

Measurable Cones and Stable, Measurable Functions

A Model for Probabilistic Higher-Order Programming

THOMAS EHRHARD, IRIF UMR 8243, Université Paris Diderot, Sorbonne Paris Cité, CNRS, France

MICHELE PAGANI, IRIF UMR 8243, Université Paris Diderot, Sorbonne Paris Cité, CNRS, France

CHRISTINE TASSON, IRIF UMR 8243, Université Paris Diderot, Sorbonne Paris Cité, CNRS, France

We define a notion of stable and measurable map between cones endowed with measurability tests and show that it forms a cpo-enriched cartesian closed category. This category gives a denotational model of an extension of PCF supporting the main primitives of probabilistic functional programming, like continuous and discrete probabilistic distributions, sampling, conditioning and full recursion. We prove the soundness and adequacy of this model with respect to a call-by-name operational semantics and give some examples of its denotations.

CCS Concepts: • **Theory of computation** → **Lambda calculus**; **Program semantics**; **Probabilistic computation**; *Linear logic*;

Additional Key Words and Phrases: Denotational Semantics, PCF

ACM Reference Format:

Thomas Ehrhard, Michele Pagani, and Christine Tasson. 2018. Measurable Cones and Stable, Measurable Functions: A Model for Probabilistic Higher-Order Programming. *Proc. ACM Program. Lang.* 2, POPL, Article 59 (January 2018), 28 pages. <https://doi.org/10.1145/3158147>

1 INTRODUCTION

Around the 80's, people started to apply formal methods to the analysis and design of probabilistic programming languages. In particular, Kozen [1981] defined a denotational semantics for a first-order while-language endowed with a random real number generator. In that setting, programs can be seen as stochastic kernels between measurable spaces: the possible configurations of the memory are described by measurable spaces, with the measurable sets expressing the observables, while kernels define the probabilistic transformation of the memory induced by program execution. For example, a while-program using n variables taking values in the set \mathbb{R} of real numbers is a stochastic kernel K over the Lebesgue σ -algebra on \mathbb{R}^n (see Compendium of Measures and Kernels, Section 2) — i.e. K is a function taking a sequence $\vec{r} \in \mathbb{R}^n$ and a measurable set $U \subseteq \mathbb{R}^n$ and giving a real number $K(\vec{r}, U) \in [0, 1]$, which is the probability of having the memory (i.e. the values of the n variables) within U after having executed the program with the memory initialized as \vec{r} .

Kozen's approach cannot be trivially extended to higher-order types, because there is no clear notion of measurable subset for a functional space, e.g. we do not know which measurable space can describe values of type, say, $\mathbb{R} \rightarrow \mathbb{R}$ (see [Aumann 1961] for details).

Authors' addresses: Thomas Ehrhard, IRIF UMR 8243, Université Paris Diderot, Sorbonne Paris Cité, CNRS, F-75205 Paris, France, ehrh@irif.fr; Michele Pagani, IRIF UMR 8243, Université Paris Diderot, Sorbonne Paris Cité, CNRS, F-75205 Paris, France, pagani@irif.fr; Christine Tasson, IRIF UMR 8243, Université Paris Diderot, Sorbonne Paris Cité, CNRS, F-75205 Paris, France, tasson@irif.fr.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2018 Copyright held by the owner/author(s). Publication rights licensed to the Association for Computing Machinery. 2475-1421/2018/1-ART59

<https://doi.org/10.1145/3158147>

Panangaden [1999] reframed the work by Kozen in a categorical setting, using the category **Kern** of stochastic kernels. This category has been presented as the Kleisli category of the so-called Giry’s monad [Giry 1982] over the category **Meas** of measurable spaces and measurable functions. One can precisely state the issue for higher-order types in this framework — both **Meas** and **Kern** are cartesian categories but not closed.

The quest for a formal syntactic semantics of higher-order probabilistic programming had more success. We mention in particular Park et al. [2008], proposing a probabilistic functional language λ_{\circ} based on sampling functions. This language has a type \mathcal{R} of sub-probabilistic distributions over the set of real numbers¹, i.e. measures over the Lebesgue σ -algebra on \mathbb{R} with total mass at most 1. Using the usual functional primitives (in particular recursion) together with the uniform distribution over $[0, 1]$ and a sampling construct, the authors encode various methods for generating distributions (like the inverse transform method and rejection sampling) and computing properties about them (such as approximations for expectation values, variances, etc). The amazing feature of λ_{\circ} is its rich expressiveness as witnessed by the number of examples and applications detailed in [Park et al. 2008], showing the relevance of the functional paradigm for probabilistic programming.

Until now, λ_{\circ} lacked a denotation model, [Park et al. 2008] sketching only an operational semantics. In particular, the correctness proof of the encodings follows a syntactic reasoning which is not compositional. Our paper fills this gap, giving a denotational model to a variant of λ_{\circ} . As a byproduct, we can check program correctness in a straight way by applying to program denotations the standard laws of calculus (Example 7.3.7.4), even for recursive programs (Example 7.9). This method is justified by the Adequacy Theorem 7.12 stating the correspondence between the operational and the denotational semantics.

If we restrict the language to *countable* data types (like booleans and natural numbers, excluding the real numbers), then the situation is much simpler. Indeed, any distribution over a countable set is *discrete*, i.e. it can be described as a linear combination of its possible outcomes and there is no need of a notion of measurable space. In previous papers [Ehrhard et al. 2011, 2014; Ehrhard and Tasson 2016], we have shown that the category **PCoh**₁ of *probabilistic coherence spaces* and entire functions gives fully abstract denotational models of functional languages extended with a random *natural number* generator. The main goal of this work is to generalize these models in order to account for *continuous* data types also.

The major difficulty for such a generalization is that a probabilistic coherence space is defined with respect to a kind of canonical basis (called *web*) that, at the level of ground types, corresponds to the possible samples of a distribution. For continuous data types, these webs should be replaced by measurable spaces, and then one is stuck on the already mentioned impossibility of associating a measurable space with a functional type — both **Meas** and **Kern** being not cartesian closed.

Our solution is to replace probabilistic coherence spaces with cones [Andô 1962], already used by Selinger [2004], allowing for an axiomatic presentation not referring to a web. A cone is similar to a normed vector space, but with non-negative real scalars (Definition 4.1). Any probabilistic coherence space can be seen as a cone (Example 4.4) as well as the set $\text{Meas}(X)$ of all bounded measures over a measurable space X (Example 4.6). In particular, the cone $\text{Meas}(\mathbb{R})$ associated with the Lebesgue σ -algebra on \mathbb{R} will be our interpretation of the ground type \mathcal{R} .

What about functional types, e.g. $\mathcal{R} \rightarrow \mathcal{R}$? Selinger [2004] studied the notion of Scott continuous maps between cones, i.e. monotone non-decreasing bounded maps which commute with the lub of non-decreasing sequences². The set of these functions also forms a cone with the algebraic

¹In [Park et al. 2008] \mathcal{R} is written $\circ\text{real}$. One should consider *sub*-probabilistic distributions because program evaluation may diverge.

²Actually, Selinger considers lubs of directed sets, but non-decreasing chains are enough for our purposes. Moreover, because we need to use the monotone convergence theorem for guaranteeing the measurability of these lubs in function

operations defined pointwise. However, this cone construction does not yield a cartesian closed category, namely the currying of a Scott continuous map can fail to be monotone non-decreasing, hence Scott continuous (see discussion in Section 4.1.1). The first relevant contribution of our paper is then to introduce a notion of *stable* map, meaning Scott continuous *and* “absolutely monotonic” (Definition 4.14), which solves the problem about currying and gives a cartesian closed category.

We borrow the term of “stable function” from Berry’s analysis of sequential computation [Berry 1978]. In fact, our definition is deeply related with a notion of “probabilistic” sequentiality, as we briefly mention in Section 4.1.1 showing that it rejects the “parallel or” (but not the “Gustave function”).

The notion of stability is however not enough to interpret all primitives of probabilistic functional programming. One should be able to integrate at least first-order functions in order to sample programs denoting probabilistic distributions (e.g. see the denotation of the `let` construct in Figure 5). The problem is that there are stable functions which are not measurable, so not Lebesgue integrable (Section 5). We therefore equip the cones with a notion of measurability tests (Definition 5.1), inducing a notion of measurable paths (Definition 5.2) in a cone. In the case the cone is associated with a standard measurable space X , *i.e.* it is of the form $\text{Meas}(X)$, then the measurability tests are the measurable sets of X . However, at higher-order types the definition is less immediate. The crucial point is that the measurable paths in $\text{Meas}(X)$ are Lebesgue integrable, as expected (Section 6.3). We then call *measurable* a stable map preserving measurable paths and we prove that it gives a cartesian closed category, denoted \mathbf{Cstab}_m (Figure 4 and Theorem 6.7).

To illustrate the expressiveness of \mathbf{Cstab}_m we consider a variant of Scott and Plotkin’s PCF [Plotkin 1977] with numerals for real numbers, a constant sample denoting the uniform distribution over $[0, 1]$ and a `let` construct over the ground type. This language is as expressive as λ_\circ of Park et al. [2008] (namely, the `let` construct corresponds to the sampling of λ_\circ). The only notable difference lies in the call-by-name operational semantics (Figure 3) that we adopt, while [Park et al. 2008] follows a call-by-value strategy.³ Our choice is motivated by the fact that the call-by-name model is simpler to present than the call-by-value one. We plan to detail this latter in a forthcoming paper.

We also decided not to consider the so-called soft-constraints, which are implemented in e.g. [Borgström et al. 2016; Staton 2017; Staton et al. 2016] with a construct called `score`. This can be added to our language by using a kind of exception monad in order to account for the possible failure of normalization, as detailed in [Staton 2017] (see Remark 2). Also in this case we prefer to omit this feature for focussing on the true novelties of our approach — the notions of stability and measurability.

Let us underline that although the definition of \mathbf{Cstab}_m and the proof of its cartesian closeness are not trivial, the denotation of the programs (Figure 5) is completely standard, extending the usual interpretation of PCF programs as Scott continuous functions [Plotkin 1977]. We prove the soundness (Proposition 7.8) and the adequacy (Theorem 7.12) of \mathbf{Cstab}_m . A major byproduct of this result is then to make it possible to reason about higher-order programs as functions between cones, which is quite convenient when working with programs acting on measures.

To conclude, let us comment Figure 1, sketching the relations between the category \mathbf{Cstab}_m achieved here and the category $\mathbf{PCoh}_!$ of probabilistic coherence spaces and entire functions which has been the starting point of our approach. The two categories give models of the functional primitives (PCF-like languages), but $\mathbf{PCoh}_!$ is restricted to discrete data types, while \mathbf{Cstab}_m extends

spaces, completeness wrt. arbitrary directed sets would be a too strong requirement (see Section 6.2): a crucial feature of measurable sets is that they are closed under *countable* (and not arbitrary) unions.

³Let us underline that our `let` construct does not allow to encode the call-by-value strategy at higher-order types, since it is restricted to the ground type \mathcal{R} . See Section 3 for more details.

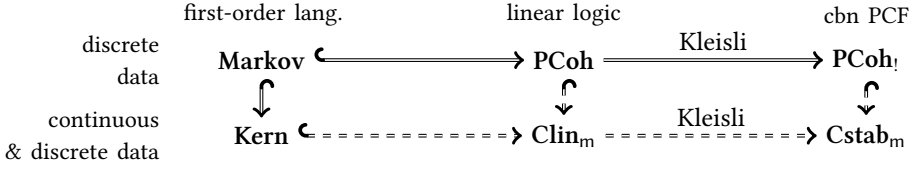


Fig. 1. Relationship between probabilistic coherence categories and \mathbf{Cstab}_m . Dashed arrows are conjectures.

the model to continuous types. We guess this extension to be conservative, hence the arrow is hooked but just dashed. We are even convinced that \mathbf{Cstab}_m is the result of a Kleisli construction from a more fundamental model \mathbf{Clin}_m of (intuitionistic) linear logic, based on positive cones and measurable, Scott continuous and linear functions. We plan to study \mathbf{Clin}_m in an extended version of this paper as a category extending the category \mathbf{Kern} of measurable spaces and stochastic kernels. This would close the loop and further confirm the analogy with $\mathbf{PCoh}_!$, which is the Kleisli category associated with the exponential comonad of the model based on the category \mathbf{PCoh} of Scott continuous and linear functions between probabilistic coherence spaces, this latter containing the category \mathbf{Markov} of Markov chains as a full sub-category.

Contents. This paper needs a basic knowledge of measure theory: we briefly recall in Section 2 the main notions and notations used. Section 3 presents the programming language PPCF— the probabilistic variant of PCF we use for showing the expressiveness of our model. Figure 2 gives the grammar of terms and the typing rules, while Equation (5) and Figure 3 define the kernel Red describing the stochastic operational semantics. Our first main contribution is presented in Section 4: after having recalled Selinger’s definition of cone (Definition 4.1) we study our notion of absolutely monotonic map (Definition 4.14), or equivalently pre-stable map (Definition 4.17 and Theorem 4.18) and we prove that it composes (Theorem 4.26). Stable maps are absolutely monotonic and Scott-continuous (Definition 4.27). Section 5 introduces our second main contribution, which is the notion of measurability test (Definition 5.1) and measurable map (Definition 5.5), giving the category \mathbf{Cstab}_m (Definition 5.5). Section 6 presents the cartesian closed structure of \mathbf{Cstab}_m , summarized in Figure 4. Finally, Section 7 details the model of PPCF given by \mathbf{Cstab}_m (Figure 5) and states soundness (Proposition 7.8) and adequacy (Theorem 7.12). Section 8 discusses the previous literature. Because of space limits, many proofs are omitted and available in [Ehrhard et al. 2017].

Notations. We use $\#I$ for the cardinality of a set I . The set of non-negative real numbers is \mathbb{R}^+ and its elements are denoted α, β, \dots . General real numbers are denoted r, s, t, \dots . The set of non-zero natural numbers is \mathbb{N}^+ . The greek letter λ will denote the Lebesgue measure over \mathbb{R} , $\lambda_{[0,1]}$ being its restriction to the unit interval. Given a measurable space X and an $x \in X$, we use δ_x for the Dirac measure over X : $\delta_x(U)$ is equal to 1 if $x \in U$ and to 0 otherwise. We also use χ_U to denote the characteristic function of U which is defined as $\chi_U(x)$ is equal to 1 if $x \in U$ and to 0 otherwise. We use \mathcal{M}^n for the set of measurable functions $\mathbb{R}^n \rightarrow \mathbb{R}^+$. We use $F(_)$ to denote the map $x \mapsto F(x)$.

2 COMPENDIUM OF MEASURES AND KERNELS

A σ -algebra Σ_X on a set X is a family of subsets of X that is nonempty and closed under complements and countable unions, so that $\emptyset, X \in \Sigma_X$. A *measurable space* is a pair (X, Σ_X) of a set X equipped with a σ -algebra Σ_X . A *measurable set* of (X, Σ_X) is an element of Σ_X . From now on, we will denote a measurable space (X, Σ_X) simply by its underlying set X , whenever the σ -algebra is clear or irrelevant. We consider \mathbb{R} and \mathbb{R}^+ as measurable spaces equipped with the Lebesgue σ -algebra, generated by the open intervals. A *bounded measure* on a measurable space X is a map $\mu : \Sigma_X \rightarrow \mathbb{R}^+$ satisfying $\mu(\bigsqcup_{i \in I} S_i) = \sum_{i \in I} \mu(S_i)$ for any countable family $\{S_i\}_{i \in I}$ of disjoint sets in Σ_X . We call

μ a *probability* (resp. *subprobability*) *measure*, whenever $\mu(X) = 1$ (resp. $\mu(X) \leq 1$). When μ is a measure on \mathbb{R}^n , we often call it a *distribution*.

A *measurable function* $f : (X, \Sigma_X) \rightarrow (Y, \Sigma_Y)$ is a function $f : X \rightarrow Y$ such that $f^{-1}(U) \in \Sigma_X$ for every $U \in \Sigma_Y$. The *pushforward measure* $f_*\mu$ from a measure μ on X along a measurable map f is defined as $(f_*\mu)(U) = \mu(f^{-1}(U))$, for every $U \in \Sigma_Y$.

These notions have been introduced in order to define the *Lebesgue integral* $\int_X f(x)\mu(dx)$ of a generic measurable function $f : X \rightarrow \mathbb{R}$ with respect to a measure μ over X . This paper uses only basic facts about the Lebesgue integral which we do not detail here.

Measures are special cases of kernels. A *bounded kernel* K from X to Y is a function $K : X \times \Sigma_Y \rightarrow \mathbb{R}^+$ such that: (i) for every $x \in X$, $K(x, _)$ is a bounded measure over Y ; (ii) for every $U \in \Sigma_Y$, $K(_, U)$ is a measurable map from X to \mathbb{R}^+ . A *stochastic kernel* K is a kernel such that $K(x, _)$ is a sub-probability measure for every $x \in X$. Notice that a bounded measure (resp. sub-probability measure) μ over X can be seen as a particular bounded kernel (resp. stochastic kernel) from the singleton measurable space $(\{\star\}, \{\emptyset, \{\star\}\})$ to X .

Categorical Approach. We use two categories having measurable spaces as objects, denoted respectively **Meas** and **Kern**.

The category **Meas** has measurable functions as morphisms. This category is cartesian (but not cartesian closed), the *cartesian product* $(X, \Sigma_X) \times (Y, \Sigma_Y)$ of (X, Σ_X) and (Y, Σ_Y) is $(X \times Y, \Sigma_X \otimes \Sigma_Y)$, where $X \times Y$ is the set-theoretic product and $\Sigma_X \otimes \Sigma_Y$ is the σ -algebra generated by the rectangles $U \times V$, where $U \in \Sigma_X$ and $V \in \Sigma_Y$. It is easy to check that the usual projections are measurable maps, as well as that the set-theoretic pairing $\langle f, g \rangle$ of two functions $f : Z \rightarrow X$, $g : Z \rightarrow Y$ is a measurable map from Z to $X \times Y$, whenever f, g are measurable.

