_i \cdot \sum \mathscr{E}_i \right) \\
&= \frac{1}{2} \mathscr{E} \cdot \sum \mathscr{F} + \frac{1}{2} \cdot \mathscr{F} \cdot \sum \mathscr{E}.
\end{aligned}$$

- Other cases are trivial.

This concludes the proof. □

Lemmas 5.12 and 5.15 together allow to prove that the $\mathcal{S}_{\mathsf{IV}}(\cdot)$ commutes well with the various constructs of $\Lambda_{\oplus}$.

Lemma 5.16. *For each term $M \in \Lambda_{\oplus}$, then*

1. *If M is a value V, then $\mathcal{S}_{\mathsf{IV}}(M) = \{V^1\}$;*
2. *If M is an application NL, then*

$$\mathcal{S}_{\mathsf{IV}}(M) = \sum_{\lambda x.P \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(N)), V \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(L))} \mathcal{S}_{\mathsf{IV}}(N)(\lambda x.P) \cdot \mathcal{S}_{\mathsf{IV}}(L)(V) \cdot \mathcal{S}_{\mathsf{IV}}(P\{V/x\});$$

3. *If M is a sum $N \oplus L$, then $\mathcal{S}_{\mathsf{IV}}(M) = \frac{1}{2} \cdot \mathcal{S}_{\mathsf{IV}}(N) \cdot \sum \mathcal{S}_{\mathsf{IV}}(L) + \frac{1}{2} \cdot \mathcal{S}_{\mathsf{IV}}(L) \cdot \sum \mathcal{S}_{\mathsf{IV}}(N)$.*

Proof. We will use the following fact throughout the proof:

Fact 5.17. If $M \mapsto_{\mathsf{v}} \overline{N}$, then $\sup_{M \Rightarrow_{\mathsf{IV}} \mathscr{D}} \mathscr{D} = \sup_{N_i \Rightarrow_{\mathsf{IV}} \mathscr{D}_i} (\sum_{i=1}^{n} \frac{1}{n} \mathscr{D}_i)$.

The inequalities above can be proved separately:

- If M is a value, the thesis follows by small step semantics rules. Indeed, $M \Rightarrow_{\mathsf{IV}} \emptyset$.
- For the other cases, the ($\leq$) direction follows from Lemma 5.12 and Fact 5.17; $\geq$ direction follows from Lemma 5.15 and Fact 5.17.

This concludes the proof. □

The fact $\mathcal{S}_{\mathsf{IV}}(M)$ can be assigned to M in the big-step semantics is an easy consequence of Lemma 5.16:

Proposition 5.18. *$M \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(M)$.*

Proof. We will prove the thesis by coinduction: we can prove that all judgments $M \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(M)$ belong to the coinductive interpretation of the underlying formal system Φ (in this case, the formal system is $\Phi = \{\mathsf{bv_v}, \mathsf{ba_v}, \mathsf{bs_v}\}$). To do that, we need to prove that the set A of all those judgments is consistent, *i.e.*, that $A \subseteq F_\Phi(A)$. This amounts to show that if $c \in A$, then there is a derivation for c whose immediate premises are themselves in A. Let's distinguish some cases:

- If $M = V$ then $V \Downarrow_{\mathsf{CV}} \{V^1\}$ by $\mathsf{bv_v}$ rule, and $\{V^1\} = \mathcal{S}_{\mathsf{IV}}(V)$ because of Lemma 5.16.
- If M is an application NL, take the judgment $c = N \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(N)$, $d = L \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(L)$ and the family of judgments

$$\{P\{V/x\} \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(P\{V/x\})\}_{\lambda x.P \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(N)), V \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(L))};$$

  we will prove that the judgment $NL \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(NL)$ can be derived in a single step from c, d and those in the family above by means of $\mathsf{ba_v}$ rule. Simply observe that:

$$\frac{\begin{array}{c} N \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(N) \qquad L \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(L) \\ \{\{P\{V/x\}\} \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(\{P\{V/x\}\})\}_{\lambda x.P \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(N)), V \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(L))} \end{array}}{NL \Downarrow_{\mathsf{CV}} \displaystyle\sum_{\lambda x.P \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(N)), V \in \mathsf{S}(\mathcal{S}_{\mathsf{IV}}(L))} \mathcal{S}_{\mathsf{IV}}(N)(\lambda x.P) \cdot \mathcal{S}_{\mathsf{IV}}(L)(V) \cdot \mathcal{S}_{\mathsf{IV}}(P\{V/x\})} \ \mathsf{ba_v}.$$

  The thesis follows applying Lemma 5.16.
- If M is a sum $N \oplus L$, take the judgment $c = N \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(N)$ and $d = M \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(L)$: we will prove that the judgment $N \oplus L \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(NL)$ can be inferred in a single step from c and d by means of $\mathsf{bs_v}$ rules. Clearly, c and d belong to A. Moreover:

$$\frac{N \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(N) \quad L \Downarrow_{\mathsf{CV}} \mathcal{S}_{\mathsf{IV}}(L)}{N \oplus L \Downarrow_{\mathsf{CV}} \frac{1}{2} \cdot \mathcal{S}_{\mathsf{IV}}(N) \cdot \sum \mathcal{S}_{\mathsf{IV}}(L) + \frac{1}{2} \cdot \mathcal{S}_{\mathsf{IV}}(L) \cdot \sum \mathcal{S}_{\mathsf{IV}}(N)} \ \mathsf{bs_v}$$

  and by Lemma 5.16, case 3 we obtain the thesis.

This concludes the proof. □

The equality between big-step and small-step semantics is a corollary of Propositions 5.14 and 5.18.

