From Hard Work to Trickery: A Systematic Approach to Probabilistic Rewriting.

Flavien BREUVART, Ugo Dal Lago

Focus, Inria team, Bologna

Crecogi: August 28 2016
Probabilistic Rewriting

Motivations
Randomized Algorithmics
Efficiency analysis
Cryptography
Security
Machine Learning
Modeling

Probabilistic $\lambda$-calculus (Weak head-reduction)

\[
M, N := x | \lambda x.M | M N | M \oplus N
\]

\[
(\lambda x.M) N \xrightarrow{1} M[N/x]
\]

\[
M \oplus N \xrightarrow{\frac{1}{2}} M \xrightarrow{\frac{1}{2}} N
\]

\[
A := \lambda u.x(\lambda y.(u(Sx) \oplus y)I)
\]

\[
\text{Prob}(\text{YA0} \downarrow) < 1
\]
The Quest for a Semantics

**Denotational Semantics:**
- a 20 years old challenge
- Finding subclass of domains that:
  - is a Cartesian close
  - have probabilistic powerdomains
- Real issue:
  - higher order probabilities

**Operational Semantics:**
- a hidden challenge
- Rewriting theory with:
  - probabilistic behaviors
  - systematic proof-schemes
- Real issue:
  - proba forces topological arguments

**Unconventional solution:**
- Probabilistic coherent spaces
  - [EhrhardPaganiTasson2014]

**Unconventional solution:**
- ???

Our objective
What is a Systematic Proof Schema?  
The Example of Infinite Rewriting

**Question:**  
How to relate small-step and multi-step?

At the beguiling:  Topology

Limits for Cantor topology of sequential small-step reductions.

Now-day:  Coinduction

\[
M \rightarrow^* f(L_1,\ldots,L_k) \quad \forall i \leq k, \quad L_i \rightarrow^\omega N_i; \\
M \rightarrow^\omega f(N_1,\ldots,N_k)
\]

Coinduction Schema

For any relation \( \rightsquigarrow \) over terms, if for all \( M \rightsquigarrow f(N_1,\ldots,N_k) \), there is \( L_1,\ldots,L_k \) such that \( M \rightarrow^* f(L_1,\ldots,L_k) \) and \( L_i \rightsquigarrow N_i \), then \( \rightsquigarrow \subseteq \rightarrow^\omega \).
What About Probabilistic Rewriting

Probabilities and Non-determinism does not mix well
For now, let’s forget about non determinism. This means:

- Fixing a strategy
- Big step rather than multistep

Probabilities are inherently topological
[0,1] is, before all, a topological space...

Most rewriting theory’s tools are continuous
Bisimulations, encoding, typing, modeling...

Can we treat those tools without referring to topology?
Probabilities and Non-determinism does not mix well

For now, let’s forget about non determinism. This means:
- Fixing a strategy
- Big step rather than multistep

Probabilities are inherently topological
[0,1] is, before all, a topological space...

Most rewriting theory’s tools are continuous
Bisimulations, encoding, typing, modeling...

Can we treat those tools without referring to topology?

Yes! but there is a price to pay: a dynamic target
### Probabilistic Rewriting System

#### Randomized function

\[
f : U \rightarrow V \text{ denotes a function } f : U \rightarrow \mathcal{D}(V)
\]

\[
\mathcal{D}(V) = \{d \in V \rightarrow [0, 1] \mid \sum_{v \in V} d(v) \leq 1\}
\]

#### Definition of (Abstract) Probabilistic Rewriting System

<table>
<thead>
<tr>
<th>Terms</th>
<th>Normal forms</th>
<th>Small step</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\Lambda)</td>
<td>(\Lambda = \Lambda_V \cup \Lambda_R)</td>
<td>reduction: (\Lambda_R \rightarrow \Lambda)</td>
</tr>
</tbody>
</table>

**Remark:** We only consider deterministic systems (with a strategy)

#### Coalgebraic approach of the big step reduction [Hasuo]

The evaluation \(\text{eval}: \Lambda \rightarrow \Lambda_V\) corresponds to the final arrow in the \((\_ \times \mathbb{N})\)-coalgebra category over set and randomized functions.
Theorem: Randomized Encoding

\[ \Lambda, \Pi \quad \text{probabilistic rewriting systems.} \]

encoding : \(\Lambda \rightarrow \Pi\) a randomized function preserving NF and reducibles.