The category **Kern** has stochastic kernels as morphisms⁴. Given a stochastic kernel H from X to Y and K from Y to Z , the *kernel composition* $K \circ H$ is a stochastic kernel from X to Z defined as, for every $x \in X$ and $U \in \Sigma_Z$:

$$(K \circ H)(x, U) = \int_Y K(y, U)H(x, dy). \quad (1)$$

Notice that the above integral is well-defined because $H(x, _)$ is a stochastic measure from condition (i) on kernels and $K(_, U)$ is a measurable function from condition (ii). A simple application of Fubini's theorem gives the associativity of the kernel composition. The *identity kernel* is the function mapping (x, U) to 1 if $x \in U$ and to 0 otherwise.

Unlike **Meas**, we consider a tensor product \otimes in **Kern** which is a symmetric monoidal product but not the cartesian product⁵. The action of \otimes over the objects X, X' is defined as the cartesian product in **Meas**, so that we still denote it as $X \times X'$. The tensor of a kernel K from X to Y and K' from X' to Y' is the kernel $K \otimes K'$ given as follows, for $(x, x') \in X \times X'$ and $U \in \Sigma_Y$, $U' \in \Sigma_{Y'}$:

$$K \otimes K'((x, x'), U \times U') = K(x, U)K'(x', U') \quad (2)$$

Notice that **Kern** is not closed with respect to \otimes . Recall that a measure can be seen as a kernel from the singleton measurable space, so that Equation (2) defines also a *tensor product* $\mu \otimes \mu'$ between measures over resp. Y and Y' .

The category **Kern** has also *countable coproducts*. Given a countable family $\{(X_i, \Sigma_i)\}_{i \in I}$ of measurable spaces, the coproduct $\coprod_{i \in I} (X_i, \Sigma_i)$ has as underlining set the disjoint union $\cup_{i \in I} X_i \times \{i\}$ of the X_i 's, and as the σ -algebra the one generated by $\cup_{i \in I} U_i \times \{i\}$ disjoint union of $U_i \in \Sigma_i$. The injections ι_j from X_j to $\coprod_{i \in I} X_i$ are defined as $\iota_j(x, \cup_{j \in I} U_j \times \{j\}) = \chi_{U_j}(x)$. Given a family K_i from X_i to Y , the copairing $[K_i]_{i \in I}$ from $\coprod_{i \in I} X_i$ to Y is defined by $[K_i]_{i \in I}((x, j), U) = K_j(x, U)$.

⁴One can well define the category of bounded kernels also, but this is not used in this paper.

⁵Indeed, **Kern** has cartesian products, but we will not use them.

$$\begin{array}{c}
\frac{}{\Delta, x : A \vdash x : A} \quad \frac{\Delta, x : A \vdash M : B}{\Delta \vdash \lambda x^A. M : A \rightarrow B} \quad \frac{\Delta \vdash M : A \rightarrow B \quad \Delta \vdash N : A}{\Delta \vdash (MN) : B} \quad \frac{\Delta \vdash M : A \rightarrow A}{\Delta \vdash (YM) : A} \\
\\
\frac{r \in \mathbb{R}}{\Delta \vdash \underline{r} : \mathcal{R}} \quad \frac{f \text{ meas. map } \mathbb{R}^n \rightarrow \mathbb{R} \quad \Delta \vdash M_i : \mathcal{R}, \forall i \leq n}{\Delta \vdash \underline{f}(M_1, \dots, M_n) : \mathcal{R}} \quad \frac{\Delta \vdash L : \mathcal{R} \quad \Delta \vdash M : \mathcal{R} \quad \Delta \vdash N : \mathcal{R}}{\Delta \vdash \text{ifz}(L, M, N) : \mathcal{R}} \\
\\
\frac{}{\Delta \vdash \text{sample} : \mathcal{R}} \quad \frac{\Delta \vdash M : \mathcal{R} \quad \Delta, x : \mathcal{R} \vdash N : \mathcal{R}}{\Delta \vdash \text{let}(x, M, N) : \mathcal{R}}
\end{array}$$

Fig. 2. The grammar of terms of PPCF and their typing rules. The variable x belongs to a fixed countable set of variables \mathcal{V} . The metavariable \underline{f} ranges over a fixed countable set \mathcal{C} of functional identifiers, while the metavariable \underline{r} may range on the whole \mathbb{R} .

Actually, the categories **Meas** and **Kern** can be related in a very similar way as the relation between the categories **Set** (of sets and functions) and **Rel** (of sets and relations). In fact, **Kern** corresponds to the Kleisli category of the so-called Giry’s monad over **Meas** [Giry 1982], exactly has the category **Rel** of relations is the Kleisli category of the powerset monad over **Set** (see [Panangaden 1999]). Since this paper does not use this construction, we do not detail it.

3 THE PROBABILISTIC LANGUAGE PPCF

3.1 Types and Terms

We give in Figure 2 the grammar of our probabilistic extension of PCF, briefly PPCF, together with the typing rules. The types are generated by $A, B ::= \mathcal{R} \mid A \rightarrow B$, where the constant \mathcal{R} is the ground type for the set of real numbers. We denote by $\Lambda^{\Gamma \vdash A}$ the set of terms typeable within the sequent $\Gamma \vdash A$. We write simply Δ if the typing sequent is not important or clear from the context.

The first line of Figure 2 contains the usual constructs of the simply typed λ -calculus extended to the fix-point combinator Y for any type A . The second line describes the primitives dealing with the ground type \mathcal{R} . Our goal is to show the expressiveness of the category **Cstab_m** introduced in the next section, therefore PPCF is an ideal language and does not deal with the issues about a realistic implementation of computations over real numbers. We refer the interested reader to e.g. [Escardó 1996; Vuillemin 1988]. We will suppose that the meta-variable \underline{f} ranges over a fixed countable set \mathcal{C} of basic measurable functions over real numbers. Examples of these functions include addition $+$, comparison $>$, and equality $=$; they are often written in infix notation. When clear from the context, we sometimes write f for \underline{f} . To be concise, we consider only the ground type \mathcal{R} , the boolean operators (like $>$ or $=$) then evaluate to 1 or 0, representing resp. true and false.

The third line of Figure 2 gives the “probabilistic core” of PPCF. The constant **sample** stands for the uniform distribution over $[0, 1]$, i.e. the Lebesgue measure $\lambda_{[0,1]}$ restricted to the unit interval. The fact that PPCF has only this distribution as a primitive is not limiting, in fact many other probabilistic measures (like binomial, geometric, gaussian or exponential distribution) can be defined from $\lambda_{[0,1]}$ and the other constructs of the language, see e.g. [Park et al. 2008] and Example 3.3. The **let** construction allows a call-by-value discipline over the ground type \mathcal{R} : the execution of **let**(x, M, N) will sample a value (i.e. a real number r) from a probabilistic distribution M and will pass it to N by replacing every free occurrence of x in N with \underline{r} . This primitive⁶ is

⁶Notice that this primitive corresponds to the **sample** construction in [Park et al. 2008].

essential for the expressiveness of PPCF and will be discussed both operationally and semantically in the next sections.

PPCF has a limited number of constructs, but it is known that many probabilistic primitives can be introduced as syntactic sugar from the ones in PPCF, as shown in the following examples. We will prove the correctness of these encodings using the denotational semantics (Section 7), this latter corresponding to the program operational behavior by the adequacy property (Theorem 7.12).

Example 3.1 (Extended branching). Let U be a measurable set of real numbers whose characteristic function χ_U is in C , let $L \in \Lambda^{\Gamma \vdash \mathcal{R}}$ and $M, N \in \Lambda^{\Gamma \vdash A}$ for $A = B_1 \rightarrow \dots B_n \rightarrow \mathcal{R}$. Then the term $\Gamma \vdash \text{if}(L \in U, M, N) : A$, branching between M and N according to the outcome of L being in U , is a syntactic sugar for $\lambda x^{B_1} \dots \lambda x^{B_n} . \text{ifz}(\chi_U(L), Nx_1 \dots x_n, Mx_1 \dots x_n)$.⁷

Example 3.2 (Extended let). Similarly, the `let` constructor can be extended to any output type $A = B_1 \rightarrow \dots B_n \rightarrow \mathcal{R}$. Given $M \in \Lambda^{\Gamma \vdash \mathcal{R}}$ and $N \in \Lambda^{\Gamma, x : \mathcal{R} \vdash A}$, we denote by $\text{let}(x, M, N)$ the term $\lambda x^{B_1} \dots \lambda x^{B_n} . \text{let}(x, M, Nx_1 \dots x_n)$ which is in $\Lambda^{\Gamma \vdash A}$. However we do not know in general how to extend the type of the bound variable x to higher types in this model. The issue is clear at the denotational level, where the `let` construction is expressed with an integral (see Figure 5). With each ground type, we associate a positive cone $\text{Meas}(X)$ which is generated by a measurable space X . At higher types, the associated cones do not have to be generated by measurable spaces.

Notice that, because of this restriction on the type of the bound variable x , our `let` constructor does not allow to embed into our language the full call-by-value PCF.

Example 3.3 (Distributions). The Bernoulli distribution takes the value 1 with some probability p and the value 0 with probability $1 - p$. It can be expressed as the term `bernoulli` of type $\mathcal{R} \rightarrow \mathcal{R}$, taking the parameter p as argument and testing whether sample draws a value within the interval $[0, p]$, i.e. $\lambda p . \text{let}(x, \text{sample}, x \leq p)$.

The exponential distribution at rate 1 is specified by its density e^{-x} . It can be implemented as the term `exp` of type \mathcal{R} by the inversion sampling method: $\text{let}(x, \text{sample}, -\log(x))$.

The standard normal distribution (gaussian with mean 0 and variance 1) is defined by its density $\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$. We use the Box Muller method to encode the normal distribution $\text{normal} = \text{let}(x, \text{sample}, \text{let}(y, \text{sample}, (-2\log(x))^{\frac{1}{2}} \cos(2\pi y)))$.

We can encode the Gaussian distribution as a function of the expected value x and standard deviation σ by $\text{gauss} = \lambda x \lambda \sigma . \text{let}(y, \text{normal}, (\sigma y) \pm x)$.

Example 3.4 (Conditioning). Let U be a measurable set of real numbers such that $\chi_U \in C$, we define a term `observe`(U) of type $\mathcal{R} \rightarrow \mathcal{R}$, taking a term M and returning the renormalization of the distribution of M on the only samples that satisfy U : $\text{observe}(U) = \lambda m . \gamma(\lambda y . \text{let}(x, m, \text{if}(x \in U, x, y)))$. This corresponds to the usual way of implementing conditioning by rejection sampling: the evaluation of `observe`(U) M will sample a real r from M , if $r \in U$ holds then the program returns r , otherwise it iterates the procedure. Notice that `observe`(U) M makes a crucial use of sampling. The program $\lambda m . \gamma(\lambda y . \text{if}(m \in U, m, y))$ has a different behavior, because the two occurrences of m correspond in this case to two independent random variables (see Example 3.10 below).

Example 3.5 (Monte Carlo Simulation). An example using the possibility of performing independent copies of a random variable is the encoding of the n -th estimate of an expectation query. The expected value of a measurable function f with respect to distribution μ is defined as $\int_{\mathbb{R}} f(x) \mu(dx)$. The Monte Carlo method relies on the laws of large number: if x_1, \dots, x_n are independent and

⁷The swap between M and N is due to fact that `ifz` is the test to zero.

$$\begin{aligned}
(\lambda x.M)N &\rightarrow M\{N/x\} & \underline{f}(\underline{r}_1, \dots, \underline{r}_n) &\rightarrow \underline{f}(r_1, \dots, r_n) \\
\text{ifz}(\underline{r}, M, N) &\rightarrow \begin{cases} M & \text{if } r = 0, \\ N & \text{otherwise.} \end{cases} & \text{let}(x, \underline{r}, M) &\rightarrow M\{\underline{r}/x\} \\
YM &\rightarrow M(YM) & \text{sample} &\rightarrow \underline{r} \quad \text{for any } r \in [0, 1].
\end{aligned}$$

(a) Reduction of a PPCF redex.

$$E[\] ::= [\] \mid E[\]M \mid \text{ifz}(E[\], M, N) \mid \text{let}(x, E[\], N) \mid \underline{f}(\underline{r}_1, \dots, \underline{r}_{i-1}, E[\], M_{i+1}, \dots, M_n)$$

$$E[M] \rightarrow E[N], \text{ whenever } M \rightarrow N$$

(b) Grammar of the evaluation contexts and context closure of the reduction.

Fig. 3. One-step operational semantics of PPCF.

identically distributed random variables of equal probability distribution μ , then the n -th estimate $\frac{f(x_1) + \dots + f(x_n)}{n}$ converges almost surely to $\int_{\mathbb{R}} f(x)\mu(dx)$. For any integer n , we can then en-

code the n -th estimate combinator by expectation $_n = \lambda f. \lambda m. \overbrace{(f(m) + \dots + f(m))}^{n \text{ times}} / n$ of type $(\mathcal{R} \rightarrow \mathcal{R}) \rightarrow \mathcal{R} \rightarrow \mathcal{R}$. Notice that it is crucial here that the variable m has n occurrences representing n independent random variables, this being in contrast with Example 3.4 (see also Example 3.10).

3.2 Operational Semantics

The operational semantics of PPCF is a Markov process defined starting from the rewriting rules of Figure 3, extending the standard call-by-name reduction of PCF [Plotkin 1977]. The probabilistic primitive `sample` draws a possible value from $[0, 1]$, like in [Park et al. 2008]. The fact that we are sampling from the uniform distribution and not from other distributions with equal support appears in the definition of the stochastic kernel `Red` (Equation (5)). In order to define this kernel, we equip Λ with a structure of measurable space (Equation (4)). This defines a σ -algebra Σ_Λ of sets of terms equivalent to the one given in e.g. [Borgström et al. 2016; Staton et al. 2016] for slightly different languages. Similarly to [Staton et al. 2016], our definition is explicitly given by a countable coproduct of copies of the Lebesgue σ -algebra over \mathbb{R}^n (for $n \in \mathbb{N}$, see Equations (3)), while in [Borgström et al. 2016] the definition is based on a notion of distance between λ -terms. The two definitions give the same measurable space, but the one adopted here allows to take advantage of the categorical structure of **Kern**.

REMARK 1. *The operational semantics associates with a program M a probabilistic distribution \mathcal{D} of values describing the possible outcomes of the evaluation of M . There are actually two different “styles” for giving \mathcal{D} : one based on samplings and another one, adopted here, based on stochastic kernels. Borgström et al. [2016] proved that the two semantics are equivalent, giving the same distribution \mathcal{D} .*

The “sampling semantics” associates with M a function mapping a trace of random samples to a weight, expressing the likelihood of getting that trace of samples from M . The final distribution \mathcal{D} is then calculated by integrating this function over the space of the possible traces, equipped with a suitable measure. This approach is usually adopted when one wants to underline an implementation of the probabilistic primitives of the language via a sampling algorithm, e.g. [Park et al. 2008].

The “kernel-based semantics” instead describes program evaluation as a discrete-time Markov process over a measurable space of states given by the set of programs $((\Lambda, \Sigma_\Lambda)$ in our case). The transition of the process is given by a stochastic kernel (here Red defined in Equation (5)) and then the probabilistic distribution \mathcal{D} of values associated with a term is given by the supremum of the family of all finite iterations of the kernel (Red^∞ , Equation (6)). This latter approach is more suitable when comparing the operational semantics with a denotational model (in order to prove soundness and adequacy for example) and it is then the one adopted in this paper.

A *redex* is a term in one of the forms at left-hand side of the \rightarrow defined in Figure 3a. A *normal form* is a term M which is no more reducible under \rightarrow . Notice that the closed normal forms of ground type \mathcal{R} are the real numerals. The definition of the evaluation context (Figure 3b) is the usual one defining the deterministic lazy call-by-name strategy: we do not reduce under an abstraction and there is always at most one redex to reduce, as stated by the following standard lemma.

LEMMA 3.6. *For any term M , either M is a normal form or there exists a unique redex R and an evaluation context $E[\]$ such that $M = E[R]$.*

It is standard to check that the property of subject reduction holds (if $\Gamma \vdash M : A$ and $M \rightarrow N$, then $\Gamma \vdash N : A$).

From now on, let us fix an enumeration without repetitions $(z_i)_{i \in \mathbb{N}}$ of variables of type \mathcal{R} . Notice that any term $M \in \Lambda^{\Gamma \vdash A}$ with n different occurrences of real numerals, can be decomposed *univocally* into a term $z_1 : \mathcal{R}, \dots, z_n : \mathcal{R}, \Gamma \vdash S : A$ without real numerals and a substitution $\sigma = \{r_1/z_1, \dots, r_n/z_n\}$, such that: (i) $M = S\sigma$; (ii) each z_i occurs exactly once in S ; (iii) z_i occurs before z_{i+1} reading the term from left to right. Because of this latter condition, we can omit the name of the substituted variables, writing simply $S\vec{r}$ with $\vec{r} = (r_1, \dots, r_n)$. We denote by $\Lambda_n^{\Gamma \vdash A}$ the set of terms in $\Lambda^{z_1:\mathcal{R}, \dots, z_n:\mathcal{R}, \Gamma \vdash A}$ with no occurrence of numerals and respecting conditions (ii) and (iii) above. We let S, T vary over such real-numeral-free terms.