Theorem 5.19. *$\mathcal{B}_{\mathsf{CV}}(M) = \mathcal{S}_{\mathsf{IV}}(M)$.*

6. Call-by-name

In Section 5 we endowed $\Lambda_\oplus$ with a *call-by-value* probabilistic operational semantics and showed that the distribution assigned to any term M is the same in big-step and in small-step semantics, independently on whether they are defined inductively or coinductively. Actually, the same holds in call-by-name: both big-step and small-step semantics can be interpreted (co)inductively and proved equivalent, following the same path used in call-by-value. In this section, we briefly sketch how this can be done, by defining $\mathcal{B}_{\mathsf{CN}}(M)$ and $\mathcal{S}_{\mathsf{IN}}(M)$ and by proving they are equivalent. We do not give technical lemmas or proofs, which retrace exactly their call-by-value counterparts. The reader can find a more detailed explanation (with proof and intermediate results) in [3].

Definition 6.1 (call-by-name reduction)**.** Leftmost reduction $\mapsto_{\mathsf{n}}$ is the least binary relation between $\Lambda_\oplus$ and $\Lambda_\oplus^*$ such that:

$$(\lambda x.M)N \mapsto_{\mathsf{n}} M\{x/N\}; \qquad MN \mapsto_{\mathsf{n}} \overline{L}N \quad \text{if } M \mapsto_{\mathsf{n}} \overline{L};$$
$$M \oplus N \mapsto_{\mathsf{n}} M, N.$$

Note that, contrarily to call-by-value, in call-by-name it *is* possible to perform a choice between terms which are not values.

6.1. Small-step semantics

As for call-by-name, we model separately terminating and non-terminating computations. The rule schema is the same, up to the different reduction relation $\mapsto_{\mathsf{n}}$. First of all, there is an inductively defined binary relation $\Rightarrow_{\mathsf{IN}}$ between $\Lambda_\oplus$ and distributions. Rules are as follows:

$$\frac{}{M \Rightarrow_{\mathsf{IN}} \emptyset}\,\mathsf{se_n} \qquad \frac{}{V \Rightarrow_{\mathsf{IN}} \{V^1\}}\,\mathsf{sv_n} \qquad \frac{M \mapsto_{\mathsf{n}} \overline{N} \quad N_i \Rightarrow_{\mathsf{IN}} \mathscr{D}_i}{M \Rightarrow_{\mathsf{IN}} \displaystyle\sum_{i=1}^{n} \frac{1}{n}\mathscr{D}_i}\,\mathsf{sm_n}$$

$\mathcal{S}_{\mathsf{IN}}(M)$ is the distribution $\sup_{M \Rightarrow_{\mathsf{IN}} \mathscr{D}} \mathscr{D}$. Moreover, there is also another, coinductively interpreted binary relation between $\Lambda_\oplus$ and $\mathbb{R}_{[0,1]}$ capturing divergence. Rules are as follows:

$$\frac{}{V \Rightarrow_0^\infty}\,\mathsf{dv_n} \qquad \frac{M \mapsto_{\mathsf{n}} \overline{N} \quad \{N_i \Rightarrow_{p_i}^\infty\}_{i\in[1,n]}}{M \Rightarrow_{\sum_{i=1}^{n} \frac{1}{n}p_i}^\infty}\,\mathsf{dm_n}$$

$\mathcal{D}_{\mathsf{CN}}(M)$ is nothing more than $\sup_{M \Rightarrow_p^\infty} p$. Notice how the differences between call-by-value and call-by-name small-step semantics all come from the reduction relation, since the rules above are analogous to their call-by-value siblings.

As done in Section 5.1.3 for call-by-value, a conductive version of call-by-name small step semantics can be easily defined.

6.2. Big-step semantics

We define call-by-name big-step semantics of terms in $\varLambda_\oplus$ as the co-inductive interpretation of a suitable set of rules. Again, this allow us to capture infinite computations. A coinductively interpreted binary relation $\Downarrow_{\mathsf{CN}}$ between $\varLambda_\oplus$ and distributions is obtained by taking all instances of the following rules:

$$\frac{}{V \Downarrow_{\mathsf{CN}} \{V\}}\ \mathsf{bv_n} \qquad \frac{M \Downarrow_{\mathsf{CN}} \mathscr{D} \quad \{P\{N/x\} \Downarrow_{\mathsf{CN}} \mathscr{E}_{P,N}\}_{\lambda x.P \in \mathsf{S}(\mathscr{D})}}{MN \Downarrow_{\mathsf{CN}} \displaystyle\sum_{\lambda x.P \in \mathsf{S}(\mathscr{D})} \mathscr{D}(\lambda x.P) \cdot \mathscr{E}_{P,N}}\ \mathsf{ba_n}$$

$$\frac{M \Downarrow_{\mathsf{CN}} \mathscr{D} \quad N \Downarrow_{\mathsf{CN}} \mathscr{E}}{M \oplus N \Downarrow_{\mathsf{CN}} \dfrac{1}{2} \cdot \mathscr{D} + \dfrac{1}{2} \cdot \mathscr{E}}\ \mathsf{bs_n}$$

$\mathcal{B}_{\mathsf{CN}}(M)$ is simply the subdistribution $\inf_{M \Downarrow_{\mathsf{CN}} \mathscr{D}} \mathscr{D}$. The way binary choices are managed reflects the reduction rules, which allow to evaluate a binary choice to one of its components even if the latter are not values. Indeed, while in call-by-value normalization factors $\sum \mathscr{D}$ and $\sum \mathscr{E}$ were necessary, they are not here anymore.

