
Probabilistic Lambda Calculi

Research Project

Pedro Amorim and Eric Jackson

1 Abstract

In section 2, we introduce a probabilistic version of Call-By-Push Value. In section 3, we introduce two probabilistic lambda-calculi, λ_{IID} and λ_{PC} , which correspond to independent and identically distributed sampling and perfectly correlated sampling, respectively.

2 Probabilistic Call-By-Push-Value

2.1 Syntax

Types

CBPV types are given by $\tau = A \mid B$ where A and B are defined inductively according to the following rules:

$$\begin{aligned} A &::= U\underline{B} \mid A + A \mid A \times A \mid 1 \mid \mathbb{R} \\ \underline{B} &::= F A \mid \underline{B} \times \underline{B} \mid A \rightarrow \underline{B} \end{aligned}$$

As discussed in Levy's thesis, A can be thought of as defining values and B computations.

Expressions

$$\begin{aligned} V, M &::= x \\ &\mid () \\ &\mid \lambda x. M \\ &\mid \text{let } x \text{ be } V. M \\ &\mid V' M \\ &\mid \text{produce } V \\ &\mid M \text{ to } x. N \\ &\mid \text{thunk } M \\ &\mid \text{force } V \\ &\mid (M, M) \\ &\mid \#1 M \\ &\mid \#2 M \\ &\mid \text{pm } V \text{ as } (x, y). M \\ &\mid \text{inl}_{A+A} V \\ &\mid \text{inr}_{A+A} V \\ &\mid \text{pm } V \text{ as } \{x.M, y.M\} \\ &\mid \text{coin} \\ &\mid \text{rand} \end{aligned}$$

2.2 Static Semantics

Unit

$$\overline{\Gamma \vdash^v () : 1}$$

Functions and Application

$$\frac{\Gamma_{x:A} \vdash^v M : \underline{B}}{\Gamma \vdash^c \lambda x. M : A \rightarrow \underline{B}}$$

$$\frac{\Gamma \vdash^v V : A \quad \Gamma_{V:A} \vdash^c M : A \rightarrow \underline{B}}{\Gamma \vdash^c V M : \underline{B}}$$

$$\frac{\Gamma \vdash^v V : A \quad \Gamma_{V:A} \vdash^c M : A \rightarrow \underline{B}}{\Gamma \vdash^c \text{let } x \text{ be } V. M : \underline{B}}$$

Produce and To

$$\frac{\Gamma \vdash^v V : A}{\Gamma \vdash^c \text{produce } V : FA}$$

$$\frac{\Gamma \vdash^c M : FA \quad \Gamma_{x:A} \vdash^c N : \underline{B}}{\Gamma \vdash^c M \text{ to } x. N : \underline{B}}$$

Thunk and Force

$$\frac{\Gamma \vdash^c M : \underline{B}}{\Gamma \vdash^v \text{thunk } M : U\underline{B}}$$

$$\frac{\Gamma \vdash^v V : U\underline{B}}{\Gamma \vdash^c \text{force } V : \underline{B}}$$

2.3 Denotational Semantics

Products

$$\frac{\Gamma \vdash^c M : \underline{B} \quad \Gamma \vdash^c M' : \underline{B'}}{\Gamma \vdash^c (M, M') : \underline{B} \times \underline{B'}}$$

$$\frac{\Gamma \vdash^c M : \underline{B} \times \underline{B'}}{\Gamma \vdash^c \#1 M : \underline{B}}$$

$$\frac{\Gamma \vdash^c M : \underline{B} \times \underline{B'}}{\Gamma \vdash^c \#1 M : \underline{B'}}$$

$$\frac{\Gamma \vdash^c (M, M') : \underline{B} \times \underline{B'} \quad \Gamma_{x:\underline{B}, y:\underline{B'}} \vdash^c M : \underline{B}}{\Gamma \vdash^c \text{pm } V \text{ as } (x, y). M : \underline{B}}$$

Sums

$$\frac{\Gamma \vdash^v V : A}{\Gamma \vdash^v \text{inl}_{A+A'} V : \tau_1 + \tau_2}$$

$$\frac{\Gamma \vdash^v V : A}{\Gamma \vdash^v \text{inr}_{A+A'} V : \tau_1 + \tau_2}$$

$$\frac{\Gamma \vdash^v V : A + A' \quad \Gamma_{x:A, y:A'} \vdash^c M : \underline{B}}{\Gamma \vdash^c \text{pm } V \text{ as } \{x.M, y.M\} : \underline{B}}$$