Given $S \in \Lambda_n^{\Gamma \vdash A}$ we then define the set $\Lambda_S^{\Gamma \vdash A} = \{M \in \Lambda^{\Gamma \vdash A} \text{ s.t. } \exists \vec{r} \in \mathbb{R}^n, M = S\vec{r}\}$. The bijection $s : \Lambda_S^{\Gamma \vdash A} \rightarrow \mathbb{R}^n$ given by $s(S\vec{r}) = \vec{r}$ endows $\Lambda_S^{\Gamma \vdash A}$ with a σ -algebra isomorphic to $\Sigma_{\mathbb{R}^n} : U \in \Sigma_{\Lambda_S^{\Gamma \vdash A}}$ iff $s(U) \in \Sigma_{\mathbb{R}^n}$. The fact that $\Lambda_n^{\Gamma \vdash A}$ is countable and that **Kern** has countable coproducts (see Section 2), allows us to define the measurable space of PPCF terms of type $\Gamma \vdash A$ as the coproduct:

$$(\Lambda^{\Gamma \vdash A}, \Sigma_{\Lambda^{\Gamma \vdash A}}) = \coprod_{n \in \mathbb{N}, S \in \Lambda_n^{\Gamma \vdash A}} (\Lambda_S^{\Gamma \vdash A}, \Sigma_{\Lambda_S^{\Gamma \vdash A}}) \quad (3)$$

Spelling out the definition, a subset $U \subseteq \Lambda^{\Gamma \vdash A}$ is measurable if and only if:

$$\forall n, \forall S \in \Lambda_n^{\Gamma \vdash A}, \{\vec{r} \text{ s.t. } S\vec{r} \in U\} \in \Sigma_{\mathbb{R}^n} \quad (4)$$

Given a set $U \subseteq \mathbb{R}$, we denote by \underline{U} the set of numerals associated with the real numbers in U . Of course U is measurable iff \underline{U} is measurable. The following lemma allows us to define Red and Red^∞ .

LEMMA 3.7. *Given $\Gamma, x : B \vdash M : A$ the function $\text{Subst}_{x,M}$ mapping $N \in \Lambda^{\Gamma \vdash B}$ to $M\{N/x\} \in \Lambda^{\Gamma \vdash A}$ is measurable.*

Given a term $M \in \Lambda$ and a measurable set $U \subseteq \Lambda$ we define $\text{Red}(M, U) \in [0, 1]$ depending on the form of M , according to Lemma 3.6:

$$\text{Red}(M, U) = \begin{cases} \delta_{E[N]}(U) & \text{if } M = E[R], R \rightarrow N \text{ and } R \neq \text{sample}, \\ \lambda \{r \in [0, 1] \text{ s.t. } E[r] \in U\} & \text{if } M = E[\text{sample}], \\ \delta_M(U) & \text{if } M \text{ normal form.} \end{cases} \quad (5)$$

The last case sets the normal forms as accumulation points of Red , so that $\text{Red}(M, U)$ gives the probability that we observe U after *at most one* reduction step applied to M . The definition in the case of $E[\text{sample}]$ specifies that sample is drawing from the uniform distribution over $[0, 1]$. Notice that, if $U \subseteq \mathbb{R}$ is measurable, then the set $\{r \in [0, 1] \text{ s.t. } E[r] \in U\}$ is measurable by Lemma 3.7. The definition of Red extends to a continuous setting the operational semantics Markov chain of [Danos and Ehrhard 2011; Ehrhard et al. 2014].

PROPOSITION 3.8. *For any sequent $\Gamma \vdash A$, the map Red is a stochastic kernel from $\Lambda^{\Gamma \vdash A}$ to $\Lambda^{\Gamma \vdash A}$.*

PROOF (SKETCH). The fact that $\text{Red}(M, _)$ is a measure is an immediate consequence of the definition of Red and the fact that any evaluation context $E[\]$ defines a measurable map (Lemma 3.7).

Given a measurable set $U \subseteq \Lambda^{\Gamma \vdash A}$, we must prove that $\text{Red}(_, U)$ is a measurable function from $\Lambda^{\Gamma \vdash A}$ to $[0, 1]$. Since $\Lambda^{\Gamma \vdash A}$ can be written as the coproduct in Equation (3), it is sufficient to prove that for any n and $S \in \Lambda_n^{\Gamma \vdash A}$, $\text{Red}_S(_, U) : \Lambda_S^{\Gamma \vdash A} \rightarrow [0, 1]$ is a measurable function. One reasons by case study on the shape of S , using Lemma 3.6 and the definition of a redex. \square

We can then iterate Red using the composition of stochastic kernels (Equation (1)):

$$\text{Red}^{n+1}(M, U) = (\text{Red} \circ \text{Red}^n)(M, U) = \int_{\Lambda} \text{Red}(t, U) \text{Red}^n(M, dt),$$

this giving the probability that we observe U after at most $n + 1$ reduction steps from M . Because the normal forms are accumulation points, one can prove by induction on n that:

LEMMA 3.9. *Let $\Gamma \vdash M : A$ and let U be a measurable set of normal forms in $\Lambda^{\Gamma \vdash A}$. The sequence $(\text{Red}^n(M, U))_n$ is monotone non-decreasing.*

We can then define, for $M \in \Lambda$ and U a measurable set of normal forms, the limit

$$\text{Red}^\infty(M, U) = \sup_n (\text{Red}^n(M, U)). \quad (6)$$

In particular, if M is a closed term of ground type \mathcal{R} , the only normal forms that M can reach are numerals, in this case $\text{Red}^\infty(M, _)$ corresponds to the probabilistic distribution over \mathbb{R} which is computed by M according to the operational semantics of PPCF (Remark 1).

Example 3.10. In order to make clear the difference between a call-by-value and a call-by-name reduction in a probabilistic setting, let us consider the following two terms:

$$M = (\lambda x. (x = x)) \text{sample}, \quad N = \text{let}(x, \text{sample}, x = x).$$

Both are closed terms of type \mathcal{R} , “applying” the uniform distribution to the diagonal function $x \mapsto x = x$. However, M implements a call-by-name application, whose reduction duplicates the probabilistic primitive before sampling the distribution, while the evaluation of N first samples a real number \underline{r} and then duplicates it:

$$\begin{aligned} M &\rightarrow \text{sample} = \text{sample} \rightarrow \underline{r} = \underline{s} && \text{for any } r \text{ and } s, \\ N &\rightarrow \text{let}(x, \underline{r}, x = x) \rightarrow \underline{r} = \underline{r} && \text{for any } r. \end{aligned}$$

The distribution associated with M by Red^∞ is the Dirac δ_0 , because $\text{Red}^\infty(M, \underline{U}) = \text{Red}^3(M, \underline{U}) = \lambda(\{(r, r) \text{ s.t. } r \in [0, 1]\}) \times \delta_1(U) + \lambda(\{(r, s) \text{ s.t. } r \neq s, r, s \in [0, 1]\}) \times \delta_0(U) = \delta_0(U)$, the last equality is because the diagonal set $\{(r, r) \text{ s.t. } r \in [0, 1]\}$ has Lebesgue measure zero. This expresses that M evaluates to $\underline{0}$ (i.e. “false”) with probability 1, although there are an uncountable number of reduction paths reaching $\underline{1}$. On the contrary, the distribution associated with N is δ_1 : $\text{Red}^\infty(N, \underline{U}) = \text{Red}^3(M, \underline{U}) = \lambda([0, 1]) \times \delta_1(U) = \delta_1(U)$, expressing that N always evaluates to $\underline{1}$ (i.e. true).

REMARK 2 (SCORE). Some probabilistic programming languages have a primitive score (e.g. [Borgström et al. 2016; Staton et al. 2016]) or factor (e.g. [Goodman and Tenenbaum 2014]), allowing to express a probabilistic distribution from a density function. A map f is the probabilistic density function of a distribution μ with respect to another measure, say the Lebesgue measure λ , whenever $\mu(U) = \int_U f(x) \lambda(dx)$, for every measurable U . Intuitively, $f(x)$ gives a “score” expressing the likelihood of sampling the value x from μ .

In our setting, the primitive score_x would be a term like $\Gamma \vdash \text{score}_x(M) : \mathcal{R}$, with $\Gamma, x : \mathcal{R} \vdash M : \mathcal{R}$ defining f . The reduction of $\text{score}_x(M)$ outputs any numeral \underline{r} (a possible sample from the distribution μ), while the value $f(r)$ is used to multiply Red , like:

$$\text{Red}(\text{score}_x(M), U) = \int_{\mathbb{R}} \chi_U(r) f(r) \lambda(dr). \quad (7)$$

This primitive allows to implement a distribution in a more efficient way than rejection sampling, this latter based on a loop (Example 3.3). However, $\text{score}_x(M)$ suffers a major drawback: there is no static way of characterizing whether a term M is implementing a probabilistic density function or rather a generic measurable map. The integral in (7) can have a value greater than one or even to be infinite or undefined for general f , in particular Red would fail to be a stochastic kernel for all terms.

This problem can be overcome by modifying the output type of a program, see e.g. [Staton et al. 2016]. We decided however to avoid these issues, convinced that PPCF is already expressive enough to test the category \mathbf{Cstab}_m , which is the true object of study of this article.

4 CONES

We now study the central semantical concept of this paper: cones and stable functions between cones. Before entering into technicalities, let us provide some intuitions and motivations. A *complete cone* P is an \mathbb{R}^+ -semimodule together with a norm $\|\cdot\|_P$ satisfying some natural axioms (Definition 4.1) and such that the unit ball $\mathcal{B}P$ defined by the norm is complete with respect to the cone order \leq_P (Definition 4.2). A type A of PPCF will be associated with a cone $\llbracket A \rrbracket$ and a closed program of type A will be denoted as an element in the unit ball $\mathcal{B}\llbracket A \rrbracket$. The order completeness of $\mathcal{B}\llbracket A \rrbracket$ is crucial for defining the interpretation of the recursive programs (Section 7.1), as usual.

There are various notions of cone in the literature and we are following Selinger [2004], who uses cones similar to the ones already presented in e.g. [Andô 1962]. Let us stress two of its crucial features. (1) The cone order \leq_P is defined by the algebraic structure and not given independently from it — this is in accordance with what happens in the category of probabilistic coherence spaces [Danos and Ehrhard 2011]. (2) The completeness of $\mathcal{B}P$ is defined with respect to the cone order, in the spirit of domain theory, rather than with respect to the norm, as it is usual in the theory of Banach spaces.

A program taking inputs of type A and giving outputs of type B will be denoted as a map from $\mathcal{B}\llbracket A \rrbracket$ to $\mathcal{B}\llbracket B \rrbracket$. The goal of Section 4.1 is to find the right properties enjoyed by such functions in order to get a cartesian closed category, namely that the set of these functions generates a complete cone compatible with the cartesian structure (which will be the denotation of the type $A \rightarrow B$). It turns out that the usual notion of Scott continuity (Definition 4.10) is too weak a condition for ensuring cartesian closeness (Section 4.1.1). A precise analysis of this point led us to the conclusion that these functions have also to be *absolutely monotonic* (Definition 4.14). This latter condition is usually expressed by saying that all derivatives are everywhere non-negative, however we define it here as the non-negativity of iterated differences. Such non-differential definitions of absolute monotonicity have already been considered various times in classical analysis, see for instance [McMillan 1954].

We call *stable functions* the Scott continuous and absolutely monotonic functions (Definition 4.27), allowing for a cpo-enriched cartesian closed structure over the category of cones. The model of PPCF needs however a further notion, that of measurability, which will be discussed in Section 5.

Definition 4.1. A cone P is an \mathbb{R}^+ -semimodule given together with an \mathbb{R}^+ -valued function $\|_P$ such that the following conditions hold for all $x, x', y, y' \in P$ and $\alpha \in \mathbb{R}^+$

- $x + y = x + y' \Rightarrow y = y'$
- $\|\alpha x\|_P = \alpha \|x\|_P$
- $\|x + x'\|_P \leq \|x\|_P + \|x'\|_P$
- $\|x\|_P = 0 \Rightarrow x = 0$
- $\|x\|_P \leq \|x + x'\|_P$.

For $\alpha \in \mathbb{R}^+$ the set $\mathcal{BP}(\alpha) = \{x \in P \mid \|x\|_P \leq \alpha\}$ is called *the ball of P of radius α* . The *unit ball* is $BP = \mathcal{BP}(1)$. A subset S of P is *bounded* if $S \subseteq \mathcal{BP}(\alpha)$ for some $\alpha \in \mathbb{R}^+$.

Observe that $\|0\|_P = 0$ by the second condition (homogeneity of the norm) and that if $x + x' = 0$ then $x' = x = 0$ by the last condition (monotonicity of the norm).

Definition 4.2. Let $x, x' \in P$, one writes $x \leq_P x'$ if there is a $y \in P$ such that $x' = x + y$. This y is then unique, and we set $x' - x = y$. The relation \leq_P is easily seen to be an order relation on P and will be called *the cone order relation* of P .

A cone P is *complete* if any non-decreasing sequence $(x_n)_{n \in \mathbb{N}}$ of elements of \mathcal{BP} has a least upper bound $\sup_{n \in \mathbb{N}} x_n \in \mathcal{BP}$.

The usual laws of calculus using subtraction hold (under the restriction that all usages of subtraction must be well-defined). For instance, if $x, y, z \in P$ satisfy $z \leq_P y \leq_P x$ then we have $x - z = (x - y) + (y - z)$. Indeed, it suffices to observe that $(x - y) + (y - z) + z = x$.

There are many examples of cones.

Example 4.3. The prototypical example is \mathbb{R}^+ with the usual algebraic operations and the norm given by $\|x\|_{\mathbb{R}^+} = x$. The cone ℓ_∞ is defined by taking as carrier set the set of all bounded elements of $(\mathbb{R}^+)^{\mathbb{N}}$, defining the algebraic laws pointwise, and equipping it with the norm $\|u\| = \sup_{n \in \mathbb{N}} u_n$. The cone ℓ_1 instead is given by taking as carrier set the set of all elements u of $(\mathbb{R}^+)^{\mathbb{N}}$ such that $\sum_{n=0}^{\infty} u_n < \infty$, defining the algebraic laws pointwise, and equipping it with the norm $\|u\| = \sum_{n=0}^{\infty} u_n < \infty$.

Example 4.4. Let $\mathcal{X} = (|\mathcal{X}|, P\mathcal{X})$ be a probabilistic coherence space (see [Danos and Ehrhard 2011]). Remember that this means that $|\mathcal{X}|$ is a countable set (called web) and $P\mathcal{X} \subseteq (\mathbb{R}^+)^{|\mathcal{X}|}$ satisfies $P\mathcal{X} = P\mathcal{X}^{\perp\perp}$ (where, given $\mathcal{F} \subseteq (\mathbb{R}^+)^{|\mathcal{X}|}$, the set $\mathcal{F}^\perp \subseteq (\mathbb{R}^+)^{|\mathcal{X}|}$ is $\mathcal{F}^\perp = \{u' \in (\mathbb{R}^+)^{|\mathcal{X}|} \mid \forall u \in \mathcal{F} \sum_{a \in |\mathcal{X}|} u_a u'_a \leq 1\}$)⁸. Then we define a cone $\widehat{\mathcal{X}}$ by setting $\widehat{\mathcal{X}} = \{u \in (\mathbb{R}^+)^{|\mathcal{X}|} \mid \exists \varepsilon > 0 \ \varepsilon u \in P\mathcal{X}\}$, defining algebraic operations in the usual componentwise way and setting $\|u\|_{\widehat{\mathcal{X}}} = \inf\{\alpha > 0 \mid \frac{1}{\alpha} u \in P\mathcal{X}\} = \sup\{\sum_{a \in |\mathcal{X}|} u_a u'_a \mid u' \in P\mathcal{X}^\perp\}$.

The cones in Example 4.3 are instances of this one.

Example 4.5. The set of all $u \in (\mathbb{R}^+)^{\mathbb{N}}$ such that $u_n = 0$ for all but a finite number of indices n , is a cone P_0 when setting $\|u\|_{P_0} = \sum_{n \in \mathbb{N}} u_n$.

Example 4.6. Let X be a measurable space. The set of all \mathbb{R}^+ -valued measures⁹ on X is a cone $\text{Meas}(X)$, algebraic operations being defined in the usual “pointwise” way (e.g. $(\mu + \nu)(U) = \mu(U) + \nu(U)$) and norm given by $\|\mu\|_{\text{Meas}(X)} = \mu(X)$. This is the main motivating example for the present paper. Observe that such a cone is not of the shape $\widehat{\mathcal{X}}$ in general.

In all these examples, the cone order can be described in a pointwise way. For instance, when \mathcal{X} is a probabilistic coherence space, one has $u \leq_{\widehat{\mathcal{X}}} v$ iff $\forall a \in |\mathcal{X}| \ u_a \leq v_a$. Similarly when X is a

⁸There are actually two additional conditions which are not essential here.

⁹So we consider only “bounded” measures, which satisfy that the measure of the total space is finite, which is not the case of the Lebesgue measure on the whole \mathbb{R} .

measurable space, one has $\mu \leq_{\text{Meas}(X)} \nu$ iff $\forall U \in \Sigma_X \mu(U) \leq \nu(U)$. This is due to the fact that when this condition holds, the function $U \mapsto \nu(U) - \mu(U)$ is easily seen to be an \mathbb{R}^+ -valued measure.

All the examples above, but Example 4.5, are examples of complete cones.

LEMMA 4.7. *P is complete iff any bounded non-decreasing sequence $(x_n)_{n \in \mathbb{N}}$ has a least upper bound $\sup_{n \in \mathbb{N}} x_n$ which satisfies $\|\sup_{n \in \mathbb{N}} x_n\|_P = \sup_{n \in \mathbb{N}} \|x_n\|_P$.*

Definition 4.8. Let P and Q be cones. A *bounded map* from P to Q is a function $f : \mathcal{B}P \rightarrow Q$ such that $f(\mathcal{B}P) \subseteq \mathcal{B}Q(\alpha)$ for some $\alpha \in \mathbb{R}^+$; the greatest lower bound of these α 's is called *the norm of f* and is denoted as $\|f\|$.