Example 6.2. Consider the term $M = \varOmega \oplus (\lambda x.x)$. Recall from Example 5.8, that $\varOmega \Downarrow_{\mathsf{CV}} \mathscr{D}$ for every $\mathscr{D}$. Moreover, $\lambda x.x \Downarrow_{\mathsf{CV}} \{(\lambda x.x)^1\}$. This implies $M \Downarrow_{\mathsf{CV}} \emptyset$, and, as a consequence, that $\mathcal{B}_{\mathsf{CV}}(M) = \emptyset$. The same behavior cannot be mimicked in call-by-name. Indeed, while $\varOmega \Downarrow_{\mathsf{CN}} \mathscr{D}$ (for every $\mathscr{D}$) and $\lambda x.x \Downarrow_{\mathsf{CN}} \{(\lambda x.x)^1\}$, the smallest distribution which can be assigned to M is $\mathcal{B}_{\mathsf{CN}}(M) = \{(\lambda x.x)^{\frac{1}{2}}\}$.

Inductive call-by-name big step semantics can be obtained adding the rule which assigns the empty distribution to any term and taking the inductive interpretation of the system.

6.3. Comparing the different notions

We mention here some results in the same style as those presented in Sections 5.3 and 5.4 for call-by-value semantics. For more details, please refer to [3].

On the one hand, the probability of convergence derived through the specific relation $M \Rightarrow_p^\infty$ equals the one induced from $\mathcal{S}_{\mathsf{IN}}(M)$:

Theorem 6.3. *For every M, $\mathcal{D}_{\mathsf{CN}}(M) + \sum \mathcal{S}_{\mathsf{IN}}(M) = 1$.*

On the other, the four different ways to give semantics to terms are all equivalent. For example:

Theorem 6.4. *$\mathcal{B}_{\mathsf{CN}}(M) = \mathcal{S}_{\mathsf{IN}}(M)$.*

The strategies one follows when proving the results above are analogous to the ones we adopted in the call-by-value case.

7. CPS translations and simulations

In this section we show that in $\Lambda_\oplus$ it is possible to simulate call-by-value by call-by-name and *vice versa*. It is well known that in the *weak* untyped λ-calculus, call-by-value and call-by-name are not equivalent notions of reduction. Moreover, the presence of binary choice exacerbates the confluence problem, as shown in Section 2. It is clear that duplications play a central role: in particular, the order in which probabilistic choices and duplications are performed matters.

The binary choices opens a related question about the possibility of including some "administrative rules" distributing sums over the other constructs. The following example shows that it may be difficult to do so:

Example 7.1. Let us consider the following terms: $M \equiv \lambda x.x(M_\top \oplus M_\perp)$ and $N \equiv (\lambda x.xM_\top) \oplus (\lambda x.xM_\perp)$. Note that M can be obtained from N by means of administrative rules like $L(P \oplus Q) = (LP) \oplus (LQ)$. Consider the term $N_{xor} \equiv (\lambda z.M_{xor}zz)$. We have $\mathcal{S}_{\mathsf{IV}}(MN_{xor}) = \{M_\perp^1\} = \mathscr{D}$ and $\mathcal{S}_{\mathsf{IN}}(MN_{xor}) = \{M_\perp^{\frac{1}{2}}, M_\top^{\frac{1}{2}}\} = \mathscr{E}$, whereas $\mathcal{S}_{\mathsf{IV}}(NN_{xor}) = \mathcal{S}_{\mathsf{IN}}(NN_{xor}) = \mathscr{D}$ and therefore M and N have different call-by-name observational behaviors, thus M and N can not be considered equivalent.

A study about the observational behavior of terms is a fascinating subject, but is out of the scope of this paper. By the way, this is extensively investigated in the non deterministic setting [7], in which an algebraic semantics of terms is defined by way of a generalization of Böhm's trees. A further generalization to the probabilistic setting is left for future work.

What we are interested in here is to develop an operational study of $\Lambda_\oplus$. And an interesting question is clearly whether call-by-value and call-by-name, although being distinct notions of reduction, can be somehow made equivalent through a suitable CPS translation, even in presence of binary choices (with a probabilistic semantics). In this section, a simulation between call-by-value and call-by-name in $\Lambda_\oplus$ is presented and proved correct. We begin with the simulation of call-by-value by call-by-name (Sect. 7.1), then we proceed by hinting at the simulation of call-by-name by call-by-value (Sect. 7.2).

It is mandatory to say that our simulation follows significantly Plotkin's CPS translation [22], extended here to accommodate probabilistic choice. The structure of the proof is the same. A linguistic translation between the original languages and a suitable extension of itself is defined. Given a call-by-value (respectively call-by-name) computation in the original languages, each step of the computation is simulated in the extended languages by means of a sequence of "structural" reduction, followed by a proper call-by-name (respectively call-by-value) reduction. However, in order to retrace the result in $\Lambda_\oplus$, some technical adjustments are required. Even if the involved techniques are quite standard, the non deterministic reduction relations and the probabilistic nature of the operational semantics make the proofs non trivial.

7.1. Simulating call-by-value with call-by-name

Suppose to extend $\Lambda_\oplus$ with a denumerable set of continuation variables $\mathsf{C} = \{\alpha, \beta, \epsilon \ldots\}$, disjoint from the original set X of variables of the language. We will call $\Lambda_\oplus^+$ the language of λ-terms extended this way. All definitions and constructions on $\Lambda_\oplus$ (including its operational semantics) extend smoothly to $\Lambda_\oplus^+$.

Call-by-value reduction on $\Lambda_\oplus$ can be simulated by call-by-name reduction of $\Lambda_\oplus^+$ by translating every term M in $\Lambda_\oplus$ to a term in $\Lambda_\oplus^+$, which will be proved to be equivalent to M in a certain sense.

