Higher-Order Probabilistic Programming
A Tutorial at POPL 2019

Part II

Ugo Dal Lago
(Based on joint work with Flavien Breuvart, Raphaëlle Crubillé, Charles Grellois, Davide Sangiorgi, . . . )

POPL 2019, Lisbon, January 14th
We are interested in a better understanding of some crucial questions about higher-order probabilistic programs, e.g.:

- How could we *formalise* and *prove* programs to have certain desirable *properties*, like being terminating or consuming a bounded amount of resources?
- How could we prove programs to be *equivalent*, or more generally to be in a certain relation?

We could in principle answer these questions directly in a programming language like OCAML.

It is methodologically much better to distill some paradigmatic calculi which expose all the essential features, but which are somehow agnostic to many unimportant details.

We will introduce and study two such calculi:

- **PCF** ⊕, a calculus for randomized higher-order programming.
- **PCF** sample, score, a calculus for bayesian programming.
We are interested in a better understanding of some crucial questions about higher-order probabilistic programs, e.g.:

- How could we formalise and prove programs to have certain desirable properties, like being terminating or consuming a bounded amount of resources?
- How could we prove programs to be equivalent, or more generally to be in a certain relation?

We could in principle answer these questions directly in a programming language like OCAML.
We are interested in a better understanding of some crucial questions about higher-order probabilistic programs, e.g.:

- How could we **formalise** and **prove** programs to have certain desirable **properties**, like being terminating or consuming a bounded amount of resources?
- How could we prove programs to be **equivalent**, or more generally to be in a certain relation?

We could in principle answer these questions directly in a programming language like **OCAML**.

It is methodologically much better to distill some paradigmatic calculi which expose all the essential features, but which are somehow agnostic to many unimportant details.

We will introduce and study two such calculi:

- \( \text{PCF} \oplus \), a calculus for randomized higher-order programming.
- \( \text{PCF}_{\text{sample, score}} \), a calculus for bayesian programming.
PCF$_\oplus$: Types, Terms, Values

**Types**  \( \tau, \rho ::= \text{UNIT} \mid \text{NUM} \mid \tau \to \rho \)

**Terms**  \( M, N ::= V \mid V \ W \mid \text{let} \ M = x \ \text{in} \ N \mid M \oplus N \mid \text{if} \ V \ \text{then} \ M \ \text{else} \ N \mid f_n(V_1,\ldots,V_n) \)

**Values**  \( V, W ::= \ast \mid x \mid r \mid \lambda x.M \mid \text{fix} \ x.V \)
PCF⊕: Type Assignment Rules

Value Typing Rules

\[ \Gamma \vdash \star : \text{UNIT} \quad S \quad \Gamma, x : \tau \vdash x : \tau \quad V \quad \Gamma \vdash r : \text{NUM} \quad R \]
\[ \Gamma, x : \tau \vdash M : \rho \quad \lambda \quad \Gamma, x : \tau \rightarrow \rho \vdash M : \tau \rightarrow \rho \]
\[ \Gamma \vdash \text{fix} \ x. M : \tau \rightarrow \rho \quad X \]

Term Typing Rules

\[ \Gamma \vdash V : \tau \rightarrow \rho \quad \Gamma \vdash W : \tau \quad \Theta \quad \Gamma \vdash M : \tau \quad \Gamma, x : \tau \vdash N : \rho \quad \Gamma \vdash \text{let} \ M = x \ \text{in} \ N : \rho \quad \Lambda \quad \Gamma \vdash M : \tau \quad \Gamma \vdash N : \tau \quad \Gamma \vdash M \oplus N : \tau \quad \oplus \]
\[ \Gamma \vdash V : \text{NUM} \quad \Gamma \vdash M : \tau \quad \Gamma \vdash N : \tau \quad \Gamma \vdash \text{if} \ V \ \text{then} \ M \ \text{else} \ N : \tau \quad I \quad \Gamma \vdash V_1 : \text{NUM} \quad \cdots \quad \Gamma \vdash V_n : \text{NUM} \quad \Gamma \vdash f_n(V_1, \ldots, V_n) : \text{NUM} \quad F \]
PCF⊕: Type Assignment Rules

