Compositionality

Simons Institute - Berkeley

Thursday, December 8th, 2016

Probabilistic Full Abstraction

joint work with **T. Ehrhard** and **M. Pagani**

Christine Tasson

Christine.Tasson@irif.fr

Laboratoire IRIF - Université Paris Diderot

A denotational semantics for discrete probabilistic higher-order functional computation, which is fully abstract

(based on **quantitative** semantics)

1. General framework: Domains vs Quantitative semantics
2. PCoh: Probabilistic Coherent Spaces
3. Probabilistic Call by Name
4. Probabilistic Call by Push Value

A denotational semantics for discrete probabilistic higher-order functional computation, which is fully abstract

(based on **quantitative** semantics)

1. **General framework: Domains vs Quantitative semantics**
2. PCoh: Probabilistic Coherent Spaces
3. Probabilistic Call by Name
4. Probabilistic Call by Push Value

General Framework	Domains Semantics	Quantitative Semantics
Types	Continuous **dcpos** $(X, \leq)$	**Proba.** spaces $(\|X\|, \mathrm{P}(X) \subseteq (\mathbb{R}^+)^{\|X\|})$
Programs	**Scott Continuous**	**Analytic** Functions
Probability	Proba. **monad**	**Values** as proba. distr.

Bibliography

1976 Plotkin
1981 Kozen
1989 Plotkin and Jones
1998 Jung and Tix
2013 Goubault Larrecq and Varraca
2013 Mislove

Bibliography

1988 Girard
1994 Blute, Panangaden and Seely
2002 Hasegawa
2004 Girard
2011 Danos and Ehrhard
2014 Ehrhard, Pagani, T.
2016 Ehrahrd, T.

General Framework	Domains Semantics	Quantitative Semantics
Types	Continuous **dcpos** $(X, \leq)$	**Proba.** spaces $(\lvert X\rvert, \mathrm{P}(X) \subseteq (\mathbb{R}^+)^{\lvert X\rvert})$
Programs	**Scott Continuous**	**Analytic** Functions
Probability	Proba. **monad**	**Values** as proba. distr.

How to interpret a program $M : \mathtt{nat} \Rightarrow \mathtt{nat}$

Type: $\mathbb{N}_\perp$ flat domain,
$\mathcal{V}(\mathbb{N}_\perp)$ proba. distr. over $\mathbb{N}_\perp$,

Prog: $[\![M]\!] : \mathbb{N}_\perp \to \mathcal{V}(\mathbb{N}_\perp)$,
$[\![\mathtt{let\ n=x\ in\ M}]\!] : \mathcal{V}(\mathbb{N}_\perp) \to \mathcal{V}(\mathbb{N}_\perp)$

$$x \mapsto \left(\sum_n [\![M]\!]_{n,q} x_n\right)_q$$

Type: $\lvert\mathbf{Nat}\rvert = \mathbb{N}$
$\mathrm{P}(\mathbf{Nat})$ subproba. dist. over $\mathbb{N}$

Prog: $[\![M]\!] : \mathrm{P}(\mathbf{Nat}) \to \mathrm{P}(\mathbf{Nat})$

$$x \mapsto \left(\sum_{\mu=[n_1,\dots,n_k]} [\![M]\!]_{\mu,q} \prod_{i=1}^{k} x_{n_i}\right)_q$$

General Framework	Domains Semantics	Quantitative Semantics
Types	Continuous **dcpos** $(X, \leq)$	**Proba.** spaces $(\lvert X\rvert, \mathrm{P}(X) \subseteq (\mathbb{R}^+)^{\lvert X\rvert})$
Programs	**Scott Continuous**	**Analytic** Functions
Probability	Proba. **monad**	**Values** as proba. distr.

Problematic in **domain**

Finding a full subcategory of continuous dcpos that is: **Cartesian Closed** and **closed** under the proba. monad $\mathcal{V}$.