Random Variables

$$\overline{\Gamma \vdash^c \text{coin} : F\mathbb{R}}$$

$$\overline{\Gamma \vdash^c \text{rand} : F\mathbb{R}}$$

3 λ_{IID} and λ_{PC}

3.1 Syntax

Types

The types of λ_{IID} and λ_{PC} are defined according to the following rules

$$\tau ::= \text{unit} \mid \mathbb{R} \mid \tau \rightarrow \tau \mid \tau + \tau \mid \tau \times \tau$$

Expressions

The expressions of λ_{IID} and λ_{PC} of the following form

$$\begin{aligned} e ::= & x \\ & | () \\ & | \lambda x : \tau. e \\ & | \text{let } x = e \text{ in } e \\ & | e e \\ & | \text{coin} \\ & | \text{rand} \\ & | \text{inl}_{\tau_1 + \tau_2} e \\ & | \text{inr}_{\tau_1 + \tau_2} e \\ & | \text{case } e \text{ of } e \mid e \\ & | (e, e) \\ & | \#1 e \\ & | \#2 e \\ & | e \text{ to } x \text{ in } e \end{aligned}$$

3.2 Static Semantics

Unit

$$\frac{}{\Gamma \vdash () : \text{unit}}$$

Functions and Application

$$\frac{\Gamma_{x:\tau} \vdash e : \tau'}{\Gamma \vdash \lambda x : \tau. e : \tau \rightarrow \tau'}$$

$$\frac{\Gamma \vdash e_1 : \tau' \quad \Gamma_{x:\tau'} \vdash e_2 : \tau}{\Gamma \vdash \text{let } x = e_1 \text{ in } e_2 : \tau}$$

$$\frac{\Gamma \vdash e_1 : \tau \rightarrow \tau' \quad \Gamma \vdash e_2 : \tau}{\Gamma \vdash e_1 e_2 : \tau'}$$

To

$$\frac{\Gamma \vdash e_1 : \tau' \quad \Gamma_{x:\tau'} \vdash e_2 : \tau}{\Gamma \vdash e_1 \text{ to } x \text{ in } e_2 : \tau}$$

Products

$$\frac{\Gamma \vdash e_1 : \tau_1 \quad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash (e_1, e_2) : \tau_1 \times \tau_2}$$

$$\frac{\Gamma \vdash e : \tau_1 \times \tau_2}{\Gamma \vdash \#1 e : \tau_1}$$

$$\frac{\Gamma \vdash e : \tau_1 \times \tau_2}{\Gamma \vdash \#2 e : \tau_2}$$

Sums

$$\frac{\Gamma \vdash e : \tau_1}{\Gamma \vdash \text{inl}_{\tau_1 + \tau_2} e : \tau_1 + \tau_2}$$

$$\frac{\Gamma \vdash e : \tau_2}{\Gamma \vdash \text{inr}_{\tau_1 + \tau_2} e : \tau_1 + \tau_2}$$

$$\frac{\Gamma \vdash e_1 : \tau_1 + \tau_2 \quad \Gamma \vdash e_2 : \tau_1 \rightarrow \tau \quad \Gamma \vdash e_3 : \tau_2 \rightarrow \tau}{\Gamma \vdash \text{case } e_1 \text{ of } e_2 \mid e_3 : \tau}$$

Random Variables

$$\frac{}{\Gamma \vdash \text{coin} : \mathbb{R}}$$

$$\frac{}{\Gamma \vdash \text{rand} : \mathbb{R}}$$

4 Translating λ_{IID} and λ_{PC} to CBPV

Rather than defining denotational semantics for λ_{IID} and λ_{PC} , we will define a translation from each of these languages into CBPV. Then, we can use the CBPV semantics to generate denotational semantics.

4.1 Types

We begin with defining type translations from λ_{IID} and λ_{PC} to CBPV. These translations are essentially identical to the translations of CBN and CBV to CBPV presented in Levy's thesis.