LEMMA 4.9. *Let f be a bounded map from P to Q , then $\|f\| = \sup_{x \in \mathcal{B}P} \|f(x)\|_Q$ and $f(\mathcal{B}P) \subseteq \mathcal{B}Q(\|f\|)$.*

Definition 4.10. A function $f : P \rightarrow Q$ is *linear* if it commutes with sums and scalar multiplication. A *Scott-continuous function* from a complete cone P to a complete cone Q is a bounded map¹⁰ from P to Q which is non-decreasing and commutes with the lubs of non-decreasing sequences.

LEMMA 4.11. *Let P be a complete cone. Addition is Scott-continuous $P \times P \rightarrow P$ and scalar multiplication is Scott-continuous $\mathbb{R}^+ \times P \rightarrow P$.*

Proofs are easy, see [Selinger 2004]. The cartesian product of cones is defined in the obvious way (see Figure 4a).

Definition 4.12. Let P be a cone and let $u \in \mathcal{B}P$. We define a new cone P_u (the *local cone of P at u*) as follows. We set $P_u = \{x \in P \mid \exists \varepsilon > 0 \ \varepsilon x + u \in \mathcal{B}P\}$ and

$$\|x\|_{P_u} = \inf\{1/\varepsilon \mid \varepsilon > 0 \text{ and } \varepsilon x + u \in \mathcal{B}P\} = (\sup\{\varepsilon \mid \varepsilon > 0 \text{ and } \varepsilon x + u \in \mathcal{B}P\})^{-1}.$$

Given a sequence $\vec{u} = (u_1, \dots, u_n)$ of elements of a cone P s.t. $u = \sum_{i=1}^n u_i \in \mathcal{B}P$, we set $P_{\vec{u}} = P_u$.

LEMMA 4.13. *For any cone P and any $u \in \mathcal{B}P$, P_u is a cone. Moreover $\mathcal{B}P_u = \{x \in P \mid x + u \in \mathcal{B}P\}$ and, for any $x \in P_u$, one has $\|x\|_P \leq \|x\|_{P_u}$. If P is complete then P_u is complete.*

4.1 Pre-stable, aka. Absolutely Monotonic, Functions

We want now to introduce a notion of morphisms between cones such that the resulting category will be cartesian closed. Given two cones P and Q , a morphism from P to Q will be a Scott-continuous function from $\mathcal{B}P$ to Q (because we need our morphisms to have least fix-points in order to interpret general recursion) such that $f(\mathcal{B}P) \subseteq \mathcal{B}Q$.

4.1.1 *Failure of the Straightforward Attempt.* Is this Scott-continuity condition sufficient for guaranteeing cartesian closeness? We argue that this not the case. Assume the opposite. Then it is easy to check that the cartesian product in our category is defined in the obvious way (algebraic operations defined pointwise, and supremum norm).

Given two cones P and Q , we will need to define a new cone $R = (P \Rightarrow Q)$ such that $\mathcal{B}R$ will coincide with the set of morphisms from P to Q that is, under our assumption, of all Scott-continuous functions $\mathcal{B}P \rightarrow \mathcal{B}Q$. In this cone R (whose elements are all the Scott-continuous functions $f : \mathcal{B}P \rightarrow Q$), the algebraic operations are defined pointwise¹¹ and so the addition of R induces the following order relation on Scott-continuous functions: $f \leq g$ if $\forall x \in \mathcal{B}P \ f(x) \leq g(x)$ and, moreover, the function $x \mapsto g(x) - f(x)$ is Scott-continuous.

¹⁰Remember that then $f : \mathcal{B}P \rightarrow Q$, according with Definition 4.8.

¹¹Because the evaluation function should be linear in its functional argument, in accordance with the call-by-name evaluation strategy of our target programming language, see Lemma 7.6.

Consider now the function $\text{wpor} : [0, 1] \times [0, 1] \rightarrow [0, 1]$ defined by $\text{wpor}(s, t) = s + t - st = (1 - s)t + s = (1 - t)s + t$ and considered for instance in [Escardó et al. 2004]. It is clearly a Scott-continuous function, so it is a morphism $\mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ in our category of cones and Scott-continuous functions. Let $\text{wpor}' : [0, 1] \rightarrow \mathcal{B}(\mathbb{R}^+ \Rightarrow \mathbb{R}^+)$ be the curried version of wpor defined by $\text{wpor}'(s) = \text{wpor}_s$ where wpor_s is the Scott-continuous function $[0, 1] \rightarrow [0, 1]$ defined by $\text{wpor}_s(t) = \text{wpor}(s, t)$. Then, by cartesian closeness, wpor' should be Scott-continuous and hence non-decreasing. But $\text{wpor}_0(t) = t$ and $\text{wpor}_1(t) = 1$ and the function $t \mapsto \text{wpor}_1(t) - \text{wpor}_0(t) = 1 - t$ is not non-decreasing. So we do not have $\text{wpor}'(0) \leq \text{wpor}'(1)$ in the cone $\mathbb{R}^+ \Rightarrow \mathbb{R}^+$ and our category is not cartesian closed.

Our methodological principle is to stick to the cone order, natural wrt. the algebraic structure, and adapt the notion of morphism so as to obtain a cartesian closed category.

It turns out that this is perfectly possible and leads to an interesting new notion of morphisms between cones, deeply related with stability [Berry 1978] and the category $\mathbf{PCoh}_!$ of probabilistic coherence spaces already mentioned in the Introduction, as we will show in a further paper. This connection with stability is already suggested by the wpor example: this function is a “probabilistic version” of the well known parallel-or function [Plotkin 1977]: we have $\text{wpor}(1, 0) = \text{wpor}(0, 1) = 1$ and $\text{wpor}(0, 0) = 0$. Stability has been introduced for rejecting such functions.

4.1.2 Absolutely Monotonic Functions. Pushing further the above analysis of the constraints imposed by cartesian closeness on the monotonicity of morphisms, one arrives naturally to the following definition. Given a function $f : \mathcal{BP} \rightarrow Q$ which is non-decreasing, we use the notation

$$\Delta f(x; u) = f(x + u) - f(x)$$

for all $x \in \mathcal{BP}$ and $u \in P$ such that $x + u \in \mathcal{BP}$. It is clear that $\Delta f(x; u) \in \mathcal{B}Q$.

Definition 4.14. An n -non-decreasing function from P to Q is a function $f : \mathcal{BP} \rightarrow Q$ such that

- $n = 0$ and f is non-decreasing
- or $n > 0$, f is non-decreasing and, for all $u \in \mathcal{BP}$, the function $\Delta f(_, u)$ is $n-1$ -non-decreasing from P_u to Q .

One says that f is *absolutely monotonic* if it is n -non-decreasing for all $n \in \mathbb{N}$.

Example 4.15. Take $P = Q = \mathbb{R}^+$. A function $f : \mathcal{BP} = [0, 1] \rightarrow Q = \mathbb{R}^+$ is 0-non-decreasing if it is non-decreasing. It is 1-non-decreasing if, moreover, for all $u \in [0, 1]$, the function $\Delta f(_, u) : [0, 1 - u] \rightarrow \mathbb{R}^+$ defined by $\Delta f(x; u) = f(x + u) - f(x)$ is non-decreasing. It is 2-non-decreasing if moreover, for all $u_1, u_2 \in [0, 1]$ such that $u_1 + u_2 \in [0, 1]$, the function $\Delta f(_, u_1, u_2) : [0, 1 - u_1 - u_2] \rightarrow \mathbb{R}^+$ defined by $\Delta f(x; u_1, u_2) = \Delta f(x + u_2; u_1) - \Delta f(x; u_1) = f(x + u_2 + u_1) - f(x + u_2) - (f(x + u_1) - f(x)) = f(x + u_1 + u_2) - f(x + u_1) - f(x + u_2) + f(x)$ is non-decreasing, etc. Typical examples of such n -non-decreasing functions for all n are the polynomial functions with non-negative coefficients.

Example 4.16. To illustrate this definition further, consider the wpor function introduced in Section 4.1.1. It is clearly 0-non-decreasing. For $s, t, u, v \in \mathbb{R}^+$ such that $s + u, t + v \in [0, 1]$ we have $\Delta \text{wpor}((s, t); (u, v)) = (s + u) + (t + v) - (s + u)(t + v) - (s + t - st) = u + v - sv - tu + st = (1 - t)u + (1 - s)v + st$. This function is not non-decreasing in s and t , so wpor is not 1-non-decreasing.

4.1.3 Pre-stable Functions. In most cases, Definition 4.14 is hard to manipulate because it is inductive and uses explicitly subtraction, an operation which is only partially defined. We thus give an equivalent notion as follows.

Let $n \in \mathbb{N}$, we use $\mathcal{P}_+(n)$ (resp. $\mathcal{P}_-(n)$) for the set of all subsets I of $\{1, \dots, n\}$ such that $n - \#I$ is even (resp odd). Given a map $f : \mathcal{BP} \rightarrow Q$, $\vec{u} \in P^n$ such that $\sum_{i=1}^n u_i \in \mathcal{BP}$ and $x \in \mathcal{BP}_{\vec{u}}$, we define

$$\Delta^\varepsilon f(x; \vec{u}) = \sum_{I \in \mathcal{P}_\varepsilon(n)} f(x + \sum_{i \in I} u_i) \in Q$$

for $\varepsilon \in \{+, -\}$. For instance $\Delta^- f(x; u_1, u_2, u_3) = f(x + u_1 + u_2) + f(x + u_2 + u_3) + f(x + u_1 + u_3) + f(x)$ and $\Delta^+ f(x; u_1, u_2, u_3) = f(x + u_1 + u_2 + u_3) + f(x + u_1) + f(x + u_2) + f(x + u_3)$.

Observe that when $n = 0$, we have $\Delta^+ f(x;) = f(x)$ and $\Delta^- f(x;) = 0$.

Definition 4.17. An n -pre-stable function from P to Q is a function $f : \mathcal{BP} \rightarrow Q$ such that, for all $k \in \{1, \dots, n+1\}$, all $\vec{u} \in P^k$ such that $\sum_{i=1}^n u_i \in \mathcal{BP}$, and all $x \in \mathcal{BP}_{\vec{u}}$, one has $\Delta^- f(x; \vec{u}) \leq \Delta^+ f(x; \vec{u})$. One says that f is pre-stable if it is n -pre-stable for all n .

Observe f is 0-pre-stable iff for all $x \in \mathcal{BP}$ and all $u \in P$ such that $x + u \in \mathcal{BP}$, one has $f(x) \leq f(x + u)$, that is, f is non-decreasing.

By generalizing the computation in Example 4.15, one can prove by induction the following.

THEOREM 4.18. For all $n \in \mathbb{N}$, a function $f : \mathcal{BP} \rightarrow Q$ is n -non-decreasing iff it is n -pre-stable. Therefore, f is absolutely monotonic iff it is pre-stable.

LEMMA 4.19. Let f be an absolutely monotonic function from P to Q (so that $f : \mathcal{BP} \rightarrow Q$). Let $n \in \mathbb{N}$, $\vec{u} \in \mathcal{BP}^n$ with $\sum_{i=1}^n u_i \in \mathcal{BP}$ and $x \in \mathcal{BP}_{\vec{u}}$. Let f_0, \dots, f_n be the functions defined by $f_0(x) = f(x)$ and $f_{i+1}(x) = \Delta f_i(x; u_{i+1})$. Then

$$f_n(x) = \Delta^+ f(x; \vec{u}) - \Delta^- f(x; \vec{u}).$$

We set $\Delta f(x; \vec{u}) = f_n(x)$. The operation Δ is linear in the function: $\Delta(\sum_{j=1}^p \alpha_j g_j)(x; \vec{u}) = \sum_{j=1}^p \alpha_j \Delta g_j(x; \vec{u})$ for g_1, \dots, g_p absolutely monotonic from P to Q .

As an immediate consequence we have that $\Delta f(x; u_1, \dots, u_n)$ is symmetric in u_1, \dots, u_n , that is: $\Delta f(x; u_1, \dots, u_n) = \Delta f(x; u_{\sigma(1)}, \dots, u_{\sigma(n)})$ for all permutation σ on $\{1, \dots, n\}$.

4.2 Composing Pre-stable Functions

It is not completely obvious that pre-stable functions are closed under composition (Theorem 4.26). The situation is a bit similar in categories of smooth functions where composability derives from the chain rule. Theorem 4.26 is an immediate consequence of Lemma 4.25, the proof of this latter needing the following auxiliary lemmas.

LEMMA 4.20. Let $f : \mathcal{BP} \rightarrow Q$ be a pre-stable function from P to Q . For all $\vec{u} \in \mathcal{BP}^n$, the functions $\Delta^- f(_, \vec{u})$, $\Delta^+ f(_, \vec{u})$ and $\Delta f(_, \vec{u})$ are pre-stable from $P_{\vec{u}}$ to Q .

LEMMA 4.21. Let $f : \mathcal{BP} \rightarrow Q$ be a pre-stable function from P to Q . Let $n \in \mathbb{N}$, $x, u, v \in \mathcal{BP}$ and $\vec{u} \in \mathcal{BP}^n$, and assume that $x + u + v + \sum_{i=1}^n u_i \in \mathcal{BP}$. Then

$$\Delta f(x + u; \vec{u}) = \Delta f(x; \vec{u}) + \Delta f(x; u, \vec{u}), \quad \Delta f(x; u + v, \vec{u}) = \Delta f(x; u, \vec{u}) + \Delta f(x + u; v, \vec{u}).$$

LEMMA 4.22. Let $f : \mathcal{BP} \rightarrow Q$ be a function which is pre-stable from P to Q . Let $n \in \mathbb{N}$, $x, u \in \mathcal{BP}$ and $\vec{u}, \vec{v} \in \mathcal{BP}^n$, and assume that $x + u + \sum_{i=1}^n (u_i + v_i) \in \mathcal{BP}$. Then

$$\begin{aligned} \Delta f(x + u; \vec{u} + \vec{v}) &= \Delta f(x; \vec{u}) + \Delta f(x; u, \vec{u} + \vec{v}) + \Delta f(x + u_1; v_1, u_2 + v_2, \dots, u_n + v_n) \\ &\quad + \Delta f(x + u_2; u_1, v_2, u_3 + v_3, \dots, u_n + v_n) + \dots + \Delta f(x + u_n; u_1, \dots, u_{n-1}, v_n). \end{aligned}$$

PROOF. Simple computation using Lemma 4.21. □

Let P be a cone and let $p \in \mathbb{N}$. Let $S^p(P) = (P^{p+1}, \|\cdot\|_{S^p(P)})$ where $\|(x, u_1, \dots, u_p)\|_{S^p(P)} = \|x + \sum_{i=1}^p u_i\|_P$. Then, with algebraic laws defined componentwise, it is easy to check that $S^p(P)$ is a cone which is complete if P is.

LEMMA 4.23. *Let $f : \mathcal{BP} \rightarrow Q$ be a pre-stable function from P to Q . Then the map $g : \mathcal{BS}^p(P) \rightarrow Q$ defined by $g(x, \vec{u}) = \Delta f(x; \vec{u})$ is non-decreasing, for all $p \in \mathbb{N}^+$.*

LEMMA 4.24. *Let $f, g : \mathcal{BP} \rightarrow Q$ be two pre-stable functions from P to Q . The function $f + g$ (sum defined pointwise) is pre-stable.*

LEMMA 4.25. *Let $p \in \mathbb{N}$, $f, h_1, \dots, h_p : \mathcal{BP} \rightarrow Q$ be pre-stable functions from P to Q and $g : \mathcal{BQ} \rightarrow R$ be pre-stable functions from Q to R such that $\forall x \in \mathcal{BP} \ f(x) + \sum_{i=1}^p h_i(x) \in \mathcal{BQ}$. Then the function $k : \mathcal{BP} \rightarrow R$ defined by $k(x) = \Delta g(f(x); h_1(x), \dots, h_p(x))$ is pre-stable from P to R .*

PROOF. Observe that our hypotheses imply that, for all $x \in \mathcal{BP}$, one has $f(x) \in \mathcal{BQ}_{h_1(x), \dots, h_n(x)}$

With the notations and conventions of the statement, we prove by induction on n that, for all $n \in \mathbb{N}$, for all $p \in \mathbb{N}$, for all f, h_1, \dots, h_p, g which are pre-stable and satisfy $\forall x \in \mathcal{BP} \ f(x) + \sum_{i=1}^p h_i(x) \in \mathcal{BQ}$, the function k is n -pre-stable.

For $n = 0$, the property results from Lemma 4.23.