Full Abs.: **PCOH/pPCF**

$$\textbf{Prob}(C[M], \underline{n}) \overset{\forall n,\ \forall C[\,]}{=} \textbf{Prob}(C[N], \underline{n})$$

iff

$$[\![M]\!] = [\![N]\!].$$

A denotational semantics to discrete probabilistic higher-order functional computation, which is fully abstract

(based on **quantitative** semantics)

1. General framework: Domains vs Quantitative semantics
2. **PCoh: Probabilistic Coherent Spaces**
3. Probabilistic Call by Name
4. Probabilistic Call by Push Value

Types as **Probabilistic Coherent Spaces**: $(|X|, P(X))$

Proba. Space

$|X|$: the **web**, a (potentially infinite) set of final states

$\mathrm{P}(X)$: a set of vectors $\subseteq (\mathbb{R}^+)^{|X|}$ such that

closure: $\mathbf{P}(\mathbf{X})^{\perp\perp} = \mathbf{P}(\mathbf{X})$ with

$$\forall u, v \in (\mathbb{R}^+)^{|X|},\ \langle u, v\rangle = \textstyle\sum_{a\in|X|} u_a v_a$$

$$\forall P \subseteq (\mathbb{R}^+)^{|X|},\ P^\perp = \{v \in (\mathbb{R}^+)^{|X|}\ ;\ \forall u \in P,\ \langle u, v\rangle \leq 1\}$$

bounded covering: $\forall a \in |X|$,

$\exists v \in P(X)\ ;\ v_a \neq 0$ and $\exists p > 0,\ ;\ \forall v \in \mathrm{P}(X),\ v_a \leq p.$

Proposition: Proba. spaces as Domains

$(|X|, \mathrm{P}(X))$ is a **Proba. space iff** $\mathrm{P}(X)$ is bounded covering, **Scott Closed** (downwards-closed and dcpo) and **Convex**.

Types as **Probabilistic Coherent Spaces**: $(|X|, P(X))$

Example

$$|\mathbf{1}| = \{*\} \qquad \mathrm{P}(\mathbf{1}) = [0,1]$$

$$|\mathbf{Bool}| = \{t, f\} \qquad \mathrm{P}(\mathbf{Bool}) = \{(p,q) \ ; \ p+q \leq 1\}$$

$$|\mathbf{Nat}| = \{0,1,2,\dots\} \qquad \mathrm{P}(\mathbf{Nat}) = \{x \in [0,1]^{\mathbb{N}} \ ; \ \sum_n x_n \leq 1\}$$

$$|\mathbf{Bool} \Rightarrow \mathbf{1}| = \{[t^n, f^m] \ ; \ n, m \in \mathbb{N}\},$$

$$\mathrm{P}(\mathbf{Bool} \Rightarrow \mathbf{1}) = \{Q \in (\mathbb{R}^+)^{|\mathbf{Bool} \Rightarrow \mathbf{1}|} \ ;$$

$$\forall x_t + x_f \leq 1, \quad \sum_{m,n=0}^{\infty} Q_{[t^m, f^n]} \, x_t^m x_f^n \leq 1\}$$

Proposition: Proba. spaces as Domains

$(|X|, \mathrm{P}(X))$ is a **Proba. space iff** $\mathrm{P}(X)$ is bounded covering, **Scott Closed** (downwards-closed and dcpo) and **Convex**.

Pcoh : Linear Category

Objects: Proba. Spaces

Morphisms: Linear Functions

- **Smcc** $(\mathbf{1}, \otimes, \multimap)$
- biproduct

Call by Name $A \Rightarrow B = !A \multimap B$

- **Comonad** $(!, \mathrm{der}, \mathrm{dig})$
- **Sym. Comonoid** $(!A, \otimes, \mathbf{1})$

$\mathrm{Pcoh}_!$: Kleisli Category

Objects: Proba. Spaces

Morphisms: Analytic Functions

- **CCC**
- (PCF+coin)

Pcoh(X, Y)

Matrices $Q \in (\mathbb{R}^+)^{|X| \times |Y|}$ such that:

$$\forall x \in \mathrm{P}(X),\ Q \cdot x = \left(\sum_{a \in |X|} Q_{a,b}\, x_a \right)_b \in \mathrm{P}(Y)$$

Example

Pcoh(**Nat**, **Nat**): Stochastic Matrices $Q \in (\mathbb{R}^+)^{\mathbb{N} \times \mathbb{N}}$.