**Value Typing Rules**

$$\Gamma \vdash \star : \text{UNIT} \quad S \quad \Gamma, x : \tau \vdash x : \tau \quad V \quad \Gamma \vdash r : \text{NUM} \quad R$$

$$\frac{\Gamma, x : \tau \vdash M : \rho}{\Gamma \vdash \lambda x. M : \tau \to \rho} \quad \lambda \quad \frac{\Gamma, x : \tau \to \rho \vdash M : \tau \to \rho}{\Gamma \vdash \text{fix} \, x. M : \tau \to \rho} \quad X$$

**Term Typing Rules**

$$\frac{\Gamma \vdash V : \tau \to \rho \quad \Gamma \vdash W : \tau}{\Gamma \vdash V \, W : \rho} \quad \Theta \quad \frac{\Gamma \vdash M : \tau \quad \Gamma, x : \tau \vdash N : \rho}{\Gamma \vdash \text{let} \, M = x \, \text{in} \, N : \rho} \quad L \quad \frac{\Gamma \vdash M : \tau \quad \Gamma \vdash N : \tau}{\Gamma \vdash M \oplus N : \tau} \quad \oplus$$

$$\frac{\Gamma \vdash V : \text{NUM} \quad \Gamma \vdash M : \tau \quad \Gamma \vdash N : \tau}{\Gamma \vdash \text{if} \, V \, \text{then} \, M \, \text{else} \, N : \tau} \quad I \quad \frac{\Gamma \vdash V_1 : \text{NUM} \quad \cdots \quad \Gamma \vdash V_n : \text{NUM}}{\Gamma \vdash f_n(V_1, \ldots, V_n) : \text{NUM}} \quad F$$

- The closed terms of type \(\tau\) forms a set \(\mathbb{C}T_{\tau}\).
- Similarly for values and \(\mathbb{C}V_{\tau}\).
Given any set $X$, a distribution on $X$ is a function $D : X \to \mathbb{R}_{[0,1]}$ such that $D(x) > 0$ only for denumerably many elements of $X$ and that $\sum_{x \in X} D(x) \leq 1$. 

The support of a distribution $D$ on $X$ is the subset $\text{SUPP}(D)$ of $X$ defined as $\text{SUPP}(D) := \{x \in X \mid D(x) > 0\}$.

The set of all distributions over $X$ is indicated as $D(X)$.

We indicate the distribution assigning probability 1 to the element $x \in X$ and 0 to any other element of $X$ as $\delta(x)$.

Given a distribution $D$ on $X$, its sum $\sum D$ is simply $\sum_{x \in X} D(x)$. 

Distributions
Distributions

- Given any set $X$, a *distribution* on $X$ is a function $\mathcal{D} : X \to \mathbb{R}_{[0,1]}$ such that $\mathcal{D}(x) > 0$ only for denumerably many elements of $X$ and that $\sum_{x \in X} \mathcal{D}(x) \leq 1$.
- The *support* of a distribution $\mathcal{D}$ on $X$ is the subset $\text{SUPP}(\mathcal{D})$ of $X$ defined as

\[
\text{SUPP}(\mathcal{D}) := \{ x \in X \mid \mathcal{D}(x) > 0 \}
\]
Given any set $X$, a distribution on $X$ is a function $\mathcal{D} : X \to \mathbb{R}_{[0,1]}$ such that $\mathcal{D}(x) > 0$ only for denumerably many elements of $X$ and that $\sum_{x \in X} \mathcal{D}(x) \leq 1$.

The support of a distribution $\mathcal{D}$ on $X$ is the subset $\text{SUPP}(\mathcal{D})$ of $X$ defined as

$$\text{SUPP}(\mathcal{D}) := \{x \in X \mid \mathcal{D}(x) > 0\}$$

The set of all distributions over $X$ is indicated as $\mathcal{D}(X)$. 