We assume the property for n and prove it for $n + 1$. Let $u \in \mathcal{BP}$ we have to prove that the function $\Delta k(\cdot; u)$ is n -pre-stable from P_u to R . Let $x \in \mathcal{BP}_u$, we have

$$\begin{aligned} \Delta k(x; u) &= \Delta g(f(x+u); h_1(x+u), \dots, h_p(x+u)) - \Delta g(f(x); h_1(x), \dots, h_p(x)) \\ &= \Delta g(f(x) + \Delta f(x; u); h_1(x) + \Delta h_1(x; u), \dots, h_p(x) + \Delta h_1(x; u)) \\ &\quad - \Delta g(f(x); h_1(x), \dots, h_p(x)) \\ &= \Delta g(f(x); \Delta f(x; u), h_1(x) + \Delta h_1(x; u), \dots, h_p(x) + \Delta h_p(x; u)) \\ &\quad + \Delta g(f(x) + h_1(x); \Delta h_1(x; u), h_2(x) + \Delta h_2(x; u), \dots, h_p(x) + \Delta h_p(x; u)) \\ &\quad + \Delta g(f(x) + h_2(x); h_1(x), \Delta h_2(x; u), h_3(x) + \Delta h_3(x; u), \dots, h_p(x) + \Delta h_p(x; u)) + \dots \\ &\quad + \Delta g(f(x) + h_p(x); h_1(x), \dots, h_{p-1}(x), \Delta h_p(x; u)) \end{aligned}$$

by Lemma 4.22. We can apply the inductive hypothesis to each of the terms of this sum. Let us consider for instance the first of these expressions. We know that the functions h'_1, \dots, h'_{p+1} defined by $h'_1(x) = \Delta f(x; u)$, $h'_2(x) = h_1(x) + \Delta h_1(x; u) = h_1(x+u), \dots, h'_{p+1}(x) = h_p(x) + \Delta h_p(x; u) = h_p(x+u)$ are pre-stable from P_u to Q : this results from Lemmas 4.24 and 4.20. Moreover we have $\forall x \in \mathcal{BP} \ f(x) + \sum_{i=1}^{p+1} h'_i(x) = f(x+u) + \sum_{i=1}^p h_i(x+u) \in \mathcal{BQ}$. Therefore the inductive hypothesis applies and we know that the function $x \mapsto \Delta g(f(x); \Delta f(x; u), h_1(x) + \Delta h_1(x; u), \dots, h_p(x) + \Delta h_p(x; u))$ is n -pre-stable. The same reasoning applies to all terms and, by Lemma 4.24, we know that the function $\Delta k(\cdot; u)$ is n -pre-stable from P_u to Q . \square

THEOREM 4.26. *Let f be a pre-stable function from P to Q and g be a pre-stable function from Q to R . If $f(\mathcal{BP}) \subseteq \mathcal{BQ}$ then $g \circ f$ is a pre-stable function from P to R .*

PROOF. This is the case $p = 0$ of Lemma 4.25. \square

Definition 4.27. A stable function from P to Q is a pre-stable (or, equivalently, an absolutely monotonic) function from P to Q which is Scott continuous. We use \mathbf{Cstab} for the category of complete cones and stable functions. More explicitly, $\mathbf{Cstab}(P, Q)$ is the set of all functions $f : \mathcal{BP} \rightarrow Q$ which are pre-stable from P to Q , Scott-continuous and satisfy $f(\mathcal{BP}) \subseteq \mathcal{BQ}$.

5 MEASURABILITY

The cone $\text{Meas}(\mathbb{R})$ of \mathbb{R}^+ -valued measures on \mathbb{R} (Example 4.6) is the natural candidate to model the ground type \mathcal{R} . In particular, a real numeral will be interpreted as the Dirac measure δ_r . Consider now a closed term $\text{let}(x, M, N)$ of type \mathcal{R} , so that $\vdash M : \mathcal{R}$ and $x : \mathcal{R} \vdash N : \mathcal{R}$. The term M will be associated with a measure μ in $\mathcal{B}(\text{Meas}(\mathbb{R}))$, while N will be a stable function f from the whole $\mathcal{B}(\text{Meas}(\mathbb{R}))$ to $\mathcal{B}(\text{Meas}(\mathbb{R}))$. However, according to the operational semantics (Figure 3), N is supposed to get a real number r for x , and not a generic measure. This means that one has to compose f with a map $\delta : \mathbb{R} \rightarrow \mathcal{B}(\text{Meas}(\mathbb{R}^+))$ defined by $\delta(r) = \delta_r$, so that $f \circ \delta : \mathbb{R} \rightarrow \mathcal{B}(\text{Meas}(\mathbb{R}))$. Now, the natural way to pass μ to $f \circ \delta$ is then by the integral $\int_{\mathbb{R}} (f \circ \delta)(r) \mu(dr)$. However, this would be meaningful only in case $f \circ \delta$ is measurable, and this is not the true of all stable functions f .¹²

We have then to slightly refine our model, endowing our cones with a structure allowing to formulate a convenient measurability property for our morphisms. This is the goal of this section.

5.1 Measurability Tests

If P is a cone, we use P' for the topological dual of P , which is the set of all functions $l : P \rightarrow \mathbb{R}^+$ which are linear (that is, commute with linear combinations) and Scott-continuous. Such a function, when restricted to $\mathcal{B}P$, clearly defines a stable function from P to \mathbb{R}^+ .

Definition 5.1. We consider cones P equipped with a collection $(M^n(P))_{n \in \mathbb{N}}$ where $M^n(P) \subseteq (P')^{\mathbb{R}^n}$ satisfies the following properties.

- $0 \in M^n(P)$
- if $l \in M^n(P)$ and $h : \mathbb{R}^P \rightarrow \mathbb{R}^n$ is measurable then $l \circ h \in M^P(P)$
- and for any $l \in M^n(P)$ any $x \in P$, the function $\mathbb{R}^n \rightarrow \mathbb{R}^+$ which maps \vec{r} to $l(\vec{r})(x)$ is in M^n , i.e. is a measurable map $\mathbb{R}^n \rightarrow \mathbb{R}^+$.

A cone P equipped with a family $(M^n(P))_{n \in \mathbb{N}}$ satisfying the above conditions will be called a *measurable cone*. The elements of the sets $M^n(P)$ will be called the *measurability tests* of P .

Measurability tests have parameters in \mathbb{R}^n and are not simply Scott-continuous linear forms for making it possible to prove that the evaluation function of the cartesian closed structure is well behaved. This will be explained in Remark 3.

Definition 5.2. Let P be a cone and let $n \in \mathbb{N}$. A *measurable path* of arity n in P is a function $\gamma : \mathbb{R}^n \rightarrow P$ such that

- $\gamma(\mathbb{R}^n)$ is bounded in P
- and, for all $k \in \mathbb{N}$ and all $l \in M^k(P)$, the function $l * \gamma : \mathbb{R}^{k+n} \rightarrow \mathbb{R}^+$ defined by $(l * \gamma)(\vec{r}, \vec{s}) = l(\vec{r})(\gamma(\vec{s}))$ is in M^{k+n} , i.e. is a measurable map $\mathbb{R}^{k+n} \rightarrow \mathbb{R}^+$.

We use $\text{Path}^n(P)$ for the set of measurable paths of P and $\text{Path}_1^n(P)$ for the set of measurable paths which take their values in $\mathcal{B}P$.

LEMMA 5.3. *For any $x \in P$ and $n \in \mathbb{N}$, the function $\gamma : \mathbb{R}^n \rightarrow P$ defined by $\gamma(\vec{r}) = x$ belongs to $\text{Path}^n(P)$. If $\gamma \in \text{Path}^n(P)$ and $h : \mathbb{R}^P \rightarrow \mathbb{R}^n$ is measurable then $\gamma \circ h \in \text{Path}^P(P)$.*

¹²Indeed, by Lebesgue decomposition theorem we can write $\text{Meas}(\mathbb{R}) = \mathcal{M}_0 \oplus \mathcal{M}_1$, a co-product of cones (it is easily checked that the category of complete cones of linear and Scott-continuous functions has co-products), where the elements of \mathcal{M}_0 are the discrete measures, that is, the countable linear combinations of Dirac measures $\sum_{i=1}^{\infty} \alpha_n \delta_{r_n}$ with $\forall n \alpha_n \in \mathbb{R}^+$ and $\sum_n \alpha_n < \infty$, and \mathcal{M}_1 is the cone of measures μ such that $\mu(\{r\}) = 0$ for all $r \in \mathbb{R}$. Let $U \subseteq \mathbb{R}$ be a non-measurable set and let $f : \mathcal{M} \rightarrow \mathbb{R}^+$ be the linear (and hence pre-stable) and Scott-continuous function defined on this co-product, by: $f(\mu) = 0$ if $\mu \in \mathcal{M}_1$ and $f(\delta_r) = \chi_U(r)$. Then $f \circ \delta = \chi_U$ is not measurable. We thank Jean-Louis Krivine for this example.

$$\begin{aligned}
\prod_{i \in I} P_i &= \{(x_i)_{i \in I} \text{ s.t. } \forall i \in I, x_i \in P_i\}, & \|(x_i)_{i \in I}\|_{\prod_{i \in I} P_i} &= \sup_{i \in I} \|x_i\|_{P_i} \\
M^n(\prod_{i \in I} P_i) &= \{\bigoplus_{i \in I} l_i \mid \forall i \in I, l_i \in M^n(P_i)\}, & \text{with } \bigoplus_{i \in I} l_i(\vec{r})(x_i)_{n \in I} &= \sum_{i \in I} l_i(\vec{r})(x_i) \\
(a) \text{ finite cartesian product } (I \text{ finite set}). & \text{We can simply write } P_1 \times P_2 \text{ for the binary product.} \\
P \Rightarrow_m Q &= \{f : \mathcal{B}P \rightarrow Q \mid \exists \varepsilon > 0, \varepsilon f \in \mathbf{Cstab}_m(P, Q)\}, & \|f\|_{P \Rightarrow_m Q} &= \sup_{x \in \mathcal{B}P} \|f(x)\|_Q \\
M^n(P \Rightarrow_m Q) &= \{\gamma \triangleright m \mid \gamma \in \text{Path}^n(P), m \in M^n(Q)\}, & \text{with } \gamma \triangleright m(\vec{r})(f) &= m(\vec{r})(f(\gamma(\vec{r}))) \\
(b) \text{ object of morphisms}
\end{aligned}$$

Fig. 4. The CCC structure of \mathbf{Cstab}_m . The projections, pairing and the evaluation are defined as standard.

Example 5.4. Let X be a measurable space. We equip the cone $\text{Meas}(X)$ with the following notion of measurability tests. For each $n \in \mathbb{N}$, we set $M^n(\text{Meas}(X)) = \{\varepsilon_U \mid U \in \Sigma_X\}$ where $\varepsilon_U(\vec{r})(\mu) = \mu(U)$. Observe that indeed ε_U is linear and Scott-continuous, see Example 4.6 and the observation that, in the complete cone $\text{Meas}(X)$, lubs are computed pointwise: given a non-decreasing and bounded sequence μ_n of elements of $\text{Meas}(X)$, one has $(\sup_{n \in \mathbb{N}} \mu_n)(U) = \sup_{n \in \mathbb{N}} \mu_n(U)$. Therefore an element of $\text{Path}_1^n(\text{Meas}(X))$ is a stochastic kernel from \mathbb{R}^n to X . This example justifies our terminology of “measurable cone” because, in $\text{Meas}(X)$, the measurable tests coincide with the measurable sets of X .

Definition 5.5. Let P and Q be measurable complete cones. A stable function from P to Q (remember that then f is actually a function $\mathcal{B}P \rightarrow Q$) is *measurable* if, for all $n \in \mathbb{N}$ and all $\gamma \in \text{Path}_1^n(P)$, one has $f \circ \gamma \in \text{Path}^n(Q)$. We use \mathbf{Cstab}_m for the subcategory of \mathbf{Cstab} whose morphisms are measurable.

This definition makes sense because if $f : \mathcal{B}P \rightarrow \mathcal{B}Q$ and $g : \mathcal{B}Q \rightarrow \mathcal{B}R$ are stable and measurable, then $g \circ f$ has the same properties.

6 THE CARTESIAN CLOSED STRUCTURE OF \mathbf{CSTAB}_M

6.1 Cartesian Product

The *cartesian product* $P = \prod_{i \in I} (P_i)$ of a finite¹³ family of cones $(P_i)_{i \in I}$ is given in Figure 4a, where addition and scalar multiplication are defined componentwise. It is clear that we have defined in that way a complete cone and that $\mathcal{B}P = \prod_{i \in I} \mathcal{B}P_i$. Given a non-decreasing sequence $(x(p))_{p \in \mathbb{N}}$ in $\mathcal{B}P$, then $x = \sup_{p \in \mathbb{N}} x(p)$ is characterized by $x_i = \sup_{p \in \mathbb{N}} x(p)_i$ (this lub being taken in $\mathcal{B}P_i$).

The projections $\text{pr}_i : \mathcal{B}P \rightarrow \mathcal{B}P_i$ are easily seen to be linear and Scott-continuous and hence stable $P \rightarrow P_i$. Let $f_i \in \mathbf{Cstab}_m(Q, P_i)$ for each $i \in I$. We define a function $f : \mathcal{B}Q \rightarrow \mathcal{B}P$ by $f(y) = (f_i(y))_{i \in I}$. It is straightforward to check that this function is stable: pre-stability follows from $\Delta f(y; v) = (\Delta f_i(y; v))_{i \in I}$.

LEMMA 6.1. *Let $f : P \times \mathcal{B}Q \rightarrow R$ be a function such that*

- *for each $y \in \mathcal{B}Q$, the function $f_y^{(1)} : P \rightarrow R$ defined by $f_y^{(1)}(x) = f(x, y)$ is linear (resp. linear and Scott-continuous)*

¹³We could easily define countable products, this will be done in an extended version of this paper.

- and for each $x \in P$, the function $f_x^{(2)} : \mathcal{B}Q \rightarrow R$ defined by $f_x^{(2)}(y) = f(x, y)$ is pre-stable (resp. pre-stable and Scott-continuous).

Then the restriction $f : \mathcal{B}P \times \mathcal{B}Q \rightarrow R$ is pre-stable (resp. pre-stable and Scott-continuous, that is, stable) from $P \times Q$ to R .

In the second line of Figure 4a we endow the cone $\prod_{i \in I} P_i$ with a notion of measurability tests $M^n(\prod_{i \in I} P_i)$ for any $n \in \mathbb{N}$. It is obvious that this notion satisfies the conditions of Definition 5.1. The fact that $(\bigoplus_{i \in I} l_i)(\vec{r})$ is Scott-continuous results from the Scott-continuity of addition (Lemma 4.11). Moreover, for $x \in \prod_{i \in I} P_i$, the map $\vec{r} \mapsto \sum_{i \in I} l_i(\vec{r})(x_i)$ is measurable as a sum of measurable functions. Given $h : \mathbb{R}^p \rightarrow \mathbb{R}^n$ measurable, we have $(\bigoplus_{i \in I} l_i) \circ h = \bigoplus_{i \in I} (l_i \circ h)$ and hence the measurability tests of $\prod_{i \in I} P_i$ are closed under precomposition with measurable maps.

LEMMA 6.2. For any $n \in \mathbb{N}$ we have $\text{Path}^n(\prod_{i \in I} P_i) = \{\langle \gamma_i \rangle_{i \in I} \mid \forall i \in I \gamma_i \in \text{Path}^n(P_i)\}$.

THEOREM 6.3. The category \mathbf{Cstab}_m is cartesian with projections $(\text{pr}_i)_{i \in I}$ and tupling defined as in the category of sets and functions. The terminal object is the cone $\{0\}$ with measurability tests equal to 0.

It suffices to prove that the usual projections are measurable and that the tupling of measurable stable functions is measurable, this is straightforward.

6.2 Function Space

Let P and Q be two measurable complete cones. We define $P \Rightarrow_m Q$ in Figure 4b as the set of all measurable stable functions from P to Q , that is, the set of all functions $f : \mathcal{B}P \rightarrow Q$ such that there exists $\varepsilon > 0$ such that $\varepsilon f \in \mathbf{Cstab}_m(P, Q)$. It is clear that $P \Rightarrow_m Q$ is closed under pointwise addition of functions and pointwise scalar multiplication: this results from the fact that measurability tests are (parameterized) linear functions and that measurable functions $\mathbb{R}^n \rightarrow \mathbb{R}^+$ have the same closure properties. We still have to check that, with the norm $\| \cdot \|_{P \Rightarrow_m Q}$, $P \Rightarrow_m Q$ is a complete cone (Lemma 6.5).

For this purpose, the next lemma which provides a characterization of the order relation in the function space similar to Berry's stable order [Berry 1978] will be essential.

LEMMA 6.4. Let $f, g \in P \Rightarrow_m Q$. We have $f \leq g$ in $P \Rightarrow_m Q$ iff the following condition holds

$$\forall n \in \mathbb{N}, \forall \vec{u} \in P^n \sum_{i=1}^n u_i \in \mathcal{B}P, \forall x \in \mathcal{B}P_{\vec{u}}, \Delta^+ f(x; \vec{u}) + \Delta^- g(x; \vec{u}) \leq \Delta^+ g(x; \vec{u}) + \Delta^- f(x; \vec{u}).$$

LEMMA 6.5. The cone $P \Rightarrow_m Q$ is complete and the lubs in $\mathcal{B}(P \Rightarrow_m Q)$ are computed pointwise.

The second line of Figure 4b defines a family $(M^n(P \Rightarrow_m Q))_{n \in \mathbb{N}}$ of sets of measurability tests. Also in this case the conditions of Definition 5.1 are respected. The fact that indeed $(\gamma \triangleright m)(\vec{r}) \in (P \Rightarrow_m Q)'$ clearly follows from the definition of the cone $P \Rightarrow_m Q$ and from the fact that lubs in that cone are computed pointwise. Since $\text{Path}_1^n(P)$ is non-empty (it contains at least the 0-valued constant path ζ) and $0 \in M^n(Q)$, we have $0 = \zeta \triangleright 0 \in M^n(P \Rightarrow_m Q)$. Let $\gamma \in \text{Path}_1^n(P)$, $m \in M^n(Q)$ and let $h : \mathbb{R}^p \rightarrow \mathbb{R}^n$ be measurable. We have $(\gamma \triangleright m) \circ h = (\gamma \circ h) \triangleright (m \circ h)$ and we know that $\gamma \circ h \in \text{Path}_1^p(P)$ (by Lemma 5.3) and $m \circ h \in M^n(Q)$ (by assumption about Q) so $(\gamma \triangleright m) \circ h \in M^p(P \Rightarrow_m Q)$. Last, with the same notations, let $f \in P \Rightarrow_m Q$. Then the map $\vec{r} \mapsto (\gamma \triangleright m)(\vec{r})(f)$ is measurable by definition of stable measurable functions. So we have equipped $P \Rightarrow_m Q$ with a collection of measurability tests.

Given $(f, x) \in \mathcal{B}((P \Rightarrow_m Q) \times P) = \mathcal{B}(P \Rightarrow_m Q) \times \mathcal{B}P$ we set $\text{Ev}(f, x) = f(x) \in \mathcal{B}Q$. It is clear that this function is non-decreasing and Scott-continuous (because lubs of non-decreasing sequences of functions are computed pointwise).