$$\forall x \in (\mathbb{R}^+)^{\mathbb{N}} ;\ \sum_{n \in \mathbb{N}} x_n \leq 1,\ \sum_{m,n \in \mathbb{N}} Q_{m,n} x_n \leq 1$$

Free Symetric Comonoid and Comonad

Exponential

$$|!X| = \mathcal{M}_{\text{fin}}(|X|) \text{ the set of finite multisets}$$

$$\mathrm{P}(!X) = \{x^! \; ; \; x \in \mathrm{P}(X)\}^{\perp\perp} \text{ where } x^!_{[a_1,\dots,a_n]} = \prod_{i=1}^n x_{a_i}$$

Example

Let $\mathbf{coin}(p) = (p, 1-p) \in \mathrm{P}(\mathbf{Bool})$.

$\mathbf{coin}(p)^!_{[]} = 1, \quad \mathbf{coin}(p)^!_{[t,t]} = p^2, \quad \mathbf{coin}(p)^!_{[t,f]} = p(1-p), \ \dots$

Theorem (2016: Crubillé - Ehrhard - Pagani - T.)

This exponential computes the free symetric comonoid.

Free Symetric Comonoid and Comonad

Exponential

$$|!X| = \mathcal{M}_{\mathsf{fin}}(|X|) \text{ the set of finite multisets}$$

$$\mathrm{P}(!X) = \{x^! \; ; \; x \in \mathrm{P}(X)\}^{\perp\perp} \text{ where } x^!_{[a_1,\dots,a_n]} = \prod_{i=1}^{n} x_{a_i}$$

Symetric Comonoid | **Comonad**

Cocontr.: $!X \xrightarrow{c^{!X}} !X \otimes !X$

Coweak.: $!X \xrightarrow{w^{!X}} \mathbf{1}$

Comult.: $\mathrm{dig}_{!X} : !!X \to !X$

Counit: $\mathrm{der}_{!X} : !X \to X$

Theorem (2016: Crubillé - Ehrhard - Pagani - T.)

This exponential computes the free symetric comonoid.

$\mathbf{Pcoh}_!(X, Y) = \mathbf{Pcoh}(!X, Y)$

Matrices $Q \in (\mathbb{R}^+)^{\mathcal{M}_{\text{fin}}(|X|)\times|Y|}$ such that

$$\forall U \in \mathrm{P}\,(!X),\;\; Q{\cdot}U = \left(\sum_{m\in\mathcal{M}_{\text{fin}}(|X|)} Q_{m,b}\, U_m\right)_b \in \mathrm{P}\,(Y)$$

Non-Linear Morphisms are **analytic** and **Scott Continuous**.

$$\mathbf{Pcoh}_!(\mathbf{Bool},\mathbf{1}) = \{Q \in (\mathbb{R}^+)^{|\mathbf{Bool}\Rightarrow\mathbf{1}|}\;\; s.t.\;\; Q_{[t^m,f^n]} \le \frac{(n+m)^{n+m}}{n^n\, m^m}\}$$

```
let rec f x =
 if x then if x then f x
                else ()
           else if x then ()
                else f x
```

is denoted as

$$\sum_{n,m=0}^{\infty} \frac{(n+m)!}{n!\,m!} x_t^{2n+1} x_f^{2m+1}$$

$\mathbf{Pcoh}_!(X, Y) = \mathbf{Pcoh}(!X, Y)$ **Density**

Matrices $Q \in (\mathbb{R}^+)^{\mathcal{M}_{\mathrm{fin}}(|X|)\times|Y|}$ such that if $x^!_m = \prod_{a\in m} x_a^{m(a)}$

$$\forall x \in \mathrm{P}(X),\ Q(x) = \left(\sum_{m\in\mathcal{M}_{\mathrm{fin}}(|X|)} Q_{m,b}\, x^!{}_m\right)_b \in \mathrm{P}(Y)$$

Non-Linear Morphisms are **analytic** and **Scott Continuous**.