Definition 7.2 (call-by-value translation)**.** The translation map $\llbracket \cdot \rrbracket$ from $\Lambda_\oplus$ to $\Lambda_\oplus^+$ is recursively defined as follows:

$$\begin{aligned}
\llbracket x \rrbracket &= \lambda\epsilon.\epsilon x;\\
\llbracket MN \rrbracket &= \lambda\epsilon.(\llbracket M \rrbracket(\lambda\alpha.\llbracket N \rrbracket(\lambda\beta.\alpha\beta\epsilon)));\\
\llbracket \lambda x.M \rrbracket &= \lambda\epsilon.\epsilon\lambda x.\llbracket M \rrbracket;\\
\llbracket M \oplus N \rrbracket &= \lambda\epsilon.\llbracket M \rrbracket(\lambda\alpha.\llbracket N \rrbracket(\lambda\beta.((\lambda\gamma.\gamma\alpha) \oplus (\lambda\gamma.\gamma\beta))\epsilon)).
\end{aligned}$$

As expected, call-by-value is simulated by way of so-called continuations. As a consequence, (call-by-value) reduction on M is *not* simulated simply reducing (in call-by-name) $\llbracket M \rrbracket$, but by feeding it with the identity continuation $\lambda x.x$. Furthermore, we do not obtain this way the same value(s) as the one(s) we would obtain by evaluating M, but something related to that by a function Ψ, which sends values in $\Lambda_\oplus$ into values in $\Lambda_\oplus^+$ as follows:

$$\Psi(x) = x; \qquad\qquad \Psi(\lambda x.M) = \lambda x.\llbracket M \rrbracket.$$

Clearly, Ψ can be naturally extended to a map on distributions (of values).

The rest of this section is devoted to showing the following theorem, which retraces in $\Lambda_\oplus$ the classic result from [22]:

Theorem 7.3 (simulation)**.** *For every M, $\Psi(\mathcal{S}_{\mathsf{IV}}(M)) = \mathcal{S}_{\mathsf{IN}}(\llbracket M \rrbracket(\lambda x.x))$.*

In the following, we will omit most proofs, which can be found in [3]. We define now the suitable extension of the infix operator ":" introduced by Plotkin in [22]:

Definition 7.4 (infix operator ":" for $\llbracket\cdot\rrbracket$)**.** The infix operator ":" for the map $\llbracket\cdot\rrbracket$ is defined as follows:

$$\begin{aligned}
V : K &= K\Psi(V);\\
LP : K &= L : (\lambda\alpha.\llbracket P \rrbracket(\lambda\beta.\alpha\beta K)) \quad \text{if} \quad L \notin \mathsf{Val};\\
VL : K &= L : ((\lambda\beta.\Psi(V)\beta K)) \quad \text{if} \quad L \notin \mathsf{Val};\\
VW : K &= \Psi(V)\Psi(W)K;\\
L \oplus P : K &= L : (\lambda\alpha.\llbracket P \rrbracket(\lambda\beta.(((\lambda\gamma.\gamma\alpha) \oplus (\lambda\gamma.\gamma\beta))K))) \quad \text{if} \quad L \notin \mathsf{Val};\\
V \oplus L : K &= L : (\lambda\beta.(((\lambda\gamma.\gamma\Psi(V)) \oplus (\lambda\gamma.\gamma\beta))K)) \quad \text{if} \quad L \notin \mathsf{Val};\\
V \oplus W : K &= ((\lambda\gamma.\gamma\Psi(V)) \oplus (\lambda\gamma.\gamma\Psi(W)))K.
\end{aligned}$$

The operator ":" can be naturally generalized to an operator on distributions. In this case we will use the notation $\mathscr{D} : K$. The following lemmas give a meaning to the operator ":", while providing intermediate results for proving Theorem 7.3. Lemma 7.5 shows that $M : K$ is the result of the computation from $[\![M]\!]K$ involving all structural reduction:

Lemma 7.5. *For all $M \in \varLambda_{\oplus}$, $[\![M]\!]K \mapsto_{\mathsf{n}}^{*} M : K$.*

Proof. By induction on M and on the definition of ":". □

As a technical tool, we here need a generalization of the small step semantic relation $\cdot \Rightarrow_{\mathsf{IN}} \cdot$, that we denote as $\cdot \Rrightarrow \cdot$. The new relation is defined inductively as $\cdot \Rightarrow_{\mathsf{IN}} \cdot$, but the first two rules are replaced by the two rules

$$\frac{}{M \Rrightarrow \{M^1\}} \qquad \frac{}{M \Rrightarrow \emptyset.}$$

This means, in particular, that the relation above maps terms to distribution over terms, rather than distributions over values. Of course, relations $\cdot \Rightarrow_{\mathsf{IN}} \cdot$ and $\cdot \Rrightarrow \cdot$ are strongly related, as shown in the following lemma:

Lemma 7.6. *If $M \Rrightarrow \mathscr{D}$, where $\mathsf{S}(\mathscr{D}) \subseteq \mathsf{Val}$, then $M \Rightarrow_{\mathsf{IN}} \mathscr{D}$. Conversely, if $M \Rightarrow_{\mathsf{IN}} \mathscr{D}$ then $M \Rrightarrow \mathscr{D}$.*

Proof. Simple inductions. □

Lemma 7.7 extends Plotkin's lemma which states that whenever a term M reduces in call-by-value to a term N, there exists a *call-by-name* computation from $M : K$ to $N : K$. In $\varLambda_{\oplus}$ we have to deal with the non deterministic nature of our relations $\mapsto_{\mathsf{v}}$ and $\mapsto_{\mathsf{n}}$ (which map terms into a set of terms), and with the fact that the evaluation of a term returns a probability distribution on values and not a single value.

Lemma 7.7. *If $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$ and K is a closed value, then $M : K \Rrightarrow \mathscr{D} : K$.*

Proof. By induction on the complexity of a derivation for $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$. □

Lemma 7.8 shows that the call-by-value evaluation of a term M dominates, in term of distributions, the call-by-name evaluation of $M : \lambda x.x$, up to $\varPsi$:

Lemma 7.8. *If $M : \lambda x.x \Rightarrow_{\mathsf{IN}} \mathscr{D}$, then $M \Rightarrow_{\mathsf{IV}} \mathscr{E}$, where $\varPsi(\mathscr{E}) \geq \mathscr{D}$.*

Proof. The proof goes by induction on the complexity of a derivation π of $M : \lambda x.x \Rightarrow_{\mathsf{IN}} \mathscr{D}$. □

Lemmas 7.5–7.8 are the basic ingredients underlying the proof of Theorem 7.3.