Given any set $X$, a distribution on $X$ is a function $\mathcal{D} : X \rightarrow \mathbb{R}_{[0,1]}$ such that $\mathcal{D}(x) > 0$ only for denumerably many elements of $X$ and that $\sum_{x \in X} \mathcal{D}(x) \leq 1$.

The support of a distribution $\mathcal{D}$ on $X$ is the subset $\text{SUPP}(\mathcal{D})$ of $X$ defined as

$$\text{SUPP}(\mathcal{D}) := \{ x \in X \mid \mathcal{D}(x) > 0 \}$$

The set of all distributions over $X$ is indicated as $\mathbf{D}(X)$.

We indicate the distribution assigning probability 1 to the element $x \in X$ and 0 to any other element of $X$ as $\delta(x)$. 
Given any set $X$, a *distribution* on $X$ is a function $\mathcal{D} : X \to \mathbb{R}_{[0,1]}$ such that $\mathcal{D}(x) > 0$ only for denumerably many elements of $X$ and that $\sum_{x \in X} \mathcal{D}(x) \leq 1$.

The *support* of a distribution $\mathcal{D}$ on $X$ is the subset $\text{SUPP}(\mathcal{D})$ of $X$ defined as

$$\text{SUPP}(\mathcal{D}) := \{x \in X \mid \mathcal{D}(x) > 0\}$$

The set of all distributions over $X$ is indicated as $\mathcal{D}(X)$.

We indicate the distribution assigning probability 1 to the element $x \in X$ and 0 to any other element of $X$ as $\delta(x)$.

Given a distribution $\mathcal{D}$ on $X$, its *sum* $\sum \mathcal{D}$ is simply $\sum_{x \in X} \mathcal{D}(x)$. 
One-Step Reduction

\[(\lambda x.M)V \rightarrow \delta(M[V/x])\]

\[\text{let } V = x \text{ in } M \rightarrow \delta(M[V/x])\]

\[\text{if } 0 \text{ then } M \text{ else } N \rightarrow \delta(M)\]

\[\text{if } r \text{ then } M \text{ else } N \rightarrow \delta(N) \text{ if } r \neq 0\]

\[M \oplus N \rightarrow \left\{ M : \frac{1}{2}, N : \frac{1}{2} \right\}\]

\[f(r_1, \ldots, r_n) \rightarrow \delta(f^*(r_1 \ldots, r_n))\]

\[M \rightarrow \{L_i : p_i\}_{i \in I}\]

\[\text{let } M = x \text{ in } N \rightarrow \{\text{let } L_i = x \text{ in } N : p_i\}_{i \in I}\]
\[
\begin{align*}
M & \Rightarrow_0 \emptyset & V & \Rightarrow_1 \delta(V) & V & \Rightarrow_{n+1} \emptyset & M & \rightarrow D & \forall N \in \text{SUPP}(D). N & \Rightarrow_n E_N \\
& & & & & & M & \Rightarrow_{n+1} \sum_{N \in \text{SUPP}(D)} D(N) \cdot E_N
\end{align*}
\]
Lemma

If $M \Rightarrow_n D$, then $\text{SUPP}(D)$ is a finite set.
Lemma

If $M \Rightarrow_n D$, then $\text{SUPP}(D)$ is a finite set.

Proposition (Progress)

For every $M \in \text{CT}_\tau$, either $M$ is a value or there is $D$ with $M \rightarrow D$.
Lemma

If $M \Rightarrow_n D$, then $\text{SUPP}(D)$ is a finite set.

Proposition (Progress)

For every $M \in \text{CT}_\tau$, either $M$ is a value or there is $D$ with $M \rightarrow D$.

Proposition (Subject Reduction)

For every $M \in \text{CT}_\tau$ and for every $n \in \mathbb{N}$, if $M \rightarrow D$ and $M \Rightarrow_n E$, then $D \in \mathcal{D}(\text{CT}_\tau)$ and $E \in \mathcal{D}(\text{CV}_\tau)$.
**PCF⊕: Some Easy Meta-Theorems**

**Lemma**

*If* $M \Rightarrow_n D$, *then* $\text{SUPP}(D)$ *is a finite set.*

**Proposition (Progress)**

*For every* $M \in \text{CT}_\tau$, *either* $M$ *is a value or there is* $D$ *with* $M \rightarrow D$.