$$\mathbf{Pcoh}_!(\mathbf{Bool}, \mathbf{1}) = \{Q \in (\mathbb{R}^+)^{|\mathbf{Bool}\Rightarrow\mathbf{1}|}\ s.t.\ Q_{[t^m, f^n]} \le \frac{(n+m)^{n+m}}{n^n\, m^m}\}$$

```
let rec f x =
 if x then if x then f x
                else ()
           else if x then ()
                else f x
```

is denoted as

$$\sum_{n,m=0}^{\infty} \frac{(n+m)!}{n!\, m!} x_t^{2n+1} x_f^{2m+1}$$

$\mathbf{Pcoh}_!(X, Y) = \mathbf{Pcoh}(!X, Y)$ **Density**

Matrices $Q \in (\mathbb{R}^+)^{\mathcal{M}_{\text{fin}}(|X|)\times|Y|}$ such that if $x^!_m = \prod_{a\in m} x_a^{m(a)}$

$$\forall x \in \mathrm{P}(X),\ Q(x) = \left(\sum_{m\in\mathcal{M}_{\text{fin}}(|X|)} Q_{m,b}\, {x^!}_m\right)_b \in \mathrm{P}(Y)$$

Non-Linear Morphisms are **analytic** and **Scott Continuous**.

$\mathbf{Pcoh}_!(\mathbf{Bool}, \mathbf{1}) = \{Q \in (\mathbb{R}^+)^{|\mathbf{Bool}\Rightarrow\mathbf{1}|}\ s.t.\ Q_{[t^m, f^n]} \leq \frac{(n+m)^{n+m}}{n^n\, m^m}\}$

```
let rec f x =
 if x then if x then f x
                else ()
           else if x then ()
                else f x
```

pb of DEFINABILITY

$$\sum_{n,m=0}^{\infty} \frac{(n+m)!}{n!\ m!} x_t^{2n+1} x_f^{2m+1}$$

A denotational semantics to discrete probabilistic higher-order functional computation, which is fully abstract

(based on **quantitative** semantics)

1. General framework: Domains vs Quantitative semantics
2. PCoh: Probabilistic Coherent Spaces
3. **Probabilistic Call by Name**
4. Probabilistic Call by Push Value

Probabilistic Full Abstraction

Theorem (2014: Ehrhard - Pagani - T.)

Pcoh		**pPCF**
$[\![M]\!] = [\![N]\!]$	Adequacy $\Longrightarrow$ $\Longleftarrow$ Full Abstraction	$P \simeq_o Q$ $\mathbf{Prob}(C[M], ()) \overset{\forall C[\,]}{=} \mathbf{Prob}(C[N], ())$

Adequacy Lemma (2011: Danos - Ehrhard):

If $\vdash M : \mathtt{nat}$, then $\forall n \in \mathbb{N}, [\![M]\!]_n = \mathbf{Prob}(M \to^* n)$.

Key Ingredients of Full Abstraction

- Find **testing terms** that depend only on points of the web.
- Use regularity of **analytic functions**.

How to encode a LasVegas Algorithm?

Input: A $\underline{0}/\underline{1}$ array of length $n \geq 2$ s.t. $\frac{1}{2}$ cells are $\underline{0}$.

0	1	2	3	4	5
$\underline{0}$	$\underline{1}$	$\underline{0}$	$\underline{1}$	$\underline{1}$	$\underline{0}$

$$f: \begin{array}{lcl} 0,2,5 & \mapsto & \underline{0} \\ 1,3,4 & \mapsto & \underline{1} \end{array}$$

Output: Find the index of a cell containing $\underline{0}$.

Caml:

```
let rec LasVegas (f: nat -> nat) (n:nat) =
              let k = random n in
                  if (f k = 0) then k
                  else LasVegas f n
```

pPCF: CBN

$$\textbf{fix}\Big(\lambda \mathtt{LasVegas}^{(\mathtt{nat}\Rightarrow\mathtt{nat})\Rightarrow\mathtt{nat}\Rightarrow\mathtt{nat}}\ \lambda \mathtt{f}^{\mathtt{nat}\Rightarrow\mathtt{nat}} \lambda \mathtt{n}^{\mathtt{nat}}$$
$$(\lambda \mathtt{k}^{\mathtt{nat}}\ \mathtt{if}\ (\mathtt{f\ k} = \underline{0})\ \mathtt{then}\ \mathtt{k}$$
$$\mathtt{else}\ \mathtt{LasVegas\ f\ n})\ (\mathtt{rand\ n})$$

How to encode a LasVegas Algorithm?