LEMMA 6.6. *The evaluation map Ev is stable and measurable, i.e. $\text{Ev} \in \mathbf{Cstab}_m((P \Rightarrow_m Q) \times P, Q)$.*

PROOF (SKETCH). Stability results from Lemma 6.1, observing that Ev is linear and Scott-continuous in its first argument. The proof that Ev is measurable follows from a simple computation. \square

REMARK 3. *The proof of Lemma 6.6 strongly uses the fact that our measurability tests have parameters in \mathbb{R}^n , see Definition 5.1. If the measurability tests were just Scott-continuous linear forms (without real parameters), we would define¹⁴ the measurability tests of $P \Rightarrow_m Q$ as the $x \triangleright m \in (P \Rightarrow_m Q)'$ where $x \in \mathcal{BP}$ and $m \in \mathcal{M}(Q)$, defined by $(x \triangleright m)(f) = m(f(x))$. So, in the proof above we would only know that, for all $m \in \mathcal{M}(Q)$ and $x \in \mathcal{BP}$, the function $\mathbb{R}^n \rightarrow \mathbb{R}^+$ which maps \vec{r} to $m(\varphi(\vec{r})(x))$ is measurable. But we would have to prove that, for all $m \in \mathcal{M}(Q)$, the function $f = m \circ \text{Ev} \circ \langle \varphi, \gamma \rangle : \mathbb{R}^n \rightarrow \mathbb{R}^+$ is measurable. We have $f(\vec{r}) = m(\varphi(\vec{r})(\gamma(\vec{r})))$ and the measurability of this function does not result from what we know about φ .*

THEOREM 6.7. *The category \mathbf{Cstab}_m is cartesian closed. The pair $(P \Rightarrow_m Q, \text{Ev})$ is the object of morphisms from P to Q in \mathbf{Cstab}_m .*

PROOF. Let $f \in \mathbf{Cstab}_m(R \times P, Q)$. Let $z \in \mathcal{BR}$, consider the function $f_z : \mathcal{BP} \rightarrow \mathcal{BQ}$ defined by $f_z(x) = f(z, x)$. This function is clearly non-decreasing and Scott-continuous. Let us prove that it is pre-stable. Let $\vec{u} \in \mathcal{BP}^n$ be such that $\sum_{i=1}^n u_i \in \mathcal{BP}$ and let $x \in \mathcal{BP}_{\vec{u}}$. We have: $\Delta^\varepsilon f_z(x; \vec{u}) = \sum_{I \in \mathcal{P}_\varepsilon(n)} f_z(x + \sum_{i \in I} u_i) = \sum_{I \in \mathcal{P}_\varepsilon(n)} f(z, x + \sum_{i \in I} u_i) = \sum_{I \in \mathcal{P}_\varepsilon(n)} f((z, x) + \sum_{i \in I} (0, u_i)) = \Delta^\varepsilon f((z, x); (0, u_1), \dots, (0, u_n))$.

So we have $\Delta^- f_z(x; \vec{u}) \leq \Delta^+ f_z(x; \vec{u})$ by the assumption that f is pre-stable. We prove now that f_z is measurable. Let $n \in \mathbb{N}$ and $\gamma \in \text{Path}_1^n(P)$, we must prove that $f_z \circ \gamma \in \text{Path}^n(Q)$. So let $k \in \mathbb{N}$ and let $m \in \mathcal{M}^k(Q)$, we must prove that $m * (f_z \circ \gamma) \in \mathcal{M}^{k+n}$. Let $\vec{r} \in \mathbb{R}^k$ and $\vec{s} \in \mathbb{R}^n$, we have $m * (f_z \circ \gamma)(\vec{r}, \vec{s}) = m(\vec{r})(f(z, \gamma(\vec{s}))) = m * (f \circ \langle \zeta, \gamma \rangle)(\vec{r}, \vec{s})$, where $\zeta \in \text{Path}_1^n(R)$ is the measurable path defined by $\zeta(\vec{s}) = z$ (using Lemma 5.3). We know that $f \circ \langle \zeta, \gamma \rangle \in \text{Path}^n(Q)$ because f is measurable and $\langle \zeta, \gamma \rangle \in \text{Path}^n(P \times Q)$ by Lemma 6.2, hence $m * (f \circ \langle \zeta, \gamma \rangle) \in \mathcal{M}^{k+n}$. So $f_z \in \mathcal{B}(P \Rightarrow_m Q)$.

Let $g : \mathcal{BR} \rightarrow \mathcal{B}(P \Rightarrow_m Q)$ be the function defined by $g(z) = f_z$. We prove that g is pre-stable. Let $\vec{w} \in \mathcal{BR}^p$ be such that $\sum_{j \in J} w_j \in \mathcal{BR}$ and let $z \in \mathcal{BR}_{\vec{w}}$. We have to prove that

$$h^- = \Delta^- g(z; \vec{w}) \leq \Delta^+ g(z; \vec{w}) = h^+$$

in $\mathcal{B}(P \Rightarrow_m Q)$; we apply Lemma 6.4. So let $\vec{u} \in \mathcal{BP}^n$ and $x \in \mathcal{BP}_{\vec{u}}$, we must prove that

$$\Delta^+ h^-(x; \vec{u}) + \Delta^- h^+(x; \vec{u}) \leq \Delta^+ h^+(x; \vec{u}) + \Delta^- h^-(x; \vec{u}) \quad (8)$$

For $\varepsilon, \varepsilon' \in \{+, -\}$, we have $\Delta^\varepsilon h^{\varepsilon'}(x; \vec{u}) = \sum_{I \in \mathcal{P}_\varepsilon(n)} h^{\varepsilon'}(x + \sum_{i \in I} u_i) = \sum_{I \in \mathcal{P}_\varepsilon(n)} (\Delta^{\varepsilon'} g(z; \vec{w}))(x + \sum_{i \in I} u_i) = \sum_{I \in \mathcal{P}_\varepsilon(n)} \sum_{J \in \mathcal{P}_{\varepsilon'}(p)} g(z + \sum_{j \in J} w_j)(x + \sum_{i \in I} u_i) = \sum_{I \in \mathcal{P}_\varepsilon(n)} \sum_{J \in \mathcal{P}_{\varepsilon'}(p)} f(z + \sum_{j \in J} w_j, x + \sum_{i \in I} u_i) = \sum_{I \in \mathcal{P}_\varepsilon(n)} \sum_{J \in \mathcal{P}_{\varepsilon'}(p)} f((z, x) + \sum_{q \in J \cup (p+I)} t_q)$, where $\vec{t} \in \mathcal{B}(R \times P)^{p+n}$ is defined by

$$t_q = \begin{cases} (w_q, 0) & \text{if } q \leq p \\ (0, u_{q-p}) & \text{if } p+1 \leq q \leq p+n \end{cases}$$

and $p+I = \{p+i \mid i \in I\}$. Now observe that the map $(J, I) \mapsto J \cup (p+I)$ defines a bijection

- between $(\mathcal{P}_+(p) \times \mathcal{P}_-(n)) \cup (\mathcal{P}_-(p) \times \mathcal{P}_+(n))$ and $\mathcal{P}_-(p+n)$
- and between $(\mathcal{P}_+(p) \times \mathcal{P}_+(n)) \cup (\mathcal{P}_-(p) \times \mathcal{P}_-(n))$ and $\mathcal{P}_+(p+n)$.

¹⁴This is certainly the most natural definition in this simplified setting.

It follows that we have

$$\Delta^+ h^-(x; \vec{u}) + \Delta^- h^+(x; \vec{u}) = \Delta^- f((z, x); \vec{t}), \quad \Delta^+ h^+(x; \vec{u}) + \Delta^- h^-(x; \vec{u}) = \Delta^+ f((z, x); \vec{t})$$

and hence (8) holds because f is pre-stable. It also follows that g is pre-stable, and Scott-continuity is proven straightforwardly.

Now we prove that g is measurable. Let $\eta \in \text{Path}_1^n(R)$, we must prove that $g \circ \eta \in \text{Path}^n(P \Rightarrow_m Q)$. So let $k \in \mathbb{N}$, $\gamma \in \text{Path}_1^k(P)$ and $m \in \mathcal{M}^k(Q)$, we must prove that $(\gamma \triangleright m) * (g \circ \eta) \in \mathcal{M}^{k+n}$.

Let $\vec{r} \in \mathbb{R}^k$ and $\vec{s} \in \mathbb{R}^n$, we have $((\gamma \triangleright m) * (g \circ \eta))(\vec{r}, \vec{s}) = (\gamma \triangleright m)(\vec{r})(g(\eta(\vec{s}))) = m(\vec{r})(g(\eta(\vec{s})(\gamma(\vec{r})))$. By definition of g , this is equal to $m(\vec{r})(f(\eta(\vec{s}), \gamma(\vec{r}))) = m * (f \circ (\eta \times \gamma))(\vec{r}, \vec{s}, \vec{r})$.

We know that $\eta \times \gamma \in \text{Path}^{n+k}(P)$ because η and γ are measurable paths (and hence $\eta \circ \pi_1$ and $\gamma \circ \pi_2$ are measurable paths, for π_1 and π_2 the projections $\mathbb{R}^{n+k} \rightarrow \mathbb{R}^n$ and $\mathbb{R}^{n+k} \rightarrow \mathbb{R}^k$, and we have $\eta \times \gamma = \langle \eta \circ \pi_1, \gamma \circ \pi_2 \rangle$). Hence $m * (f \circ (\eta \times \gamma)) \in \mathcal{M}^{k+n+k}$ and therefore $(\gamma \triangleright m) * (g \circ \eta) \in \mathcal{M}^{k+n}$.

So we have proven that $g \in \text{Cstab}_m(R, P \Rightarrow_m Q)$, which ends the proof of the Theorem. \square

6.3 Integrating Measurable Paths

The map $\delta : \mathbb{R} \rightarrow \text{Meas}(\mathbb{R})$ such that $\delta(r) = \delta_r$ belongs to $\text{Path}_1^1(\text{Meas}(\mathbb{R}))$ because, given $n \in \mathbb{N}$, $U \in \Sigma_{\mathbb{R}}$, $\vec{r} \in \mathbb{R}^n$ and $r \in \mathbb{R}$ we have $\varepsilon_U(\vec{r})(\delta(r)) = \chi_U(r)$ (where χ_U is the characteristic function of U). Recall from Example 5.4 that the elements of $\mathcal{M}^n(\text{Meas}(\mathbb{R}))$ are precisely these functions ε_U .

Therefore, given $f \in \text{Cstab}_m(\text{Meas}(\mathbb{R}), P)$, the function $f \circ \delta$ is a measurable path from \mathbb{R} to P .

We have now to check that such paths are sufficiently regular for being “integrated”. More precisely, given $\gamma \in \text{Path}^1(P)$, we would like to define a linear bounded map $\gamma^\dagger : \text{Meas}(\mathbb{R}) \rightarrow P$ by integrating in P (using its algebraic structure and completeness):

$$\gamma^\dagger(\mu) = \int \gamma(r)\mu(dr)$$

but we do not know yet how to do that in general.

We will focus instead on the (not so) particular case where $P = (Q \Rightarrow_m \text{Meas}(X))$ for a cone Q and a measurable space X , which will be sufficient for our purpose in this paper (every cone which is the denotation of a PPCF type is isomorphic to a cone of the form $Q \Rightarrow_m \text{Meas}(\mathbb{R})$ for some Q).

LEMMA 6.8. *Let X be a measurable space and let $f : X \rightarrow \mathbb{R}^+$ be a measurable and bounded function. Then the function $F : \text{Meas}(X) \rightarrow \mathbb{R}^+$ defined by $F(\mu) = \int f(x)\mu(dx)$ is linear and Scott-continuous.*

LEMMA 6.9. *Let Q be a cone and X be a measurable space. A function $f : \mathcal{B}Q \rightarrow \text{Meas}(X)$ is stable iff for all $U \in \Sigma_X$, the function $f_U : \mathcal{B}Q \rightarrow \mathbb{R}^+$ defined by $f_U(y) = f(y)(U)$ is stable.*

THEOREM 6.10. *Let Q be a cone and X be a measurable space. For any $\gamma \in \text{Path}_1^1(Q \Rightarrow_m \text{Meas}(X))$, there is a measurable stable (actually linear) function $\gamma^\dagger : \text{Meas}(\mathbb{R}) \rightarrow (Q \Rightarrow_m \text{Meas}(X))$ such that $\gamma^\dagger \circ \delta = \gamma$. This function is given by*

$$\gamma^\dagger(\mu)(y)(U) = \int \gamma(r)(y)(U)\mu(dr)$$

for each $\mu \in \text{Meas}(\mathbb{R})$, $y \in Q$ and $U \in \Sigma_X$.

PROOF. Let $\gamma \in \text{Path}_1^1(Q \Rightarrow_m \text{Meas}(X))$, that we prefer to consider as a map $\gamma_0 : \mathbb{R} \times \mathcal{B}Q \times \Sigma_X \rightarrow [0, 1]$ with $\gamma(r)(y)(U) = \gamma_0(r, y, U)$. By Lemma 6.9, the fact that γ is a measurable path means that the following properties hold.

- For any $r \in \mathbb{R}$ and $y \in \mathcal{B}Q$, the map $U \mapsto \gamma_0(r, y, U)$ from Σ_X to $[0, 1]$ is a sub-probability measure;
- for any $n \in \mathbb{N}$, $\eta \in \text{Path}_1^n(Q)$ and $U \in \Sigma_X$, the map $(r, \vec{r}) \mapsto \gamma_0(r, \eta(\vec{r}), U)$ from \mathbb{R}^{1+n} to $[0, 1]$ belongs to \mathcal{M}^{1+n} (that is, is measurable);

$$\begin{aligned}
\llbracket x \rrbracket(\vec{g}, a) &= a & \llbracket r \rrbracket \vec{g} &= \delta_r & \llbracket \text{sample} \rrbracket \vec{g} &= \lambda_{[0,1]} \\
\llbracket \lambda x^A. M \rrbracket \vec{g} &= a \mapsto \llbracket M \rrbracket(\vec{g}, a) & \llbracket f(M_1, \dots, M_n) \rrbracket \vec{g} &= U \mapsto (\llbracket M_1 \rrbracket \vec{g} \otimes \dots \otimes \llbracket M_n \rrbracket \vec{g})(f^{-1}(U)) \\
\llbracket MN \rrbracket \vec{g} &= \llbracket M \rrbracket \vec{g}(\llbracket N \rrbracket \vec{g}) & \llbracket \text{ifz}(L, M, N) \rrbracket \vec{g} &= (\llbracket L \rrbracket \vec{g}(\{0\}) \llbracket M \rrbracket \vec{g} + (\llbracket L \rrbracket \vec{g}(\mathbb{R} \setminus \{0\})) \llbracket N \rrbracket \vec{g}) \\
\llbracket YM \rrbracket \vec{g} &= \sup_n ((\llbracket M \rrbracket \vec{g})^n 0) & \llbracket \text{let}(x, M, N) \rrbracket \vec{g} &= U \mapsto \int_{\mathbb{R}} \llbracket N \rrbracket(\vec{g}, \delta_r)(U) \llbracket M \rrbracket \vec{g}(dr)
\end{aligned}$$

Fig. 5. Interpretation of PPCF in Cstab_m . The terms are supposed typed as in Figure 2, and $\vec{g} \in [\Gamma]$, $a \in [A]$.

- for any $r \in \mathbb{R}$ and $U \in \Sigma_X$, the map $y \mapsto \gamma_0(r, y, U)$ from $\mathcal{B}Q$ to \mathbb{R}^+ is stable.

Therefore (by applying the second condition to $n = 0$ and η mapping the empty sequence to y) we can define a function $\varphi : \text{Meas}(\mathbb{R}) \times \mathcal{B}Q \times \Sigma_X \rightarrow \mathbb{R}^+$ by $\varphi(\mu, y, U) = \int \gamma_0(r, y, U) \mu(dr)$.

Let $\mu \in \mathcal{B}\text{Meas}(\mathbb{R})$ and $y \in \mathcal{B}Q$. The function $\Sigma_X \rightarrow [0, 1]$ which maps U to $\varphi(\mu, y, U)$ is σ -additive by linearity and continuity of integration and defines therefore an element of $\mathcal{B}\text{Meas}(X)$. We denote by φ' the function $\text{Meas}(\mathbb{R}) \times \mathcal{B}Q \rightarrow \text{Meas}(X)$ defined by $\varphi'(\mu, y)(U) = \varphi(\mu, y, U) \in \mathbb{R}^+$. This function is linear and Scott-continuous in its first argument by Lemma 6.8.

Let $\mu \in \mathcal{B}\text{Meas}(\mathbb{R})$ and $U \in \Sigma_X$. We prove that the map $f : \mathcal{B}Q \rightarrow \mathbb{R}^+$ defined by $f(y) = \varphi(\mu, y, U)$ is stable. For any $\varepsilon \in \{+, -\}$, $n \in \mathbb{N}$, $\vec{u} \in \mathcal{B}Q^n$ such that $\sum_{i=1}^n u_i \in \mathcal{B}Q$ and $y \in \mathcal{B}Q_{\vec{u}}$ one has $\Delta^\varepsilon f(y; \vec{u}) = \int \Delta^\varepsilon f_r(y; \vec{u}) \mu(dr)$, where $f_r(y) = \gamma_0(r, y, U)$, by linearity of integration. Since the function f_r is pre-stable, we have $\Delta^- f_r(y; \vec{u}) \leq \Delta^+ f_r(y; \vec{u})$ for each $r \in \mathbb{R}$ and hence $\Delta^- f(y; \vec{u}) \leq \Delta^+ f(y; \vec{u})$ as required. Given a non-decreasing sequence $(y_n)_{n \in \mathbb{N}}$ in $\mathcal{B}Q$, we must prove that $f(\sup_n y_n) = \sup_n f(y_n)$. The sequence of measurable functions $g_n : r \mapsto f_r(y_n)$ from \mathbb{R} to $[0, 1]$ is non-decreasing (for the pointwise order) and satisfies $\sup_{n \in \mathbb{N}} g_n(r) = f_r(\sup_n y_n)$ by the last condition on γ_0 , and therefore, by the monotone convergence theorem, we have $f(\sup_{n \in \mathbb{N}} y_n) = \sup_{n \in \mathbb{N}} f(y_n)$.