**Proposition (Subject Reduction)**

*For every* $M \in \text{CT}_\tau$ *and for every* $n \in \mathbb{N}$, *if* $M \rightarrow D$ *and* $M \Rightarrow_n E$, *then* $D \in D(\text{CT}_\tau)$ *and* $E \in D(\text{CV}_\tau)$.

**Corollary**

*For every* $M \in \text{CT}_\tau$ *and for every* $n \in \mathbb{N}$, *there is exactly one distribution* $D_n$ *such that* $M \Rightarrow_n D_n$. *We will write* $\langle M \rangle_n$ *for such a distribution.*
PCF⊕: The Operational Semantics of a Term

- Given two distributions $\mathcal{D}, \mathcal{E} \in \mathcal{D}(X)$, we write $\mathcal{D} \leq \mathcal{E}$ iff $\mathcal{D}(x) \leq \mathcal{E}(x)$ for every $x \in X$. This relation endows $\mathcal{D}(X)$ with the structure of a partial order, which is actually an $\omega$CPO:

- Given a closed term $M \in \mathcal{CT}_\tau$, the operational semantics of $M$ is defined to be the distribution $\langle M \rangle \in \mathcal{CV}_\tau$ defined as $\sum_{n \in \mathbb{N}} \langle M \rangle_n$. 

Given two distributions $\mathcal{D}, \mathcal{E} \in \mathbf{D}(X)$, we write $\mathcal{D} \leq \mathcal{E}$ iff $\mathcal{D}(x) \leq \mathcal{E}(x)$ for every $x \in X$. This relation endows $\mathbf{D}(X)$ with the structure of a partial order, which is actually an $\omega\text{CPO}$:

Given a closed term $M \in \mathbb{C}\mathbf{T}_\tau$, the \textit{operational semantics} of $M$ is defined to be the distribution $\langle M \rangle \in \mathbb{C}\mathbf{V}_\tau$ defined as $\sum_{n \in \mathbb{N}} \langle M \rangle_n$.

**Term Contexts**

\[
C_T, D_T ::= C_V \mid [\cdot] \mid C_V V \mid V C_V \\
\quad \mid \text{let } C_T = x \text{ in } N \mid \text{let } M = x \text{ in } C_T \mid C_T \oplus D_T \\
\quad \mid \text{if } V \text{ then } C_T \text{ else } D_T
\]

**Value Contexts**

\[
C_V, D_V ::= \lambda x. C_T \mid \text{fix } x. C_V
\]
PCF⊕: The Operational Semantics of a Term

- Given two distributions \( D, E \in \mathcal{D}(X) \), we write \( D \leq E \) iff \( D(x) \leq E(x) \) for every \( x \in X \). This relation endows \( \mathcal{D}(X) \) with the structure of a partial order, which is actually an \( \omega\text{CPO} \):

- Given a closed term \( M \in \mathbb{C}T_{\tau} \), the operational semantics of \( M \) is defined to be the distribution \( \langle M \rangle \in \mathbb{C}V_{\tau} \) defined as \( \sum_{n \in \mathbb{N}} \langle M \rangle_n \).

\[
\begin{align*}
\text{Term Contexts} & : \quad C_T, D_T ::= C_V \mid [:] \mid C_V V \mid V C_V \\
& \quad \mid \text{let } C_T = x \text{ in } N \mid \text{let } M = x \text{ in } C_T \mid C_T \oplus D_T \\
& \quad \mid \text{if } V \text{ then } C_T \text{ else } D_T
\end{align*}
\]

\[
\begin{align*}
\text{Value Contexts} & : \quad C_V, D_V ::= \lambda x.C_T \mid \text{fix } x.C_V
\end{align*}
\]

- Given two terms \( M, N \) such that \( \Gamma \vdash M : \tau \) and \( \Gamma \vdash N : \tau \), we say that \( M \) and \( N \) are \( (\Gamma, \tau)\)-equivalent, and we write \( M \equiv^\tau N \) iff whenever \( \emptyset \vdash C[\Gamma \vdash \cdot : \tau] : \text{UNIT} \), it holds that \( \sum \langle C[M] \rangle = \sum \langle C[N] \rangle \).
PCF⊕: from Equivalences to Metrics?