Input: A $\underline{0}/\underline{1}$ array of length $n \geq 2$ s.t. $\frac{1}{2}$ cells are $\underline{0}$.

0	1	2	3	4	5
$\underline{0}$	$\underline{1}$	$\underline{0}$	$\underline{1}$	$\underline{1}$	$\underline{0}$

$$f: \begin{array}{lcl} 0,2,5 & \mapsto & \underline{0} \\ 1,3,4 & \mapsto & \underline{1} \end{array}$$

Output: Find the index of a cell containing $\underline{0}$.

Caml:
let in
CBV

```
let rec LasVegas (f: nat -> nat) (n:nat) =
            let k = random n in
                if (f k = 0) then k
                else LasVegas f n
```

pPCF:
CBN

$$\mathbf{fix}\left(\lambda \mathtt{LasVegas}^{(\mathtt{nat}\Rightarrow\mathtt{nat})\Rightarrow\mathtt{nat}\Rightarrow\mathtt{nat}}\ \lambda \mathtt{f}^{\mathtt{nat}\Rightarrow\mathtt{nat}} \lambda \mathtt{n}^{\mathtt{nat}}\right.$$
$$(\lambda \mathtt{k}^{\mathtt{nat}}\ \mathtt{if}\ (\mathtt{f}\ \mathtt{k} = \underline{0})\ \mathtt{then}\ \mathtt{k}$$
$$\mathtt{else}\ \mathtt{LasVegas}\ \mathtt{f}\ \mathtt{n})\ (\mathtt{rand}\ \mathtt{n})$$

Semantics gives the answer

Storage Operator

```
let k = rand n in if k = 0 then 42 else k
```

Integer in Pcoh: $[\![\mathtt{nat}]\!] = (\mathbb{N}, \mathrm{P}(\mathtt{nat}) = \{(\lambda_n) \mid \sum_n \lambda_n \leq 1\})$

Equipped with a coalgebraic structure in the *linear* Pcoh:

- Cocontraction: $c^{\mathtt{nat}} : \mathtt{nat} \to \mathtt{nat} \otimes \mathtt{nat}$
- Coweakening: $w^{\mathtt{nat}} : \mathtt{nat} \to \mathbf{1}$

Bibliography

1990 Krivine, Opérateurs de mise en mémoire et Traduction.

1999 Levy, Call by Push Value, a subsuming paradigm.

2000 Nour, On Storage operator.

2016 Curien, Fiore, Munch-Maccagnoni, A Theory of Effects and Resources .

What sem. object to encode Storage Operator.

The Eilenberg Moore Category: $\mathbf{Pcoh}^!$

Coalgebras $P = (\underline{P}, h_P)$ with $\underline{P} \in \mathbf{Pcoh}$ and $h_P \in \mathbf{Pcoh}(\underline{P}, !\underline{P})$:

$$\begin{array}{ccc} \underline{P} & \xrightarrow{h_P} & !\underline{P} \\ & \searrow^{\mathrm{Id}} & \downarrow \mathrm{der}_{\underline{P}} \\ & & \underline{P} \end{array} \qquad\qquad \begin{array}{ccc} \underline{P} & \xrightarrow{h_P} & !\underline{P} \\ {\scriptstyle h_P}\downarrow & & \downarrow \mathrm{dig}_{\underline{P}} \\ !\underline{P} & \xrightarrow{!h_P} & !!\underline{P} \end{array}$$

Coalgebras have a comonoid structure: values can be **stored**.

Types interpreted as coalgebras:

$!X$ by def. of the exp. | $\otimes$, $\oplus$ and fix preserve coalgebras.