So φ' is stable in its second argument (using the fact that the order relation on measures in the cone $\text{Meas}(X)$ coincides with the “pointwise order”: $\mu \leq \nu$ iff $\forall U \in \Sigma_X \mu(U) \leq \nu(U)$), and linear and Scott-continuous in its first argument. Therefore, considered as a function $\mathcal{B}\text{Meas}(X) \times \mathcal{B}Q \rightarrow \text{Meas}(\mathbb{R})$, φ' is stable by Lemma 6.1. Now we must prove that this function is measurable in the sense of Definition 5.5.

So let $U \in \Sigma_X$. Let $n \in \mathbb{N}$, $\theta \in \text{Path}_1^n(\text{Meas}(\mathbb{R}))$ and $\eta \in \text{Path}_1^n(Q)$. The map $\rho : \mathbb{R}^n \rightarrow [0, 1]$ defined by $\rho(\vec{r}) = \int \gamma_0(r, \eta(\vec{r}), U) \theta(\vec{r}, dr)$ is measurable since θ is a stochastic kernel and $g : \mathbb{R}^{1+n} \rightarrow [0, 1]$ defined by $g(r, \vec{r}) = \gamma_0(r, \eta(\vec{r}), U)$ is measurable by our assumptions about γ . Therefore $\varphi' \in \text{Cstab}_m(\text{Meas}(\mathbb{R}) \times Q, \text{Meas}(X))$.

Let $\gamma^\dagger \in \text{Cstab}_m(\text{Meas}(\mathbb{R}), Q \Rightarrow_m \text{Meas}(X))$ be the currying of φ' , that is $\gamma^\dagger(\mu)(y) = \varphi'(\mu, y)$. By Theorem 6.7, γ^\dagger is stable and measurable. Observe that this function is actually linear. \square

7 SOUNDNESS AND ADEQUACY

7.1 The Interpretation of PPCF into Cstab_m

The interpretation of PPCF in Cstab_m extends the standard model of PCF in a cpo-enriched category. The ground type \mathcal{R} is denoted as the cone $\text{Meas}(\mathbb{R})$ of bounded measures over \mathbb{R} , the arrow $A \rightarrow B$ by the object of morphisms $[A] \Rightarrow_m [B]$ and a sequence A_1, \dots, A_n by the cartesian product $\prod_{i=1}^n [A_i]$ (recall Figure 4). The denotation of a judgement $\Gamma \vdash M : A$ is a morphism $\llbracket M \rrbracket^{\Gamma \vdash A} \in \text{Cstab}_m([\Gamma], [A])$, given in Figure 5 by structural induction on M . We omit the type exponent when clear from the context. Notice that if $\Gamma \vdash M : \mathcal{R}$, then for $\vec{g} \in [\Gamma]$, $\llbracket M \rrbracket \vec{g}$ is a measure on \mathbb{R} .

The fact that the definitions of Figure 5 lead to morphisms in the category \mathbf{Cstab}_m results easily from the cartesian closeness of this category and from the algebraic and order theoretic properties of its objects. The only construction which deserves further comments is the `let` construction. We use the notations of Figure 5, the typing context is $\Gamma = (x_1 : C_1, \dots, x_n : C_n)$. Let $Q = \llbracket \Gamma \rrbracket = \llbracket C_1 \rrbracket \times \dots \times \llbracket C_n \rrbracket$. By inductive hypothesis we have $\llbracket M \rrbracket \in \mathbf{Cstab}_m(Q, \text{Meas}(\mathbb{R}))$ and $\llbracket N' \rrbracket \in \mathbf{Cstab}_m(\text{Meas}(\mathbb{R}), Q \Rightarrow \text{Meas}(\mathbb{R}))$ where $N' = \lambda x_1^{C_1} \dots \lambda x_n^{C_n} N$ (up to trivial isos resulting from the cartesian closeness of \mathbf{Cstab}_m). Then $\llbracket N' \rrbracket \circ \delta \in \text{Path}_1^1(Q \Rightarrow \text{Meas}(\mathbb{R}))$ because $\delta \in \text{Path}_1^1(\text{Meas}(\mathbb{R}))$ (see Section 6.3). Hence we define $(\llbracket N' \rrbracket \circ \delta)^\dagger \in \mathbf{Cstab}_m(\text{Meas}(\mathbb{R}), Q \Rightarrow \text{Meas}(\mathbb{R}))$ by setting $(\llbracket N' \rrbracket \circ \delta)^\dagger(\mu)(\vec{g})(U) = \int \llbracket N' \rrbracket(\delta_r)(\vec{g})(U) \mu(dr) = \int \llbracket N \rrbracket(\vec{g}, \delta_r)(U) \mu(dr)$ for $\mu \in \text{Meas}(\mathbb{R})$, $\vec{g} \in Q = \llbracket \Gamma \rrbracket$ and $U \in \Sigma_{\mathbb{R}}$, by Theorem 6.10 (remember that $\Gamma, x : \mathcal{R} \vdash N : \mathcal{R}$). By cartesian closeness, we define f in $\mathbf{Cstab}_m(Q, \text{Meas}(\mathbb{R}))$ by $f(\vec{g})(U) = (\llbracket N' \rrbracket \circ \delta)^\dagger(\llbracket M \rrbracket(\vec{g}))(\vec{g})(U) = \int \llbracket N \rrbracket(\vec{g}, \delta_r)(U) \llbracket M \rrbracket(\vec{g})(dr)$. Hence, $\llbracket \text{let}(x, M, N) \rrbracket$ belongs to $\mathbf{Cstab}_m(Q, \text{Meas}(\mathbb{R}))$ as required. Observe moreover that, for $r \in \mathbb{R}$, we have $\llbracket \text{let}(x, \underline{r}, N) \rrbracket \vec{g} = \llbracket N \rrbracket(\vec{g}, \delta_r)$.

Example 7.1. Numerals are associated with Dirac measures and a functional constant \underline{f} yields the pushforward measure of the product of the measures denoting the arguments of \underline{f} . For example, we have: $\llbracket \underline{\pm}(3, \underline{2}) \rrbracket^{\mathcal{R}} = U \mapsto \delta_3 \otimes \delta_2(\{(r_1, r_2) \text{ s.t. } r_1 + r_2 \in U\}) = \delta_5$.

The construct `ifz` sums up the denotation of the two branches according to the probability that the first term evaluates to $\underline{0}$ or not. Given a measurable set $U \subseteq \mathbb{R}$, a closed term L of ground type and two closed terms M, N of a type A , we have that, recalling the notation of Example 3.1, $\llbracket \text{if}(L \in U, M, N) \rrbracket^A = (\llbracket \chi_U(L) \rrbracket^{\mathcal{R}}(\mathbb{R} \setminus \{0\})) \llbracket M \rrbracket^A + (\llbracket \chi_U(L) \rrbracket^{\mathcal{R}}(\{0\})) \llbracket N \rrbracket^A = (\llbracket L \rrbracket^{\mathcal{R}}(U)) \llbracket M \rrbracket + (\llbracket L \rrbracket^{\mathcal{R}}(\mathbb{R} \setminus U)) \llbracket N \rrbracket$.

Example 7.2. The two terms implementing the diagonal in Example 3.10 have different semantics: for any measurable U of \mathbb{R} , for any $r, s \in U$, $r = s$ has value 0 or 1. Besides, the diagonal $\{(r, s) \text{ s.t. } r = s \in \{1\}\}$ in $[0, 1]^2$ has measure 0, and its complementary $\{(r, s) \text{ s.t. } r = s \in \{0\}\}$ has measure 1. Thus,

$$\llbracket (\lambda x. (x = x)) \text{sample} \rrbracket^{\mathcal{R}}(U) = (\lambda_{[0,1]} \otimes \lambda_{[0,1]})\{(r, s) \text{ s.t. } r = s \in U\} = \delta_0(U).$$

On the contrary, $\llbracket \text{let}(x, \text{sample}, x = x) \rrbracket^{\mathcal{R}}(U) = \delta_1(U)$. Indeed, $\llbracket \text{let}(x, \text{sample}, x = x) \rrbracket^{\mathcal{R}}(U)$ is

$$\int_{\mathbb{R}} (\delta_r \otimes \delta_r)\{(x, y) \text{ s.t. } x = y \in U\} \lambda_{[0,1]}(dr) = \int_{\mathbb{R}} \delta_1(U) \lambda_{[0,1]}(dr) = \delta_1(U).$$

Example 7.3. Let us compute the semantics of the encodings of the distributions in Example 3.3. Let $p \in [0, 1]$, then $\llbracket \text{bernoulli } p \rrbracket^{\mathcal{R}} = p\delta_1 + (1 - p)\delta_0$ is given by, for U measurable: $\llbracket \text{bernoulli } p \rrbracket^{\mathcal{R}}(U) = \int_{\mathbb{R}} \delta_r \otimes \delta_p(\{(x, y) \text{ s.t. } x \leq y \in U\}) \lambda_{[0,1]}(dr)$, this latter being equal to $\lambda_{[0,1]}([0, p])\delta_1(U) + \lambda_{[0,1]}((p, 1])\delta_0(U)$.

The exponential distribution `exp` computes the probability that an exponential random variable belongs to U : $\llbracket \text{exp} \rrbracket^{\mathcal{R}}(U) = \llbracket \text{let}(x, \text{sample}, -\log(x)) \rrbracket^{\mathcal{R}}(U) = \int_{\mathbb{R}} \delta_r(\{x \text{ s.t. } -\log x \in U\}) \lambda_{[0,1]}(dr) = \int_{\mathbb{R}} \chi_U(-\log r) \lambda_{[0,1]}(dr)$, which is equal to $\int_{\mathbb{R}^+} \chi_U(s) e^{-s} \lambda(ds)$ by substitution $r = e^{-s}$. We compute the semantics of normal and check that we get a normal distribution:

$$\begin{aligned} \llbracket \text{normal} \rrbracket^{\mathcal{R}}(U) &= \llbracket \text{let}(x, \text{sample}, \text{let}(y, \text{sample}, \underline{(-2 \log(x))}^{\frac{1}{2}} \cos(2\pi y))) \rrbracket^{\mathcal{R}}(U) \\ &= \int_{\mathbb{R}^2} \chi_U(\sqrt{-2 \log u} \cos(2\pi v)) \lambda_{[0,1]}(du) \lambda_{[0,1]}(dv). \end{aligned}$$

By polar substitution with $x = \sqrt{-2 \log u} \cos(2\pi v)$, $y = \sqrt{-2 \log u} \sin(2\pi v)$, we then have:
 $\llbracket \text{normal} \rrbracket^{\mathcal{R}}(U) = \frac{1}{2\pi} \int_{\mathbb{R}^2} \chi_U(x) e^{-\frac{(x^2+y^2)}{2}} \lambda(dx) \lambda(dy) = \frac{1}{\sqrt{2\pi}} \int_U e^{-\frac{x^2}{2}} \lambda(dx)$, which is what we wanted.

Similarly, $\llbracket \text{gauss } r \sigma \rrbracket^{\mathcal{R}}(U) = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} \chi_U(\sigma y + r) e^{-\frac{y^2}{2}} \lambda(dy) = \frac{1}{\sigma \sqrt{2\pi}} \int_U e^{-\frac{(z-r)^2}{\sigma^2}} \lambda(dz)$.

Example 7.4. Recall Example 3.5, let $f \in C$ and M be a term of type \mathcal{R} . We want to check that $\text{expectation}_n f M$ corresponds to the n -th estimate of the expectation of f with respect to the measure $\llbracket M \rrbracket^{\mathcal{R}}$, meaning that $\llbracket \text{expectation}_n f M \rrbracket^{\mathcal{R}}$ has the same measure as $\frac{f(x_1) + \dots + f(x_n)}{n}$ where x_i 's are iid random variables of measure $\llbracket M \rrbracket^{\mathcal{R}}$. For all $U \subseteq \mathbb{R}$ measurable, $\llbracket \text{expectation}_n f M \rrbracket^{\mathcal{R}}(U) = \underbrace{\llbracket M \rrbracket^{\mathcal{R}} \otimes \dots \otimes \llbracket M \rrbracket^{\mathcal{R}}}_n(\{(x_1, \dots, x_n) \text{ s.t. } \frac{f(x_1) + \dots + f(x_n)}{n} \in U\})$ which is what we wanted.

The following two lemmas are standard and proven by structural induction.

LEMMA 7.5 (SUBSTITUTION PROPERTY). *Given $y : B, \Gamma \vdash M : A$ and $\Gamma \vdash N : B$ we have, for every $\vec{g} \in \llbracket \Gamma \rrbracket$, that $\llbracket M \rrbracket^{y:B, \Gamma \vdash A}(\llbracket N \rrbracket^{\Gamma \vdash B} \vec{g}) \vec{g} = \llbracket M\{N/y\} \rrbracket^{\Gamma \vdash A} \vec{g}$.*

LEMMA 7.6 (LINEARITY EVALUATION CONTEXT). *Let $y : B, \Gamma \vdash E[y] : A$ for $E[\]$ an evaluation context and y a fresh variable. Then $\llbracket E[y] \rrbracket^{y:B, \Gamma \vdash A} \in \mathbf{Cstab}_m(\llbracket B \rrbracket \times \llbracket \Gamma \rrbracket, \llbracket A \rrbracket)$ is a linear function in its first argument $\llbracket B \rrbracket$.*

7.2 Soundness

The soundness property states that the interpretation is invariant under reduction. In a non-deterministic case, this means that the semantics of a term is the sum of the semantics of all its possible one-step reducts, see e.g. [Laird et al. 2013]. In our setting, the reduction is a stochastic kernel, so this sum becomes an integral, i.e. for all $M \in \Lambda^{\Gamma \vdash A}$,

$$\llbracket M \rrbracket^{\Gamma \vdash A} = \int_{\Lambda^{\Gamma \vdash A}} \llbracket t \rrbracket^{\Gamma \vdash A} \text{Red}(M, dt) \quad (9)$$

The following lemma actually proves that the above integral is a meaningful notation for the function mapping $\vec{g} \in \llbracket \Gamma \rrbracket$, and, supposing $A = B_1 \rightarrow \dots \rightarrow B_k \rightarrow \mathcal{R}$, $b_1 \in \llbracket B_1 \rrbracket, \dots, b_k \in \llbracket B_k \rrbracket$ and $U \in \Sigma_{\mathcal{R}}$, to $\int_{\Lambda^{\Gamma \vdash A}} \llbracket t \rrbracket^{\Gamma \vdash A} \vec{g} b_1 \dots b_k(U) \text{Red}(M, dt)$, this latter being well-defined because $\llbracket t \rrbracket^{\Gamma \vdash A} \vec{g} b_1 \dots b_k(U)$ is measurable (Lemma 7.7) and $\text{Red}(M, _)$ is a measure (Proposition 3.8).

LEMMA 7.7. *Let $\Gamma \vdash M : A$, with $A = B_1 \rightarrow \dots \rightarrow B_k \rightarrow \mathcal{R}$. For all $i \leq k$, let $b_i \in \llbracket B_i \rrbracket$ and $\vec{g} \in \llbracket \Gamma \rrbracket$, then the map $M \mapsto \llbracket M \rrbracket^{\Gamma \vdash A} \vec{g} b_1 \dots b_k$ is a stochastic kernel from $\Lambda^{\Gamma \vdash A}$ to \mathbb{R} .*

PROOF (SKETCH). By (3), it is enough to prove that, for any $S \in \Lambda_n^{\Gamma \vdash A}$, the restriction $\llbracket _ \rrbracket_S \vec{g} \vec{b}$ of $\llbracket _ \rrbracket \vec{g} \vec{b}$ to $\Lambda_S^{\Gamma \vdash A}$ is a kernel. This is done by using the crucial fact that the map $h = \llbracket S \rrbracket \circ (\delta^n \times \vec{g})$ is a measurable path in $\text{Path}_1^n(\llbracket A \rrbracket)$. This implies that $\vec{r} \mapsto h(\vec{r}) \vec{b}$ is in $\text{Path}_1^n(\text{Meas}(\mathbb{R}))$, so it is a stochastic kernel from \mathbb{R}^n to \mathbb{R} (Example 5.4). We are done, since \mathbb{R}^n and $\Lambda_S^{\Gamma \vdash A}$ are isomorphic. \square

PROPOSITION 7.8 (SOUNDNESS). *Let $A = B_1 \rightarrow \dots \rightarrow B_k \rightarrow \mathcal{R}$, for all $i \leq k$, $b_i \in \llbracket B_i \rrbracket$, and let $\vec{g} \in \llbracket \Gamma \rrbracket$, then $(\llbracket _ \rrbracket^{\Gamma \vdash A} \vec{g} b_1 \dots b_k) \circ \text{Red} = \llbracket _ \rrbracket^{\Gamma \vdash A} \vec{g} b_1 \dots b_k$, i.e. Equation (9) holds for any $M \in \Lambda^{\Gamma \vdash A}$.*

PROOF (SKETCH). If M is a normal form, then the statement is trivial. Otherwise, let $M = E[R]$ with R a redex (Lemma 3.6). If $R \neq \text{sample}$, let $R \rightarrow N$. By the substitution property (Lemma 7.5)) it is sufficient to prove $\llbracket R \rrbracket = \llbracket N \rrbracket$ to conclude. This is done by cases, depending on the type of R .