▶ The definition of contextual equivalence asks that $\sum\langle C[M] \rangle = \sum\langle C[N] \rangle$ for every context $C$.
▶ But what if $\sum\langle C[M] \rangle$ and $\sum\langle C[N] \rangle$ are very close, without being really equal to each other?
▶ It makes sense to generalize contextual equivalence to a notion of distance:

$$\delta^{Γ,τ}(M, N) = \sup_{\emptyset ⊢ C[Γ⊢·:\tau]:UNIT} \left| \sum\langle C[M] \rangle - \sum\langle C[N] \rangle \right|.$$ 

▶ For every $Γ, τ$, $δ^{Γ,τ}$ is indeed a pseudo-metric:

$$\begin{align*}
δ^{Γ,τ}(M, M) &= 0 \\
δ^{Γ,τ}(M, N) &= δ^{Γ,τ}(N, M) \\
δ^{Γ,τ}(M, L) &\leq δ^{Γ,τ}(M, N) + δ^{Γ,τ}(N, L)
\end{align*}$$
Termination in a Probabilistic Setting

- Let $M$ be any closed term. We say that $M$ is **almost surely terminating** if $\sum \langle M \rangle = 1$, namely if its probability of convergence is 1.
Termination in a Probabilistic Setting

- Let $M$ be any closed term. We say that $M$ is **almost surely terminating** if $\sum \langle M \rangle = 1$, namely if its probability of convergence is 1.

- Example:

  $$GEO := (\text{fix } f.\lambda x.x \oplus (\text{let succ}_1(x) = y \text{ in } f y))0$$
Termination in a Probabilistic Setting

- Let $M$ be any closed term. We say that $M$ is **almost surely terminating** if $\sum \langle M \rangle = 1$, namely if its probability of convergence is 1.

- **Example:**

  $$GEO := (\text{fix } f.\lambda x.x \oplus (\text{let } \text{succ}_1(x) = y \text{ in } f \ y))0$$

- The **expected evaluation length** of any closed term as follows:

  $$ExLen(M) := \sum_{m=0}^{\infty} \left( 1 - \sum_{n=0}^{m} \sum \langle M \rangle_n \right)$$

  Let $M$ be any closed term. We say that $M$ is **positively almost surely terminating** if $ExLen(M) < +\infty$. 
Termination in a Probabilistic Setting

- Let $M$ be any closed term. We say that $M$ is **almost surely terminating** if $\sum\langle M \rangle = 1$, namely if its probability of convergence is 1.

- **Example:**

  $$GEO := (\text{fix } f. \lambda x. x \oplus (\text{let } \text{succ}_1(x) = y \text{ in } f y))0$$

- The **expected evaluation length** of any closed term as follows:

  $$ExLen(M) := \sum_{m=0}^{\infty} \left( 1 - \sum_{n=0}^{m} \sum_{\langle M \rangle_n} \right)$$

  Let $M$ be any closed term. We say that $M$ is **positively almost surely terminating** if $ExLen(M) < +\infty$.