Example

Stream: $\mathsf{S}_\phi = \phi \otimes !\mathsf{S}_\phi$ | **List:** $\lambda_0 = \mathbf{1} \oplus (\phi \otimes \lambda_0)$

A denotational semantics to discrete probabilistic higher-order functional computation, which is fully abstract

(based on **quantitative** semantics)

1. General framework: Domains vs Quantitative semantics
2. PCoh: Probabilistic Coherent Spaces
3. Probabilistic Call by Name
4. **Probabilistic Call by Push Value**

Probabilistic Call By Push Value

Types:

(positive) $\phi, \psi, \ldots := \mathbf{1} \mid !\sigma \mid \phi \otimes \psi \mid \phi \oplus \psi \mid \alpha \mid \mathsf{Fix}\, \alpha \cdot \phi$

(general) $\sigma, \tau \ldots := \phi \mid \phi \multimap \sigma$

Values: (positive type)

$V, W \ldots := x \mid () \mid M^! \mid (V, W) \mid \mathsf{in}_1 V \mid \mathsf{in}_2 V \mid \mathsf{fold}(V)\,.$

Programs: (general type)

$$M, N \ldots := x \mid () \mid M^! \mid \mathsf{der}(M) \mid \lambda x^{\phi}\, M \mid \langle M \rangle N \mid \mathsf{fix}\, x^{!\sigma}\, M$$
$$\mid \mathsf{in}_1 M \mid \mathsf{in}_2 M \mid \mathsf{case}(M, x_1 \cdot N_1, x_2 \cdot N_2)$$
$$\mid (M, N) \mid \mathsf{pr}_1 M \mid \mathsf{pr}_2 M \mid \mathsf{fold}(M) \mid \mathsf{unfold}(M)$$
$$\mid \mathsf{coin}(p),\ p \in [0,1] \cap \mathbb{Q}$$

Typing context: $\mathcal{P} = (x_1 : \phi_1, \ldots, x_k : \phi_k)$

Probabilistic Call By Push Value

Types: $\sigma \Rightarrow \tau = !\sigma \multimap \tau$

(positive) $\phi, \psi, \ldots := \mathbf{1} \mid !\sigma \mid \phi \otimes \psi \mid \phi \oplus \psi \mid \alpha \mid \mathsf{Fix}\,\alpha \cdot \phi$

(general) $\sigma, \tau \ldots := \phi \mid \phi \multimap \sigma$

Values:

$$V, W \ldots := x \mid () \mid M^! \mid (V, W) \mid \mathsf{in}_1 V \mid \mathsf{in}_2 V \mid \mathsf{fold}(V).$$

Programs: $\lambda x^{\sigma}\, M = \lambda x^{!\sigma}\, M$ and $(M)N = \langle M \rangle N^!$

$$\begin{aligned} M, N \ldots := {} & x \mid () \mid M^! \mid \mathsf{der}(M) \mid \lambda x^{\phi}\, M \mid \langle M \rangle N \mid \mathsf{fix}\, x^{!\sigma}\, M \\ & \mid \mathsf{in}_1 M \mid \mathsf{in}_2 M \mid \mathsf{case}(M, x_1 \cdot N_1, x_2 \cdot N_2) \\ & \mid (M, N) \mid \mathsf{pr}_1 M \mid \mathsf{pr}_2 M \mid \mathsf{fold}(M) \mid \mathsf{unfold}(M) \\ & \mid \mathsf{coin}(p),\ p \in [0,1] \cap \mathbb{Q} \end{aligned}$$

Typing context: $\mathcal{P} = (x_1 : \phi_1, \ldots, x_k : \phi_k)$

Types: $\mathtt{nat} = \mathsf{Fix}\,\alpha \cdot \mathbf{1} \oplus \alpha$ and $\sigma \Rightarrow \tau = !\sigma \multimap \tau$

(positive) $\phi, \psi, \ldots := \mathbf{1} \mid !\sigma \mid \phi \otimes \psi \mid \phi \oplus \psi \mid \alpha \mid \mathsf{Fix}\,\alpha \cdot \phi$

(general) $\sigma, \tau \ldots := \phi \mid \phi \multimap \sigma$

Values: $\underline{0} = \mathsf{in}_1()$ and $\underline{n+1} = \mathsf{in}_2\underline{n}$

$$V, W \ldots := x \mid () \mid M^! \mid (V, W) \mid \mathsf{in}_1 V \mid \mathsf{in}_2 V \mid \mathsf{fold}(V).$$