Proof of Theorem 7.3. To prove that $\Psi(\mathcal{S}_{\mathsf{IV}}(M)) = \mathcal{S}_{\mathsf{IN}}(\llbracket M \rrbracket \lambda x.x)$, we proceed by showing the following:

1. If $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, then $\llbracket M \rrbracket \lambda x.x \Rightarrow_{\mathsf{IN}} \mathscr{E}$ and $\mathscr{E} \geq \Psi(\mathscr{D})$;
2. If $\llbracket M \rrbracket \lambda x.x \Rightarrow_{\mathsf{IN}} \mathscr{D}$, then $M \Rightarrow_{\mathsf{IV}} \mathscr{E}$ where $\Psi(\mathscr{E}) \geq \mathscr{D}$.

To prove point 1, we observe that if $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, then both $\llbracket M \rrbracket K \mapsto_{\mathsf{n}}^{*} M : K$ (by Lem. 7.5) and $M : K \Rightarrow_{\mathsf{IV}} \mathscr{D} : K$ (by Lems. 7.7 and 7.6). Now, notice that, for every value V, $V : (\lambda x.x) = (\lambda x.x)\Psi(V) \mapsto_{\mathsf{v}} \Psi(V)$. As a consequence, it's clear that $\llbracket M \rrbracket : \lambda x.x \Rightarrow_{\mathsf{IN}} \mathscr{D}$. Point 2 is nothing more than Lemma 7.8. This concludes the proof. □

7.2. Simulating call-by-name by call-by-value

In Section 7.1 we proved the "imperfect" simulation of call-by-value by call-by name strategy. The inverse result can be proved very similarly. We omit all technical details: the proof follows as an expected modification of the one in Section 7.1. See again [3] for all the details.

Let us define the extended language $\Lambda_{\oplus}^{+}$ as in Section 7.1.

Definition 7.9 (call-by-value translation)**.** The translation map $\llfloor \cdot \rrfloor$ from $\Lambda_{\oplus}$ to $\Lambda_{\oplus}^{+}$ is recursively defined as follows:

$$\begin{aligned}
\llfloor x \rrfloor &= x;\\
\llfloor \lambda x.M \rrfloor &= \lambda \epsilon.\epsilon \lambda x.\llfloor M \rrfloor;\\
\llfloor MN \rrfloor &= \lambda \epsilon.\llfloor M \rrfloor(\lambda \alpha.\alpha \llfloor N \rrfloor \epsilon);\\
\llfloor M \oplus N \rrfloor &= \lambda \epsilon.(((\lambda \alpha.\llfloor M \rrfloor \alpha) \oplus (\lambda \alpha.\llfloor N \rrfloor \alpha))\epsilon).
\end{aligned}$$

We define a function Φ, which sends values into terms as

$$\begin{aligned}
\Phi(x) &= x(\lambda y.y);\\
\Phi(\lambda x.M) &= \lambda x.\llfloor M \rrfloor.
\end{aligned}$$

Observe that Φ sends closed values to closed values. As such, it can be naturally extended to distributions of closed values. The dual of Theorem 7.3 follows:

Theorem 7.10 (simulation)**.** *For every M, $\Phi(\mathcal{S}_{\mathsf{IN}}(M)) = \mathcal{S}_{\mathsf{IV}}(\llfloor M \rrfloor(\lambda x.x))$.*

8. On the expressive power of $\Lambda_{\oplus}$

The λ-calculus $\Lambda_{\oplus}$ is endowed with a very restricted form of (probabilistic) choice, which is binary and such that both possible outcomes have probability 1/2. It is thus natural to ask oneself whether this is an essential restriction or not. In this section, we show that this is *not* essential, by proving that the set of *representable* distributions on the natural numbers, namely those which can be denoted by a term of $\Lambda_{\oplus}$ equals the set of *computable* distributions, defined in terms of Turing machines. In this section, we assume to work with call-by-value reduction, even if everything could be rephrased in call-by-name, in view of the results in Section 7.1.

We first of all need to define what a computable distribution (on the natural numbers) is. This is based on a notion of *approximation* for the function assigning probabilities to natural numbers:

Definition 8.1 (approximating function)**.** Given any function $f : Q \to \mathbb{R}_{[0,1]}$, the *approximating function* for f, denoted $Appr_f$ is the function from $Q \times \mathbb{N}$ to $\mathbb{Q}$ which on input (a, n), returns a rational number $Appr_f(a, n)$ such that $|Appr_f(a, n) - f(a)| < 1/n$.

A distribution on the natural numbers is computable if its approximating function is itself computable in the usual sense:

Definition 8.2 (computable distributions)**.** A distribution $\mathcal{D} : \mathbb{N} \to \mathbb{R}_{[0,1]}$ is computable iff the function $Appr_{\mathcal{D}} : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$ is computable in the sense of Turing machines.