**Lemma**

*Every positively almost-surely terminating term is almost-surely terminating.*
Variations on \( \text{PCF} \)

- **Pree, Untyped** rather than applied, typed.
  - **Terms:** \( M ::= x \mid \lambda M. \mid MM \mid M \oplus M; \)
  - **Values:** \( V ::= \lambda M. ; \)
  - **One-Step Reduction:**

\[
(\lambda x.M)V \rightarrow \delta(M[V/x]) \\
M \oplus N \rightarrow \left\{ M : \frac{1}{2}, N : \frac{1}{2} \right\}
\]

\[
M \rightarrow \{ L_i : p_i \}_{i \in I} \\
MN \rightarrow \{ L_iN : p_i \}_{i \in I} \\
VM \rightarrow \{ VL_i : p_i \}_{i \in I}
\]

- The obtained calculus will be referred to as \( \Lambda \).
Variations on $\text{PCF}_\oplus$

- **Pree, Untyped** rather than applied, typed.
  - Terms: $M := x \mid \lambda M. \mid MM \mid M \oplus M$;
  - Values: $V := \lambda M.$;
  - One-Step Reduction:
    
    $$(\lambda x.M)V \rightarrow \delta(M[V/x])$$
    $$M \oplus N \rightarrow \left\{\frac{1}{2} : M, \frac{1}{2} : N\right\}$$
    
    $$M \rightarrow \{L_i : p_i\}_{i \in I}$$
    $$MN \rightarrow \{L_iN : p_i\}_{i \in I}$$
    $$VM \rightarrow \{VL_i : p_i\}_{i \in I}$$

- The obtained calculus will be referred to as $\Lambda_\oplus$.

- **CBN** rather than CBV.
  - One-Step Reduction:
    
    $$(\lambda x.M)N \rightarrow \delta(M[N/x])$$
    $$M \oplus N \rightarrow \left\{\frac{1}{2} : M, \frac{1}{2} : N\right\}$$
    
    $$M \rightarrow \{L_i : p_i\}_{i \in I}$$
    $$MN \rightarrow \{L_iN : p_i\}_{i \in I}$$
Binary probabilistic choice is perfectly adequate to model randomised computation.
Binary probabilistic choice is perfectly adequate to model randomised computation.

**Theorem**

*The class of computable probabilistic functions coincides with the class of probabilistic functions computable by PCF\(^N\).*
Binary probabilistic choice is perfectly adequate to model randomised computation.

**Theorem**

*The class of computable probabilistic functions coincides with the class of probabilistic functions computable by $\text{PCF}^\oplus_N$.***

In recent years, starting from the pioneering works on languages like *CHURCH*, *ANGLICAN*, or *HANSEI*, functional programs have been also employed as means to represent probabilistic *models* rather than *algorithms*.

The languages above can be modeled [Staton2017] as $\lambda$-calculi endowed with two new operators:

- **sample**, modeling sampling from the uniform distribution on $[0, 1]$.
- **score**, which takes a positive real number $r$ as a parameter, and modify the *weight* of the current probabilistic branch by multiplying it by $r$. 
\[ M, N ::= \text{sample} \mid \text{score}(V). \]
PCF<sub>sample,score</sub>: Terms, Typing Rules, and Reduction

Terms

\[ M, N ::= \text{sample} \mid \text{score}(V). \]

Typing Rules

\[
\frac{\Gamma \vdash \text{sample} : \text{NUM}}{A} \quad \frac{\Gamma \vdash V : \text{NUM}}{C}
\]

One needs to switch from distributions to measures, and assume the underlying set, namely \( \mathbb{R} \) to have the structure of a measurable space.

Adapting the rule for let-terms naturally leads to

\[
\text{let } M = x \text{ in } N \rightarrow \text{let } \mu = x \text{ in } N
\]

where \( \text{let } \mu = x \text{ in } N \) should itself be a measure.

The key rule in step-indexed reduction needs to be adapted:

\[
M \rightarrow \mu \forall N \in \text{SUPP}(\mu). N \Rightarrow n \sigma N \Rightarrow n+1 A \mapsto \int \sigma N (A) \mu (dN)
\]
PCF\textsubscript{sample, score}: Terms, Typing Rules, and Reduction

**Terms** \[ M, N ::= \text{sample} \mid \text{score}(V). \]

**Typing Rules**

\[
\begin{array}{c}
\frac{\Gamma \vdash \text{sample} : \text{Num}}{A} \\
\frac{\Gamma \vdash V : \text{Num}}{\text{C}}
\end{array}
\]