Programs: $\mathbf{succ}(M) = \mathsf{in}_2 M$

$$M, N \ldots := x \mid () \mid M^! \mid \mathsf{der}(M) \mid \lambda x^{\phi}\, M \mid \langle M \rangle N \mid \mathsf{fix}\, x^{!\sigma}\, M$$
$$\mid \mathsf{in}_1 M \mid \mathsf{in}_2 M \mid \mathsf{case}(M, x_1 \cdot N_1, x_2 \cdot N_2)$$
$$\mid (M, N) \mid \mathsf{pr}_1 M \mid \mathsf{pr}_2 M \mid \mathsf{fold}(M) \mid \mathsf{unfold}(M)$$
$$\mid \mathsf{coin}(p),\; p \in [0,1] \cap \mathbb{Q}$$

Typing context: $\mathcal{P} = (x_1 : \phi_1, \ldots, x_k : \phi_k)$

Probabilistic Call By Push Value

Types: $\mathtt{nat} = \mathsf{Fix}\,\alpha \cdot \mathbf{1} \oplus \alpha$ and $\sigma \Rightarrow \tau = !\sigma \multimap \tau$

(positive) $\phi, \psi, \ldots := \mathbf{1} \mid !\sigma \mid \phi \otimes \psi \mid \phi \oplus \psi \mid \alpha \mid \mathsf{Fix}\,\alpha \cdot \phi$

(general) $\sigma, \tau \ldots := \phi \mid \phi \multimap \sigma$

Values: $\underline{0} = \mathsf{in}_1()$ and $\underline{n+1} = \mathsf{in}_2\,\underline{n}$

$V, W \ldots := x \mid () \mid M^! \mid (V, W) \mid \mathsf{in}_1 V \mid \mathsf{in}_2 V \mid \mathsf{fold}(V).$

Programs: $\mathbf{let}\ x = M\ \mathbf{in}\ N = \mathsf{case}(M, y \cdot N[\underline{0}/x], z \cdot N[\mathbf{succ}(z)/x])$

$M, N \ldots := x \mid () \mid M^! \mid \mathsf{der}(M) \mid \lambda x^{\phi}\, M \mid \langle M \rangle N \mid \mathsf{fix}\, x^{!\sigma}\, M$
$\mid \mathsf{in}_1 M \mid \mathsf{in}_2 M \mid \mathsf{case}(M, x_1 \cdot N_1, x_2 \cdot N_2)$
$\mid (M, N) \mid \mathsf{pr}_1 M \mid \mathsf{pr}_2 M \mid \mathsf{fold}(M) \mid \mathsf{unfold}(M)$
$\mid \mathsf{coin}(p),\ p \in [0,1] \cap \mathbb{Q}$

Typing context: $\mathcal{P} = (x_1 : \phi_1, \ldots, x_k : \phi_k)$

Probabilistic Full Abstraction

Theorem (2016: Ehrhard - T.)

Pcoh		**pCBPV**
$[\![M]\!] = [\![N]\!]$	Adequacy $\Longrightarrow$ $\Longleftarrow$ Full Abstraction	$P \simeq_o Q$
		$\mathbf{Prob}(C[M],()) \overset{\forall C[\,]}{=} \mathbf{Prob}(C[N],())$

Key Ingredients of Adequacy:

- Handle **values** separately
- Logical relations: **fixpoint** of types (hidden step indexing, biorthogonality closure, fixpoints of pairs of logical relations)
- **Density:** Morphisms on positive types are characterized by their action on coalgebric points.

Key Ingredients of Full Abstraction:

- Find **testing terms** that depend only on points of the web.
- Use regularity of **analytic functions** and **density**

pCBPV a funct. lang. suitable for writing proba. programs

- Combines CBN and CBV
- Values are of positive Types, they can be duplicated or erased accordingly to the comonoidal structure of their interpretation
- Programs can handle base types such as streams
- Full abstraction (with no quotient needed)

Further directions:

- Combine Non-determinism and Probability by constructing a non-deterministic monad on $\mathbf{Pcoh}_!$.
- Move up from discrete to continuous probability.