The last case is $M = E[\text{sample}]$. This is obtained by using the linearity of the evaluation context $E[\]$ (Lemma 7.6) and the substitution property (Lemma 7.5). \square

Example 7.9. Suppose M a closed term of type \mathcal{R} and consider $\vdash \text{observe}(U)M : \mathcal{R}$ introduced in Example 3.4 as an encoding of the conditioning. We compute its semantics by using soundness. Since $\text{observe}(U)M \rightarrow^* \text{let}(x, M, \text{if}(x \in U, x, \text{observe}(U)M))$, we get by soundness that for all $V \subseteq \mathbb{R}$ measurable, $\llbracket \text{observe}(U)M \rrbracket(V) = \int_{\mathbb{R}} \llbracket \text{if}(x \in U, x, \text{observe}(U)M) \rrbracket^{x:\mathcal{R} \vdash \mathcal{R}}(\delta_r)(V) \llbracket M \rrbracket(dr) = \int_{\mathbb{R}} (\delta_r(U) \delta_r(V) + (\delta_r(\mathbb{R} \setminus U)) (\llbracket \text{observe}(U)M \rrbracket(V))) \llbracket M \rrbracket(dr)$. Since $\llbracket \text{observe}(U)M \rrbracket$ does not depend on r , the latter integral can be rewritten to: $\llbracket \text{observe}(U)M \rrbracket(V) = \int_{\mathbb{R}} (\chi_U(r) \chi_V(r)) \llbracket M \rrbracket(dr) + (\llbracket \text{observe}(U)M \rrbracket(V)) \int_{\mathbb{R}} \chi_{\mathbb{R} \setminus U}(r) \llbracket M \rrbracket(dr)$.

Whenever M represents a probability distribution, so that $\llbracket M \rrbracket(U) = 1 - \llbracket M \rrbracket(\mathbb{R} \setminus U)$ and if moreover $\llbracket M \rrbracket(U) \neq 0$, this equation gives the conditional probability:

$$\llbracket \text{observe}(U)M \rrbracket(V) = \frac{\int_{\mathbb{R}} (\chi_U(r) \chi_V(r)) \llbracket M \rrbracket(dr)}{1 - \int_{\mathbb{R}} \chi_{\mathbb{R} \setminus U}(r) \llbracket M \rrbracket(dr)} = \frac{\llbracket M \rrbracket(V \cap U)}{\llbracket M \rrbracket(U)}$$

If $\llbracket M \rrbracket(U) = 0$, then as $(\llbracket \text{let}(x, M, \text{if}(x \in U, x, y)) \rrbracket)^n 0 = 0$, the denotation of the fixpoint is $\llbracket \text{observe}(U)M \rrbracket = 0$. By adequacy, the program then loops with probability 1 when $\llbracket M \rrbracket(U) = 0$.

Now, consider the term $O = \lambda m. Y(\lambda y. \text{if}(m \in U, m, y))$ presented in Example 3.4 as a wrong implementation of $\text{observe}(U)$. Since $OM \rightarrow^* \text{if}(M \in U, M, OM)$, assuming that $\llbracket M \rrbracket$ is a probability distribution and V a measurable set, one gets with a similar reasoning that, in case $\llbracket M \rrbracket U \neq 0$, $\llbracket OM \rrbracket(V) = (\llbracket M \rrbracket(V) \llbracket M \rrbracket(U)) / \llbracket M \rrbracket(U) = \llbracket M \rrbracket(V)$. As before, if $\llbracket M \rrbracket(U) = 0$, then $\llbracket OM \rrbracket = 0$.

7.3 Adequacy

Let M be a closed term of ground type of PPCF. Both the operational and the denotational semantics associate with M a distribution over \mathbb{R} — the adequacy property states that these two distributions are actually the same (Theorem 7.12). The proof is standard: the soundness property gives as a corollary that the “operational” distribution is bounded by the “denotational” one. The converse is obtained by using a suitable logical relation (Definition 7.10, Lemma 7.11).

Definition 7.10. By induction on a type A , we define a relation $<^A \subseteq \llbracket A \rrbracket \times \Lambda^{\vdash A}$ as follows:

$$\begin{aligned} \mu <^{\mathcal{R}} M & \quad \text{iff} \quad \forall U \in \Sigma_{\mathbb{R}}, \mu(U) \leq \text{Red}^{\infty}(M, \underline{U}), \\ f <^{A \rightarrow B} M & \quad \text{iff} \quad \forall u <^A N, f(u) <^B MN. \end{aligned}$$

LEMMA 7.11. *Let $x_1 : B_1, \dots, x_n : B_n \vdash M : A$ and $\forall i \leq n, u_i <^{B_i} N_i$, then: $\llbracket M \rrbracket \vec{u} <^A M\{\vec{N}/\vec{x}\}$.*

THEOREM 7.12 (ADEQUACY). *Let $\vdash M : \mathcal{R}$, then for every measurable set $U \subseteq \mathbb{R}$, we have:*

$$\llbracket M \rrbracket^{\vdash \mathcal{R}}(U) = \text{Red}^{\infty}(M, \underline{U})$$

where \underline{U} is the set of numerals corresponding to the real numbers in U .

8 RELATED WORK AND CONCLUSION

The first denotational models for higher-order probabilistic programming were based on probabilistic power domains [Jones and Plotkin 1989; Saheb-Djahromi 1980]. This setting follows a monadic approach, considering a program as a function from inputs to the probabilistic power domain of its outputs. The major issue here is to find a cartesian closed category which is also closed under the probabilistic power domain monad [Jung and Tix 1998]. Some advances have been obtained by Barker [2016], using a monad based on random variables inspired by Goubault-Larrecq and Varacca [2011]. Besides, Mislove [2016] has introduced a domain theory of random variables. Another approach is based on game semantics, designing models of probabilistic languages with references [Danos and Harmer 2002] or concurrent features [Winskel 2014].

The notions of *d-cones* [Tix et al. 2009] and *Kegelspitzen* [Keimel and Plotkin 2017] are promising for getting a family of models different from ours. Rennela [2016] has recently used this approach for studying a probabilistic extension of FPC. A *Kegelspitzen* is a convex set of a positive cone equipped with an order compatible with the algebraic structure of the cone. Notice that this notion differs from ours because the order of a *Kegelspitzen* might be independent from the one induced by its algebraic structure. It is likely that the two approaches live in two different but related frameworks as the continuous and the stable semantics of standard PCF.

The denotational semantics approach to probabilistic programming has been recently relaunched by the increasing importance of continuous distributions and sampling primitives. Indeed, this raises the question of the *measurability of a morphism* as the interpretation of the sampling primitives requires integration. This question has not been investigated yet in the domain theoretic approach and forces to introduce a new line of works which puts the focus on measurability.

The challenge is to define a cartesian closed category in which base types such as reals would be interpreted as measurable spaces. As mentioned in the Introduction, the category **Meas** of measurable spaces and functions is cartesian but not closed. To overcome this problem, Staton et al. [2016] embed **Meas** in a functor category which is cartesian closed although not well-pointed. Then, to get a more concrete and a well-pointed category, they introduce the category of *quasi-borel spaces* [Heunen et al. 2017] which are sets endowed with a set of random variables. Notice that both categories miss the order completeness, and thus the possibility of interpreting higher-order recursion. This is a big difference with our model \mathbf{Cstab}_m which is order complete.

Let us also cite the ongoing efforts presented last year at the workshop PPS by Huang and Morrisett [2017], aiming to give a model based on computable distributions, and by Faissole and Spitters [2017], working on a Coq formalization of a semantics built on top of the constructions detailed in [Staton et al. 2016].

In this paper, we have presented \mathbf{Cstab}_m , a new model of higher-order probabilistic computations with full recursion, as a cartesian closed category enriched over posets which are complete for non-decreasing sequences. The objects of \mathbf{Cstab}_m are cones equipped with a notion of measurability tests and morphisms are functions which are measurable in the sense that they behave well wrt. this notion of measurability tests. These functions are also Scott-continuous, but this is not sufficient for guaranteeing cartesian closeness: they must satisfy an hereditary monotonicity condition that we call *stability* because, when adapted to coherence spaces, it coincides with Berry-Girard stability. The introduction of this notion of “probabilistic stability” is a relevant byproduct of our approach.

A typical example of such a cone is the set of \mathbb{R}^+ -valued measures on the real line that we use to interpret the type of real numbers, the unique ground type of PPCF, a probabilistic version of PCF. This language also features a *sample* primitive allowing to sample a real number according to a prescribed probability measure on the reals (intuitively, a closed PPCF term of ground type represents a sub-probability measure on the real line). We have presented the semantics of PPCF in \mathbf{Cstab}_m and proven adequacy for a call-by-name operational semantics.

There are many research directions suggested by these new constructions, namely to study the category \mathbf{Clin}_m of linear and measurable Scott continuous maps mentioned in the Introduction and prove the conjectures sketched in Figure 1. Also, full-abstraction will be addressed, following [Ehrhard et al. 2014].

ACKNOWLEDGMENTS

We would like to thank Ugo Dal Lago and Jean-Louis Krivine for useful discussions. This material is based upon work supported by the ANR grant Elica (No. ANR-14-CE25-0005).

REFERENCES

- Tsuyoshi Andô. 1962. On fundamental properties of a Banach space with a cone. *Pacific J. Math.* 12, 4 (1962), 1163–1169.
- Robert J. Aumann. 1961. Borel structures for function spaces. *Illinois J. Math.* 5, 4 (12 1961), 614–630. <https://projecteuclid.org/443/euclid.ijm/1255631584>
- Tyler Barker. 2016. *A monad for randomized algorithms*. Ph.D. Dissertation. Tulane University.
- Gérard Berry. 1978. Stable Models of Typed Lambda-calculi. In *Automata, Languages and Programming, Fifth Colloquium, Udine, Italy, July 17-21, 1978, Proceedings (Lecture Notes in Computer Science)*, Vol. 62. Springer, 72–89.
- Johannes Borgström, Ugo Dal Lago, Andrew D. Gordon, and Marcin Szymczak. 2016. A lambda-calculus foundation for universal probabilistic programming. In *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming, ICFP 2016, Nara, Japan, September 18-22, 2016*, Jacques Garrigue, Gabriele Keller, and Eijiro Sumii (Eds.). ACM, 33–46. <https://doi.org/10.1145/2951913.2951942>
- Vincent Danos and Thomas Ehrhard. 2011. Probabilistic coherence spaces as a model of higher-order probabilistic computation. *Information and Computation* 209, 6 (2011), 966–991.
- Vincent Danos and Russel Harmer. 2002. Probabilistic game semantics. *ACM Transactions on Computational Logic* 3, 3 (July 2002), 359–382.
- Thomas Ehrhard, Michele Pagani, and Christine Tasson. 2011. The Computational Meaning of Probabilistic Coherence Spaces. In *Proceedings of the 26th Annual IEEE Symposium on Logic in Computer Science (LICS 2011)*, Martin Grohe (Ed.). IEEE Computer Society Press, 87–96.
- Thomas Ehrhard, Michele Pagani, and Christine Tasson. 2014. Probabilistic Coherence Spaces are Fully Abstract for Probabilistic PCF. In *The 41th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL14, San Diego, USA, P. Sewell (Ed.)*. ACM.
- Thomas Ehrhard, Michele Pagani, and Christine Tasson. 2017. The cartesian closed category of measurable cones and stable, measurable functions A model for probabilistic higher-order programming. (Oct. 2017). <https://hal.archives-ouvertes.fr/hal-01622046> working paper or preprint.
- Thomas Ehrhard and Christine Tasson. 2016. Probabilistic call by push value. (2016). Submitted, preprint available at <http://arxiv.org/abs/1607.04690>.
- Martin Escardó, Martin Hofmann, and Thomas Streicher. 2004. On the non-sequential nature of the interval-domain model of real-number computation. *Mathematical Structures in Computer Science* 14, 6 (2004), 803–814. <https://doi.org/10.1017/S0960129504004360>
- Martin Escardó. 1996. PCF extended with real numbers. *Theoretical Computer Science* 162, 1 (1996), 79 – 115. [https://doi.org/10.1016/0304-3975\(95\)00250-2](https://doi.org/10.1016/0304-3975(95)00250-2)
- Florian Faissole and Bas Spitters. 2017. Synthetic topology in Homotopy Type Theory for probabilistic programming. (2017). available at: <https://pps2017.soic.indiana.edu/files/2016/12/ProbProg.pdf>.
- Michèle Giry. 1982. *A categorical approach to probability theory*. Springer Berlin Heidelberg, Berlin, Heidelberg, 68–85. <https://doi.org/10.1007/BFb0092872>
- Noah D. Goodman and Joshua B. Tenenbaum. 2014. Probabilistic models of cognition. (2014). <http://probmods.org>.
- Jean Goubault-Larrecq and Daniele Varacca. 2011. Continuous Random Variables. In *LICS*. IEEE Computer Society, 97–106.
- Chris Heunen, Ohad Kammar, Sam Staton, and Hongseok Yang. 2017. A Convenient Category for Higher-Order Probability Theory. In *Proceedings of the 32st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '17, Reykjavik, June 20-23, 2017*, Joel Ouaknine (Ed.). ACM.
- Daniel Huang and Greg Morrisett. 2017. An application of computable distributions to the semantics of probabilistic programs: part 2. (2017). available at: <https://pps2017.soic.indiana.edu/files/2016/12/comp-dist-sem.pdf>.
- Claire Jones and Gordon D. Plotkin. 1989. A Probabilistic Powerdomain of Evaluations. In *Proceedings of the Fourth Annual Symposium on Logic in Computer Science (LICS '89), Pacific Grove, California, USA, June 5-8, 1989*. IEEE Computer Society, 186–195.
- Achim Jung and Regina Tix. 1998. The troublesome probabilistic powerdomain. *Electr. Notes Theor. Comput. Sci.* 13 (1998), 70–91. [https://doi.org/10.1016/S1571-0661\(05\)80216-6](https://doi.org/10.1016/S1571-0661(05)80216-6)
- Klaus Keimel and Gordon D. Plotkin. 2017. Mixed powerdomains for probability and nondeterminism. *Logical Methods in Computer Science* 13, 1 (2017). [https://doi.org/10.23638/LMCS-13\(1:2\)2017](https://doi.org/10.23638/LMCS-13(1:2)2017)
- Dexter Kozen. 1981. Semantics of Probabilistic Programs. *J. Comput. System Sci.* 22, 3 (1981), 328–350. [https://doi.org/10.1016/0022-0000\(81\)90036-2](https://doi.org/10.1016/0022-0000(81)90036-2)
- Jim Laird, Giulio Manzonetto, Guy McCusker, and Michele Pagani. 2013. Weighted relational models of typed lambda-calculi. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013*. IEEE Computer Society.
- Brockway McMillan. 1954. Absolutely Monotone Functions. *Annals of Mathematics* 60, 3 (1954), 467–501.
- Michael W. Mislove. 2016. Domains and Random Variables. *CoRR* abs/1607.07698 (2016). <http://arxiv.org/abs/1607.07698>

- Prakash Panangaden. 1999. The Category of Markov Kernels. *Electronic Notes in Theoretical Computer Science* 22 (1999), 171 – 187. [https://doi.org/10.1016/S1571-0661\(05\)80602-4](https://doi.org/10.1016/S1571-0661(05)80602-4) PROBMIV'98, First International Workshop on Probabilistic Methods in Verification.
- Sungwoo Park, Frank Pfenning, and Sebastian Thrun. 2008. A Probabilistic Language Based on Sampling Functions. *ACM Trans. Program. Lang. Syst.* 31, 1, Article 4 (Dec. 2008), 46 pages. <https://doi.org/10.1145/1452044.1452048>
- Gordon D. Plotkin. 1977. LCF Considered as a Programming Language. *Theor. Comput. Sci.* 5, 3 (1977), 225–255.
- Mathys Rennela. 2016. Convexity and Order in Probabilistic Call-by-Name FPC. *CoRR* abs/1607.04332 (2016). <http://arxiv.org/abs/1607.04332>
- N. Saheb-Djahromi. 1980. CPO'S of Measures for Nondeterminism. *Theor. Comput. Sci.* 12 (1980), 19–37.
- Peter Selinger. 2004. Toward a semantics for higher-order quantum computation. In *Proceedings of the 2nd International Workshop on Quantum Programming Languages*, Peter Selinger (Ed.), Vol. 33. TUCS General Publication, 127–143.
- Sam Staton. 2017. Commutative Semantics for Probabilistic Programming. In *Programming Languages and Systems - 26th European Symposium on Programming, ESOP 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings (Lecture Notes in Computer Science)*, Hongseok Yang (Ed.), Vol. 10201. Springer, 855–879. https://doi.org/10.1007/978-3-662-54434-1_32
- Sam Staton, Hongseok Yang, Frank Wood, Chris Heunen, and Ohad Kammar. 2016. Semantics for probabilistic programming: higher-order functions, continuous distributions, and soft constraints. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016*. ACM, 525–534. <https://doi.org/10.1145/2933575.2935313>
- Regina Tix, Klaus Keimel, and Gordon D. Plotkin. 2009. Semantic Domains for Combining Probability and Non-Determinism. *Electr. Notes Theor. Comput. Sci.* 222 (2009), 3–99. <https://doi.org/10.1016/j.entcs.2009.01.002>
- Jean Vuillemin. 1988. Exact Real Computer Arithmetic with Continued Fractions. In *Proceedings of the 1988 ACM Conference on LISP and Functional Programming (LFP '88)*. ACM, New York, NY, USA, 14–27. <https://doi.org/10.1145/62678.62681>
- Glynn Winskel. 2014. Probabilistic and Quantum Event Structures. In *Horizons of the Mind. A Tribute to Prakash Panangaden - Essays Dedicated to Prakash Panangaden on the Occasion of His 60th Birthday (Lecture Notes in Computer Science)*, Franck van Breugel, Elham Kashefi, Catuscia Palamidessi, and Jan Rutten (Eds.), Vol. 8464. Springer, 476–497. https://doi.org/10.1007/978-3-319-06880-0_25