- One needs to switch from distributions to *measures*, and assume the underlying set, namely $\mathbb{R}$ to have the structure of a measurable space.
- Adapting the rule for \texttt{let}-terms naturally leads to

\[
\begin{array}{c}
M \rightarrow \mu \\
\text{let } M = x \text{ in } N \rightarrow \text{let } \mu = x \text{ in } N
\end{array}
\]

where \texttt{let } $\mu = x \text{ in } N$ should itself be a measure.
- The key rule in step-indexed reduction needs to be adapted:

\[
\begin{array}{c}
M \rightarrow \mu \\
\forall N \in \text{SUPP}(\mu) \cdot N \Rightarrow_n \sigma_N
\end{array}
\]

\[
M \Rightarrow_{n+1} A \leftrightarrow \int \sigma_N(A) \mu(dN)
\]
It can well be that $\langle M \rangle = \mu$, where $\mu$ sums to something strictly higher than 1.

How should we interpret $\mu(A)$ for a measurable set of terms $A$?

We need to normalize $\mu$!

Explicitly building a normalized version of $\mu$ (if it exists) is the goal of so-called inference algorithms.

The operational semantics we have just introduced, called distribution-based is thus just an idealized form of semantics.

An executable semantics can be given in the form of sampling-based semantics.
The Operational Meaning of Bayesian Terms

- It can well be that $\langle M \rangle = \mu$, where $\mu$ sums to something strictly higher than 1.
- How should we interpret $\mu(A)$ for a measurable set of terms $A$?
- We need to normalize $\mu$!
The Operational Meaning of Bayesian Terms

- It can well be that $\langle M \rangle = \mu$, where $\mu$ sums to something strictly higher than 1.
- How should we interpret $\mu(A)$ for a measurable set of terms $A$?
- We need to normalize $\mu$!
- Explicitly building a normalized version of $\mu$ (if it exists) is the goal of so-called inference algorithms.
The Operational Meaning of Bayesian Terms

- It can well be that $\langle M \rangle = \mu$, where $\mu$ sums to something strictly higher than 1.
- How should we interpret $\mu(A)$ for a measurable set of terms $A$?
- We need to normalize $\mu$!
- Explicitly building a normalized version of $\mu$ (if it exists) is the goal of so-called inference algorithms.
- The operational semantics we have just introduced, called distribution-based is thus just an idealized form of semantics.
- An executable semantics can be given in the form of sampling-based semantics.
Sampling-Based Semantics (1)

\[
\langle (\lambda x.M)V, s \rangle \xrightarrow{1} \langle M[V/x], s \rangle
\]

\[
\langle \text{let } V = x \text{ in } M, s \rangle \xrightarrow{1} \langle M[V/x], s \rangle
\]

\[
\langle \text{if } 0 \text{ then } M \text{ else } N, s \rangle \xrightarrow{1} \langle M, s \rangle
\]

\[
\langle \text{if } r \text{ then } M \text{ else } N, s \rangle \xrightarrow{1} \langle N, s \rangle \text{ if } r \neq 0
\]

\[
\langle \text{sample, } r :: s \rangle \xrightarrow{1} \langle r, s \rangle
\]

\[
\langle \text{score}(r), s \rangle \xrightarrow{r} \langle *, s \rangle
\]

\[
\langle f(r_1, \ldots, r_n), s \rangle \xrightarrow{1} \langle f^*(r_1 \ldots, r_n), s \rangle
\]

\[
\langle M, s \rangle \xrightarrow{r} \langle L, t \rangle
\]

\[
\langle \text{let } M = x \text{ in } N, s \rangle \xrightarrow{r} \langle \text{let } L = x \text{ in } N, s \rangle
\]
\[ (V, s) \xrightarrow{1} (V, s) \]
\[ (M, s) \xrightarrow{r} (N, t) \]
\[ (N, t) \xrightarrow{s} (L, u) \]
\[ (M, s) \xrightarrow{r \cdot s} (L, u) \]
Thank You!

Questions?