λ_{IID}	λ_{PC}
$\llbracket \text{unit} \rrbracket_{\text{IID}} \triangleq F1$	$\llbracket \text{unit} \rrbracket_{\text{PC}} \triangleq 1$
$\llbracket \mathbb{R} \rrbracket_{\text{IID}} \triangleq F\mathbb{R}$	$\llbracket \mathbb{R} \rrbracket_{\text{PC}} \triangleq \mathbb{R}$
$\llbracket \tau_1 \rightarrow \tau_2 \rrbracket_{\text{IID}} \triangleq (U\llbracket \tau_1 \rrbracket_{\text{IID}}) \rightarrow \llbracket \tau_2 \rrbracket_{\text{IID}}$	$\llbracket \tau_1 \rightarrow \tau_2 \rrbracket_{\text{PC}} \triangleq U(\llbracket \tau_1 \rrbracket_{\text{PC}} \rightarrow F\llbracket \tau_2 \rrbracket_{\text{PC}})$
$\llbracket \tau_1 + \tau_2 \rrbracket_{\text{IID}} \triangleq F(U\llbracket \tau_1 \rrbracket_{\text{IID}} + U\llbracket \tau_2 \rrbracket_{\text{IID}})$	$\llbracket \tau_1 + \tau_2 \rrbracket_{\text{PC}} \triangleq U\llbracket \tau_1 \rrbracket_{\text{PC}} + U\llbracket \tau_2 \rrbracket_{\text{PC}}$
$\llbracket \tau_1 \times \tau_2 \rrbracket_{\text{IID}} \triangleq \llbracket \tau_1 \rrbracket_{\text{IID}} \times \llbracket \tau_2 \rrbracket_{\text{IID}}$	$\llbracket \tau_1 \times \tau_2 \rrbracket_{\text{PC}} \triangleq \llbracket \tau_1 \rrbracket_{\text{PC}} \times \llbracket \tau_2 \rrbracket_{\text{PC}}$

4.2 Expressions

Now, we define translations from expressions in IID and PC to CBPV.

IID
$\llbracket x \rrbracket_{\text{IID}} \triangleq \text{force } x$
$\llbracket \lambda x. e \rrbracket_{\text{IID}} \triangleq \lambda x. \llbracket e \rrbracket_{\text{IID}}$
$\llbracket \text{let } x = e_1 \text{ in } e_2 \rrbracket_{\text{IID}} \triangleq \text{let } x \text{ be thunk } \llbracket e_1 \rrbracket_{\text{IID}}. \llbracket e_2 \rrbracket_{\text{IID}}$
$\llbracket e_1 e_2 \rrbracket_{\text{IID}} \triangleq (\text{thunk } \llbracket e_2 \rrbracket_{\text{IID}})' \llbracket e_1 \rrbracket_{\text{IID}}$
$\llbracket \text{coin} \rrbracket_{\text{IID}} \triangleq \text{coin}$
$\llbracket \text{rand} \rrbracket_{\text{IID}} \triangleq \text{rand}$
$\llbracket \text{inl}_{\tau_1 + \tau_2} e \rrbracket_{\text{IID}} \triangleq \text{produce inl thunk } \llbracket e \rrbracket_{\text{IID}}$
$\llbracket \text{inr}_{\tau_1 + \tau_2} e \rrbracket_{\text{IID}} \triangleq \text{produce inr thunk } \llbracket e \rrbracket_{\text{IID}}$
$\llbracket \text{case } e_1 \text{ of } e_2 \mid e_3 \rrbracket_{\text{IID}} \triangleq \llbracket e_1 \rrbracket_{\text{IID}} \text{ to } z. \text{ pm } z \text{ as } \{\text{inl } x. \llbracket e_2 \rrbracket_{\text{IID}}, \text{inr } x. \llbracket e_3 \rrbracket_{\text{IID}}\}$
$\llbracket (e_1, e_2) \rrbracket_{\text{IID}} \triangleq$
$\llbracket \#1 e \rrbracket_{\text{IID}} \triangleq$
$\llbracket \#2 e \rrbracket_{\text{IID}} \triangleq$
$\llbracket e_1 \text{ to } x \text{ in } e_2 \rrbracket_{\text{IID}} \triangleq \text{let } x = \llbracket e_1 \rrbracket_{\text{PC}} \text{ in } \llbracket e_2 \rrbracket_{\text{IID}}$