The next step consists in understanding which class of distributions on the natural numbers can be captured by $\Lambda_\oplus$, *i.e.* is the semantics of a λ-term. In $\Lambda_\oplus$ we are able to represent (up to approximation) probability functions from a suitable domain to the real interval $\mathbb{R}_{[0,1]}$. For this reason, we assume to work with a fixed encoding of the natural numbers into λ-terms, namely the one usually attributed to Scott [29]. More specifically, natural numbers and binary strings can be encoded as follows:

$$\ulcorner 0 \urcorner = \lambda xy.x;$$
$$\ulcorner n+1 \urcorner = \lambda xy.y\ulcorner n \urcorner.$$

Moreover, we encode pairs of values as follows: $\langle V, W \rangle = \lambda x.xVW$. In the rest of the section we will use the following notation:

Notation 8.3. Given a proper distribution $\mathscr{D} : \mathsf{Val} \to \mathbb{R}_{[0,1]}$ that assigns nonzero probability only to representation of natural numbers, the corresponding probability distribution over the natural numbers will be denoted as $\{\mathscr{D}\}$.

Distributions on the natural numbers which can be captured, via approximation, by $\Lambda_\oplus$ are said to be representable:

Definition 8.4 (representable distribution)**.** A probability distribution $\mathcal{D}$ over the natural numbers is said to be *representable* iff there is a λ-term $M_{\mathcal{D}}$ such that $\{\mathcal{S}_{\mathsf{IV}}(M_{\mathcal{D}})\} = \mathcal{D}$.

Our main goal will be to prove that the class of representable distributions coincides with the class of computable distributions: on one hand each distribution obtained by small step evaluation of a λ-term in $\Lambda_\oplus$ is proved to be computable (Thm. 8.5); on the other hand, each computable distribution can be represented by a term in $\Lambda_\oplus$ (Thm. 8.10).

Whatever can be denoted by a λ-term is actually a computable distribution, *i.e.* one which can be approximated up to any degree of precision by a Turing machine:

Theorem 8.5 (soundness)**.** *Every representable distribution is computable.*

Proof. Suppose $\mathcal{D}$ is a representable probability distribution. Then there is a term M such that $\{\mathcal{S}_{\mathsf{IV}}(M)\} = \mathcal{D}$. This implies, in particular, that $\sum \mathcal{S}_{\mathsf{IV}}(M) = 1$, because $\sum \mathcal{D} = 1$ itself. An algorithm computing $Appr_{\mathcal{D}}$ can be easily designed as an evaluator for M: on input (a, n), simply compute distributions $\mathscr{D}$ such that $M \Rightarrow_{\mathsf{IV}} \mathscr{D}$, until you find one, call it $\mathscr{E}$, such that $\sum \mathscr{E} > 1 - 1/n$. Such a distribution $\mathscr{E}$ can *always* be reached, provided the evaluation of M is performed in a breadth-first order rather than in depth-first order (where the underlying tree, called the *evaluation tree* is the one induced by probabilistic choice). Concretely, one can proceed as follows: for increasing values of m, compute the 2^m terms $N_1, \ldots, N_{2^m}$ which are the leaves of the evaluation tree capturing all reduction sequences of length m starting from M (where we assume that every reduction step produces a branching and that values themselves can be reduced to themselves). If enough of $N_1, \ldots, N_{2^m}$ are values, we are done, then we increase m and iterate. □

The completeness theorem requires other definitions and some auxiliary results. We define the class of the so-called *finite distribution terms*:

Definition 8.6 (finite distribution terms)**.** Finite distribution terms are λ-terms generated inductively as follows:

- For every n, $\lambda xy.x\ulcorner n \urcorner$ is a finite distribution term;
- If M and N are finite distribution terms, then $\lambda xy.yMN$ is a finite distribution term.

Intuitively, any finite distribution term denotes a probability distribution on the natural numbers. Formally, given a finite distribution term M, the underlying probability distribution on the natural numbers $\{M\}$ is defined as follows:

$$\{\lambda xy.x\ulcorner n \urcorner\} = \{n^1\};$$
$$\{\lambda xy.yMN\} = \frac{1}{2}\{M\} + \frac{1}{2}\{N\}.$$

Lemma 8.7 is an intermediate result towards Lemma 8.8.

Lemma 8.7 (Fixed-point combinator)**.** *There is a term H such that for every value V, HV rewrites deterministically to $V(\lambda x.HVx)$.*

Proof. The term H is simply WW where

$$W = \lambda x.\lambda y.y(\lambda z.xxyz).$$

Indeed:

$$\begin{aligned}HV &= WWV\\ &\mapsto_{\mathsf{v}} (\lambda y.y(\lambda z.WWyz))V\\ &\mapsto_{\mathsf{v}} V(\lambda z.WWVz)\\ &= V(\lambda z.(HV)z).\end{aligned}$$

This concluses the proof. □

Finite distribution terms are *values* which do not represent natural numbers by themselves, but rather *distributions* on the natural numbers. As a matter of fact, any finite distribution term can be "executed", obtaining the underlying distribution, and this can be done by a λ-term:

Lemma 8.8. *There is a λ-term M_{fdt} such that for every finite distribution term N, $\{\mathcal{S}_{\mathsf{lV}}(M_{fdt}N)\} = \{N\}$.*

Proof. The term M_{fdt} is simply HV where V is

$$\lambda xy.y(\lambda z.z)(\lambda zw.(xz) \oplus (xw))$$

Indeed, by induction (X is $\lambda x.HVx$):

$$\begin{aligned}M_{fdt}(\lambda xy.x\ulcorner n\urcorner) &\mapsto^*_{\mathsf{v}} (\lambda xy.x\ulcorner n\urcorner)(\lambda z.z)(\lambda zw.(Xz) \oplus (Xw))\\ &\mapsto^*_{\mathsf{v}} (\lambda z.z)\ulcorner n\urcorner \mapsto_{\mathsf{v}} \ulcorner n\urcorner;\\ M_{fdt}(\lambda xy.yMN) &\mapsto^*_{\mathsf{v}} VX(\lambda xy.yMN)\\ &\mapsto^*_{\mathsf{v}} (\lambda zw.Xz \oplus Xw)MN\\ &\mapsto^*_{\mathsf{v}} (XM) \oplus (XN);\end{aligned}$$

and $(XM) \oplus (XN)$, applying the i.h. on the subterms, evaluates with equiprobable distribution to either $\{M\}$, $\{N\}$. This concludes the proof. □