\[ \Lambda_R \xrightarrow{\text{encoding}} \Pi_R \]

\[ \Lambda \xrightarrow{\text{reduction}}\Pi \]

\[ \Lambda \xleftarrow{\text{encoding}} \]

\[ \Lambda_{v} \xrightarrow{\text{encoding}} \Pi_{v} \]

\[ \Lambda \xrightarrow{\text{eval}} \Pi \]

\[ \Lambda \xleftarrow{\text{encoding}} \]

\[ \Lambda_{v} \xrightarrow{\text{encoding}} \Pi_{v} \]
Dynamic is Essential

Only true if $||\text{interpret}(M)|| \leq ||\text{eval}(M)||$

Require a nontrivial realisability proof.
Example: Performing Choices First

In any (binary) probabilistic rewriting system, the probabilistic choices can be chosen uniformly over $\text{Bool}^\omega$ at the beginning.

\[
\Lambda_R \xrightarrow{\text{Unif}([\_] \times \text{Bool}^\omega)} \Lambda_R \times \text{Bool}^\omega \xrightarrow{\text{reduction}} \Lambda \times \text{Bool}^\omega \xrightarrow{\text{eval}} \Lambda_V \times \text{Bool}^\omega
\]
Example: Performing Choices First

In any (binary) probabilistic rewriting system, the probabilistic choices can be chosen uniformly over $\text{Bool}^\omega$ at the beginning.

\[
\text{Unif}(\{\_\} \times \text{Bool}^\omega) \quad \text{Unif}(\{\_\} \times \text{Bool}^\omega) \quad \text{Unif}(\{\_\} \times \text{Bool}^\omega)
\]

Our theorem is Valid for continuous probabilities!
Probabilistic Intersection Types

From Probabilistic Coherence Spaces
To Probabilistic Intersection Types

Standard translation:
- compacts points $\sim\sim$ intersection types
- prime algebraic points $\sim\sim$ linear types

The adequation (reformulation)

$$\text{Prob}(M \downarrow) = \sum W(M)$$

where $W(M) = \left[ p \left| \frac{\pi}{\vdash M : p \cdot \alpha} \right. \right]$}

Underlying function

$$||\text{eval}(M)|| = ||\text{deriv}(M)||$$

$$\text{deriv}: \left( \begin{array}{c}
\Lambda \\
M
\end{array} \right) \rightarrow \left\{ \frac{\varnothing(\Pi)}{\frac{\pi}{\vdash M : p \cdot \alpha}} \rightarrow p \right\}$$

$\text{probabilistic bound or weight}$
Probabilistic Intersection Types

From Probabilistic Coherence Spaces
to Probabilistic Intersection Types

Standard translation:

- compacts points ⇝ intersection types
- prime algebraic points ⇝ linear types

\[ \pi \vdash M : p \cdot \alpha \]

The adequation (reformulation)

\[ \text{Prob}(M \Downarrow) = \sum W(M) \]

where \( W(M) = [p \mid \pi \vdash M : p \cdot \ast] \)

The underlying function

\[ ||\text{eval}(M)|| = ||\text{deriv}(M)|| \quad \text{der} : \begin{cases} \Lambda \rightarrow \mathcal{D}(\Pi) \\ M \rightarrow \{ \pi \mid \vdash M : p \cdot \ast \rightarrow p \} \end{cases} \]

\( p \) IS NOT the probability
for \( M \) to be of type \( \alpha \)

Rather, \( p \) IS the probability
for \( \pi \) to be a proof of \( \vdash M : \alpha \)
Sketching the Proof of intersection types

**Cut Elimination**

\[ \pi \vdash M : p \cdot \ast \rightsquigarrow \pi' \vdash M' : q \cdot \ast \]

such that: \rightsquigarrow is

- normalizing
- deterministic

“Poliadic \( \lambda \)-calculus”

**Small-step distrib.**

\[ \Lambda_R \xrightarrow{\text{deriv}} \Pi_R \]

\[ \Lambda \xrightarrow{\text{reduction}} \text{red.} \]

\[ \vdash V : 1 \cdot \ast \]

**Value determinism**

\[ \forall \text{ normal form } V, \text{ Unicity of derivation} \]

\[ \vdash \lambda x. M : 1 \cdot \ast \]

**Big-step distribution**

\[ \Lambda \xrightarrow{\text{deriv}} \Pi \]

\[ \Lambda_V \xrightarrow{\text{deriv}} \Pi_V \]

**Conclusion**

\[ ||\text{eval}(M)|| = ||\text{deriv(eval}(M))|| \]

\[ = ||\text{eval(deriv}(M))|| \]

\[ = ||\text{deriv}(M)|| \]
And Then?
Introduce non-determinism

Convex set of distributions

A randomized simulation is a function

\[ f : U \rightarrow \mathcal{C}(\mathcal{D}(V)) \]

targeting convex sets of distributions.

- Our Theorem holds for randomized simulation,
- A randomized encoding is a functional randomized simulation,
- A probabilistic bisimulation a derandomized randomized simulation.

Maybe a direction to treat real rewriting issues

Probabilistic confluence, Powerful bisimulations...