$$\begin{aligned}
\llbracket () \rrbracket_{\text{PC}} &\triangleq \text{produce } () \\
\llbracket x \rrbracket_{\text{PC}} &\triangleq \text{produce } x \\
\llbracket \lambda x. e \rrbracket_{\text{PC}} &\triangleq \text{produce thunk } \lambda x. \llbracket e \rrbracket_{\text{PC}} \\
\llbracket \text{let } x = e_1 \text{ in } e_2 \rrbracket_{\text{PC}} &\triangleq \llbracket e_1 \rrbracket_{\text{PC}} \text{ to } x. \llbracket e_2 \rrbracket_{\text{PC}} \\
\llbracket e_1 e_2 \rrbracket_{\text{PC}} &\triangleq \llbracket e_2 \rrbracket_{\text{PC}} \text{ to } x. \llbracket e_1 \rrbracket_{\text{PC}} \text{ to } f. x'(\text{force } f) \\
\llbracket \text{coin} \rrbracket_{\text{PC}} &\triangleq \text{produce coin} \\
\llbracket \text{rand} \rrbracket_{\text{PC}} &\triangleq \text{produce rand} \\
\llbracket \text{inl}_{\tau_1 + \tau_2} e \rrbracket_{\text{PC}} &\triangleq \llbracket e \rrbracket_{\text{PC}} \text{ to } z. \text{produce inl } z \\
\llbracket \text{inr}_{\tau_1 + \tau_2} e \rrbracket_{\text{PC}} &\triangleq \llbracket e \rrbracket_{\text{PC}} \text{ to } z. \text{produce inr } z \\
\llbracket \text{case } e_1 \text{ of } e_2 \mid e_3 \rrbracket_{\text{PC}} &\triangleq \llbracket e_1 \rrbracket_{\text{PC}} \text{ to } z. \text{pm } z \text{ as } \{\text{inl } x. \llbracket e_2 \rrbracket_{\text{PC}}, \text{inl } y, \llbracket e_3 \rrbracket_{\text{PC}}\} \\
\llbracket (e_1, e_2) \rrbracket_{\text{PC}} &\triangleq \text{produce } (\llbracket e_1 \rrbracket_{\text{PC}}, \llbracket e_2 \rrbracket_{\text{PC}}) \\
\llbracket \#1 e \rrbracket_{\text{PC}} &\triangleq \text{pm } \llbracket e \rrbracket_{\text{PC}} \text{ as } (x, y). \text{produce } x \\
\llbracket \#2 e \rrbracket_{\text{PC}} &\triangleq \text{pm } \llbracket e \rrbracket_{\text{PC}} \text{ as } (x, y). \text{produce } y \\
\llbracket e_1 \text{ to } x \text{ in } e_2 \rrbracket_{\text{PC}} &\triangleq
\end{aligned}$$

4.3 Contexts

IID

PC

Theorem.

- If $\Gamma \vdash_{\text{IID}} e : \tau$ then $\llbracket \Gamma \rrbracket_{\text{IID}} \vdash_c \llbracket e \rrbracket_{\text{IID}} : \llbracket \tau \rrbracket_{\text{IID}}$.
- If $\Gamma \vdash_{\text{PC}} e : \tau$ then $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket e \rrbracket_{\text{PC}} : F\llbracket \tau \rrbracket_{\text{PC}}$.

Proof.

We proceed by mutual induction on $\Gamma \vdash_{\text{IID}} e : \tau$ and $\Gamma \vdash_{\text{PC}} e : \tau$.

- $\Gamma \vdash_{\text{PC}} () : \text{unit}$. We would like to show $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket () \rrbracket_{\text{PC}} : F\llbracket \text{unit} \rrbracket_{\text{PC}}$. By the expression and type translation rules, this is equal to $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \text{produce } () : F1$ which holds by the CBPV produce and unit type rules.
- $\Gamma \vdash_{\text{PC}} x : \tau$. We would like to show $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket x \rrbracket_{\text{PC}} : F\llbracket \tau \rrbracket_{\text{PC}}$. By the expression and type translation rules, this is equal to $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \text{produce } x : F$. By inspection $\Gamma(x) = \tau$, so by the context rule $\llbracket \Gamma \rrbracket_{\text{PC}}(x) = \llbracket \tau \rrbracket_{\text{PC}}$. Thus, this typing holds.
- $\Gamma \vdash_{\text{PC}} \lambda x : \tau. e : \tau \rightarrow \tau'$. We have

$$\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket \lambda x : \tau. e \rrbracket_{\text{PC}} : F\llbracket \tau \rightarrow \tau' \rrbracket_{\text{PC}} = \llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \text{produce thunk } \lambda x : \llbracket \tau \rrbracket_{\text{PC}}. \llbracket e \rrbracket_{\text{PC}} : FU(\llbracket \tau \rrbracket_{\text{PC}} \rightarrow F\llbracket \tau' \rrbracket_{\text{PC}})$$

By the inductive hypothesis, $\llbracket \Gamma_{x:\tau} \rrbracket_{\text{PC}} \vdash_c \llbracket e \rrbracket_{\text{PC}} : \llbracket \tau' \rrbracket_{\text{PC}}$, thus this typing is valid.