The following proposition guarantees the existence of a sort of successive approximations function, which computes the distribution of a given term in a finite number of steps:

Proposition 8.9 (splitting)**.** *There is a λ-term M_{split} such that for every term N computing the distribution $\mathcal{D}$ (in the sense of Def. 8.2), $M_{split}N$ rewrites deterministically to $\langle L, P\rangle$ where*

- *L is a finite distribution term such that $\{L\} = \mathcal{E}$;*
- *P computes a distribution $\mathcal{F}$ (again in the sense of Def. 8.2);*
- *$\mathcal{D} = \frac{1}{2}\mathcal{E} + \frac{1}{2}\mathcal{F}$.*

Proof. First, observe that N computes a proper distribution, and as a consequence there are a natural number n and a sequence of pairs of natural numbers

$$(m_1, p_1), \dots, (m_n, p_n)$$

such that $m_1, \ldots, m_n$ are distinct and moreover

$$\sum_{i=1}^{n}\left(f(m_i,p_i)-\frac{1}{p_i}\right)>\frac{1}{2}$$

where f is the function computed by N. The sequence below is guaranteed to be reached by successively querying N in a dovetail order. This way, the (sub)distribution $\mathcal{E}$ can always be determined, together with a finite distribution term L for it. P is nothing more that N where however we pay attention to “subtract” the (finitely many) values given by $\mathcal{E}$. □

By means of Lemma 8.8 and Proposition 8.9 it is possible to prove the following completeness result:

Theorem 8.10 (completeness)**.** *Every computable probability distribution $\mathcal{D}$ over the natural numbers is representable by a term $M_{\mathcal{D}}$.*

Proof. Define a term M simply as HV where V is

$$\lambda xy.(M_{split}y)(\lambda zw.[\lambda s.(M_{fdt}z)\oplus\lambda s.(xw)](\lambda s.s))$$

We will prove the following statement: for any $n\in\mathbb{N}$ e for all $N\in\Lambda_{\oplus}$ which computes a distribution $\mathscr{E}$, there exists a distribution $\mathscr{D}$ such that $MN\Rightarrow_{\mathsf{IV}}\mathscr{D}$ with $\sum\mathscr{D}\geq(1-\frac{1}{2^n})$ and $\{\mathscr{D}\}\leq\mathscr{E}$. We will prove the thesis by induction on the natural number n.

- If $n=0$ then $\mathscr{D}=\emptyset$ and $MN\Rightarrow_{\mathsf{IV}}\emptyset$ trivially.
- If $n>0$ observe that MN rewrites as follows:

$$\begin{aligned}MN=HVN&\mapsto^{*}V(\lambda t.(HV)t)N\\&\mapsto^{*}(M_{split}N)\underbrace{(\lambda zw.((((\lambda s.M_{fdt}z)\oplus(\lambda s.(\lambda t.HVt)w))(\lambda s.s))))}_{R.}.\end{aligned}$$

Observe that, by Proposition 8.9, $M_{split}N$ rewrites deterministically to the pair $\langle L,P\rangle$, where L is a finite distribution term such that $\{L\}=\mathscr{E}_L$ and P is a finite distribution term such that $\{P\}=\mathscr{E}_P$ and $\mathscr{E}=\frac{1}{2}\mathscr{E}_L+\frac{1}{2}\mathscr{E}_P$. Then:

$$\begin{aligned}(\lambda x.xLP)R&\mapsto(\lambda zw.((((\lambda s.M_{fdt}z)\oplus(\lambda s.(\lambda t.HVt)w))(\lambda s.s))))LP\\&\mapsto^{2}((\lambda s.M_{fdt}L)\oplus(\lambda s.(\lambda t.HVt)P))(\lambda s.s)\\&\mapsto(\lambda s.M_{fdt}L)(\lambda s.s),(\lambda s.(\lambda t.HVt)P)(\lambda s.s).\end{aligned}$$

Now observe that, $(\lambda s.M_{fdt}L)(\lambda s.s)\mapsto M_{fdt}L$ and by Lemma 8.8 this term evaluates to $\{L\}=\mathscr{E}_L$. Moreover, $(\lambda s.(\lambda t.HVt)P)(\lambda s.s)\mapsto^{2}(HV)P=MP$ and we can apply to P the induction hypothesis, obtaining that $MP\Rightarrow_{\mathsf{IV}}\mathscr{D}_P$

with $\sum \mathscr{D}_P \geq (1 - \frac{1}{2^n})$ and $\{\mathscr{D}_P\} \leq \mathscr{E}_P$. Let us take the distribution $\mathscr{D}$ as $\mathscr{D} = \frac{1}{2}\mathscr{E}_L + \frac{1}{2}\mathscr{D}_P$. Clearly, $MN \Rightarrow_{\mathsf{IV}} \mathscr{D}$. Moreover:

$$\begin{aligned}\sum \mathscr{D} &= \frac{1}{2}\sum \mathscr{E}_L + \frac{1}{2}\sum \mathscr{D}_P \\ &\overset{i.h.}{\geq} \frac{1}{2} + \frac{1}{2}\left(1 - \frac{1}{2^n}\right) \\ &= \frac{1}{2} + \frac{1}{2} - \frac{1}{2^{n+1}} = 1 - \frac{1}{2^{n+1}}.\end{aligned}$$

Finally, $\mathscr{D} \leq \mathscr{E}$. The thesis follows easily. □

9. Conclusions and future work

In this paper we studied probabilistic operational semantics for $\Lambda_\oplus$, a nondeterministic extension of untyped λ-calculus. We prove strong equivalence results between small-step and big-step semantics, both in call-by-value and in call-by-name. We also extend Plotkin's simulation to our probabilistic setting and we state and prove some results about the expressive power of $\Lambda_\oplus$.