- $\Gamma \vdash_{\text{PC}} \text{let } x = e_1 \text{ in } e_2 : \tau$. We have

$$\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket \text{let } x = e_1 \text{ in } e_2 \rrbracket_{\text{PC}} : F\llbracket \tau \rrbracket_{\text{PC}} = \llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket e_1 \rrbracket_{\text{PC}} \text{ to } x. \llbracket e_2 \rrbracket_{\text{PC}} : F\llbracket \tau' \rrbracket_{\text{PC}}$$

By inspection, $\Gamma \vdash_{\text{PC}} e_1 : \tau'$ and $\Gamma_{x:\tau'} \vdash_{\text{PC}} e_2 : \tau$. Thus by the inductive hypothesis $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_{\text{PC}} \llbracket e_1 \rrbracket_{\text{PC}} : F\llbracket \tau' \rrbracket_{\text{PC}}$ and $\llbracket \Gamma_{x:\tau'} \rrbracket_{\text{PC}} \vdash_{\text{PC}} \llbracket e_2 \rrbracket_{\text{PC}} : F\llbracket \tau \rrbracket_{\text{PC}}$. Thus, by the CVPB to rule, this typing holds.

- $\Gamma \vdash_{\text{PC}} e_1 e_2 : \tau$. We have

$$\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket e_1 e_2 \rrbracket_{\text{PC}} : F\llbracket \tau \rrbracket_{\text{PC}} = \llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket e_2 \rrbracket_{\text{PC}} \text{ to } x. \llbracket e_1 \rrbracket_{\text{PC}} \text{ to } f. x'(\text{force } f) : F\llbracket \tau' \rrbracket_{\text{PC}}$$

By inspection, $\Gamma \vdash_{\text{PC}} e_1 : \tau \rightarrow \tau'$ and $\Gamma \vdash_{\text{PC}} e_2 : \tau$. By the inductive hypothesis, $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket e_1 \rrbracket_{\text{PC}} : F\llbracket \tau \rightarrow \tau' \rrbracket_{\text{PC}}$ and $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket e_2 \rrbracket_{\text{PC}} : F\llbracket \tau \rrbracket_{\text{PC}}$. Checking the to and force type rules allows us to verify that this typing is valid.

- $\Gamma \vdash_{\text{PC}} \text{coin} : \mathbb{R}$.

$$\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket \text{coin} \rrbracket_{\text{PC}} : F\llbracket \mathbb{R} \rrbracket_{\text{PC}} = \llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \text{produce coin} : F\mathbb{R}$$

which clearly holds by the CBPV typing rules for coin and produce.

- $\Gamma \vdash_{\text{PC}} \text{rand} : \mathbb{R}$.

$$\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket \text{rand} \rrbracket_{\text{PC}} : F\llbracket \mathbb{R} \rrbracket_{\text{PC}} = \llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \text{produce rand} : F\mathbb{R}$$

which clearly holds by the CBPV typing rules for coin and produce.

- $\Gamma \vdash_{\text{PC}} (e_1, e_2) : \tau_1 \times \tau_2$.

$$\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \llbracket (e_1, e_2) \rrbracket_{\text{PC}} : F\llbracket \tau_1 \times \tau_2 \rrbracket_{\text{PC}} = \llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c \text{produce}(\llbracket e_1 \rrbracket_{\text{PC}}, \llbracket e_2 \rrbracket_{\text{PC}}) : FU(F\llbracket \tau_1 \rrbracket_{\text{PC}}, F\llbracket \tau_2 \rrbracket_{\text{PC}})$$

By the inductive hypothesis, $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c e_1 : F\llbracket \tau_1 \rrbracket_{\text{PC}}$ and $\llbracket \Gamma \rrbracket_{\text{PC}} \vdash_c e_2 : F\llbracket \tau_2 \rrbracket_{\text{PC}}$. Thus, by the CBPV typing rules, this typing holds.

QED

Theorem. *If $\Gamma \vdash_{\text{IID}} e : \tau_1 \times \tau_2$ then $\llbracket e \rrbracket = \mu_1 \times \mu_2$.*

Proof. TODO!

QED

5 Potential Applications

5.1 System Security

5.2 Key Reuse

5.3 Pseudo-Number Generators

5.4 Random Variables