Starting from the present paper, several directions for future work are open. On the one hand, some theoretical aspects of the calculus remain unexplored: for example, an interesting topic consists in developing an observational theory for $\Lambda_\oplus$ terms. On the other hand, it is possible to consider $\Lambda_\oplus$ as a paradigmatic language for stochastic functional programming: it would be instructive to analyze carefully the relationship between $\Lambda_\oplus$ and other probabilistic languages (for example, Park's language λ_γ [19]). Our results should be revisited in a measure theoretic setting. For example, the equivalence between inductive and coinductive semantics is reminiscent of equality between outer measure and inner measure in measure theory, and the definition of call-by-value small-step semantics is reminiscent of the notion of dominated convergence in Lebesgue's integration theory, similarly to what happens in [9, 10].

References

[1] P. Audebaud and C. Paulin-Mohring, Proofs of randomized algorithms in Coq, in *Proc. of Mathematics of Program Construction. Lect. Notes Comput. Sci.* **4014** 49–68 (2006).

[2] P.-L. Curien and H. Herbelin, The duality of computation, in *Proc. of International Conference on Functional Programming* (2000) 233–243.

[3] U. Dal Lago and M. Zorzi, Probabilistic operational semantics for the lambda calculus. Long Version. Available at `http://arxiv.org/abs/1104.0195`, 2012.

[4] O. Danvy and A. Filinski, Representing control: A study of the CPS transformation. *Math. Struct. Comput. Sci.* **2** (1992) 361–391.

[5] O. Danvy and L.R. Nielsen, CPS transformation of beta-redexes. *Inform. Process. Lett.* **94** (2005) 217–224.

[6] B.A. Davey and H.A. Priestley, *Introduction to Lattices and Order*. Cambridge University Press (2002).

[7] U. de' Liguoro and A. Piperno, Nondeterministic extensions of untyped λ-calculus. *Inform. Comput.* **122** (1995) 149–177.

[8] A. Di Pierro, C. Hankin and H. Wiklicky, Probabilistic λ-calculus and quantitative program analysis. *J. Logic Comput.* **15** (2005) 159–179.

[9] A. Edalat, Domains for computation in mathematics, physics and exact real arithmetic. *Bull. Symbolic Logic* **3** (1997) 401–452.

[10] A. Edalat and M.H. Escard, Integration in real PCF, in *Proc. of IEEE Symposium on Logic in Computer Science*. Society Press (1996) 382–393.

[11] M. Gaboardi, *Inductive and coinductive techniques in the operational analysis of functional programs: an introduction*. Master's thesis, Universita' di Milano, Bicocca (2004).

[12] M. Giry, A categorical approach to probability theory, in *Categorical Aspects of Topology and Analysis*, edited by B. Banaschewski. Springer, Berlin, Heidelberg (1982) 68–85.

[13] B. Jacobs and J. Rutten, A tutorial on (co)algebras and (co)induction. *Bull. EATCS* **62** (1996) 222–259.

[14] C. Jones, *Probabilistic non-determinism*. Ph.D. thesis, University of Edinburgh, Edinburgh, Scotland, UK (1989).

[15] C. Jones and G. Plotkin, A probabilistic powerdomain of evaluations, in *Proc. of IEEE Symposium on Logic in Computer Science*. IEEE Press (1989) 186–195.

[16] X. Leroy and H. Grall, Coinductive big-step operational semantics. *Inform. Comput.* **207** (2009) 284–304.

[17] E. Moggi, Computational lambda-calculus and monads, in *Proc. of IEEE Symposium on Logic in Computer Science*. IEEE Computer Society Press (1989) 14–23.

[18] E. Moggi, Notions of computation and monads. *Inform. Comput.* **93** (1989) 55–92.

[19] S. Park, A calculus for probabilistic languages, in *Proc. of ACM SIGPLAN International Workshop on Types in Languages Design and Implementation*. ACM Press (2003) 38–49.

[20] S. Park, F. Pfenning and S. Thrun, A monadic probabilistic language. Manuscript. Available at http://www.cs.cmu.edu/~fp/papers/prob03.pdf (2003).

[21] S. Park, F. Pfenning and S. Thrun, A probabilistic language based upon sampling functions, in *Proc. of ACM Symposium on Principles of Programming Languages* **40** (2005) 171–182.

[22] G.D. Plotkin, Call-by-name, call-by-value and the λ-calculus. *Theoret. Comput. Sci.* **1** (1975) 125–159.

[23] G.D. Plotkin, LCF considered as a programming language. *Theoret. Comput. Sci.* **5** (1977) 223–255.

[24] N. Ramsey and A. Pfeffer, Stochastic lambda calculus and monads of probability distributions, in *Proc. of ACM Symposium on Principles of Programming Languages*. ACM Press (2002) 154–165.

[25] J. Rutten, Elements of Stream Calculus (An Extensive Exercise In Coinduction). *Electron. Notes Theor. Comput. Sci* **45** (2001) 358–423.

[26] N. Saheb-Djaromi, Probabilistic LCF, in *Proc. of International Symposium on Mathematical Foundations of Computer Science. Lect. Notes Comput. Sci.* **64** (1978) 442–451.

[27] D. Sangiorgi, *Introduction to Bisimulation and Coinduction*. Cambridge University Press (2012).

[28] P. Selinger and B. Valiron, A lambda calculus for quantum computation with classical control. *Math. Struct. Comput. Sci.* **16** (2006) 527–552.

[29] C. Wadsworth, Some unusual λ-calculus numeral systems, in *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, edited by J.P. Seldin and J.R. Hindley. Academic Press (1980).

Communicated by G. Longo.
Received August 23, 2011. Accepted April 18, 2012.