Distribution Types: A Type-Theoretic Approach To Almost Sure Termination

Ugo dal Lago Charles Grellois

FOCUS Team – INRIA & University of Bologna

Institute for Mathematical Sciences, Singapore
September 9, 2016
Motivations

- **Probabilistic** programming languages are important in computer science: modeling uncertainty, machine learning, AI...

- **Quantitative notion of termination**: almost-sure termination (AST)

- AST has been studied for imperative programs in the last years...

- ...but what about the **functional** probabilistic languages?

We introduce a **monadic, affine sized type system** sound for AST.
Sized types: the deterministic case

Simply-typed $\lambda$-calculus is strongly normalizing (SN).

No longer true with the letrec construction…

**Sized types**: a decidable extension of the simple type system ensuring SN for $\lambda$-terms with letrec.

See notably:

- Hughes-Pareto-Sabry 1996, *Proving the correctness of reactive systems using sized types*,
- Barthe-Frade-Giménez-Pinto-Uustalu 2004, *Type-based termination of recursive definitions*. 
Sized types: the deterministic case

Sizes: \( s, r ::= i | \infty | \hat{s} \)

+ size comparison inducing subtyping. Notably \( \hat{\infty} = \infty \).

Idea:

- \( \text{Nat}^i \) is 0,
- \( \text{Nat}^\hat{i} \) is 0 or \( S \ 0 \),
- \( \ldots \)
- \( \text{Nat}^\infty \) is any natural number. Often denoted simply \( \text{Nat} \).

The same for lists,\ldots
Sized types: the deterministic case

Sizes: \( \mathcal{s}, \mathcal{r} ::= \mathcal{i} \mid \infty \mid \hat{s} \)

+ size comparison inducing subtyping. Notably \( \hat{\infty} = \infty \).

Fixpoint rule:

\[
\frac{\Gamma, f : \text{Nat}^i \rightarrow \sigma \vdash M : \text{Nat}^{\hat{i}} \rightarrow \sigma[i/\hat{i}] \quad \text{i pos } \sigma}{\Gamma \vdash \text{letrec } f = M : \text{Nat}^{\hat{s}} \rightarrow \sigma[i/\hat{s}]}
\]

Typable \( \implies \) SN. Proof using reducibility candidates.

Decidable type inference.
Sized types: example in the deterministic case

From Barthe et al. (op. cit.):

\[
\text{plus} \equiv \text{letrec} \quad \text{plus} : \text{Nat} \to \text{Nat} \to \text{Nat} = \\
\quad \lambda x : \text{Nat}. \lambda y : \text{Nat}. \quad \text{case } x \text{ of } \begin{cases} 
\text{o} & \Rightarrow y \\
\text{s} & \Rightarrow \lambda x' : \text{Nat}. \quad \text{s} \underbrace{(\text{plus} \ x' \ y)}_{\text{:Nat}} 
\end{cases} \\
) : \quad \text{Nat}^\ast \to \text{Nat} \to \text{Nat}
\]

Size decreases during recursive calls ⇒ SN.
A probabilistic $\lambda$-calculus

$M, N, \ldots ::= V \mid V V \mid \text{let } x = M \text{ in } N \mid M \oplus_p N$

\[ \text{letrec } f = V \]

\[ \text{case } V \text{ of } \{ S \rightarrow W \mid 0 \rightarrow Z \} \]

$V, W, Z, \ldots ::= x \mid 0 \mid S V \mid \lambda x. M \mid \text{letrec } f = V$

- Formulation equivalent to $\lambda$-calculus with $\oplus_p$, but constrained for technical reasons
- Restriction to base type Nat for simplicity, but can be extended to general inductive datatypes (as in sized types)
A probabilistic $\lambda$-calculus: operational semantics

\[
\text{let } x = V \text{ in } M \rightarrow_v \left\{ (M[x/V])^1 \right\}
\]

\[
(\lambda x. M) V \rightarrow_v \left\{ (M[x/V])^1 \right\}
\]

\[
(\text{letrec } f = V) \left( c \overrightarrow{W} \right) \rightarrow_v \left\{ \left( V[f/(\text{letrec } f = V)] \left( c \overrightarrow{W} \right) \right)^1 \right\}
\]
A probabilistic $\lambda$-calculus: operational semantics

\[
\text{case } S \ V \text{ of } \{ S \to W \ | \ 0 \to Z \} \rightarrow_{\nu} \left\{ (W \ V)^1 \right\}
\]

\[
\text{case } 0 \text{ of } \{ S \to W \ | \ 0 \to Z \} \rightarrow_{\nu} \left\{ (Z)^1 \right\}
\]
A probabilistic $\lambda$-calculus: operational semantics

$$
M \oplus_p N \rightarrow_v \{ M^p, N^{1-p} \}
$$

$$
\frac{M \rightarrow_v \{ L_i^{p_i} \mid i \in I \}}{
\text{let } x = M \text{ in } N \rightarrow_v \{ (\text{let } x = L_i \text{ in } N)^{p_i} \mid i \in I \}}
$$
A probabilistic $\lambda$-calculus: operational semantics

\[
\mathcal{D} \quad \overset{VD}{=} \quad \left\{ M_j^p_j \mid j \in J \right\} + \mathcal{D}_V \quad \forall j \in J, \quad M_j \rightarrow_v E_j
\]

\[
\mathcal{D} \quad \rightarrow_v \quad \left( \sum_{j \in J} p_j \cdot E_j \right) + \mathcal{D}_V
\]

For $\mathcal{D}$ a distribution of terms:

\[
\llbracket \mathcal{D} \rrbracket = \sup_{n \in \mathbb{N}} \left( \left\{ \mathcal{D}_n \mid \mathcal{D} \Rightarrow^n_v \mathcal{D}_n \right\} \right)
\]

where $\Rightarrow^n_v$ is $\rightarrow^n_v$ followed by projection on values.

We let $\llbracket M \rrbracket = \llbracket \left\{ M^1 \right\} \rrbracket$.

$M$ is AST iff $\sum \llbracket M \rrbracket = 1$. 
Random walks as probabilistic terms

- **Biased** random walk:

\[
M_{bias} = \left( \text{letrec } f = \lambda x. \text{case } x \text{ of } \{ S \rightarrow \lambda y. f(y) \oplus \frac{2}{3} (f(SSf(y))) \mid 0 \rightarrow 0 \} \right)^n
\]

- **Unbiased** random walk:

\[
M_{unb} = \left( \text{letrec } f = \lambda x. \text{case } x \text{ of } \{ S \rightarrow \lambda y. f(y) \oplus \frac{1}{2} (f(SSf(y))) \mid 0 \rightarrow 0 \} \right)^n
\]

\[
\sum [M_{bias}] = \sum [M_{unb}] = 1
\]

Capture this in a sized type system?
Another term

We also want to capture terms as:

\[ M_{nat} = \left( \text{letrec } f = \lambda x. x \oplus \frac{1}{2} S (f \ x) \right) \ 0 \]

of semantics

\[ \llbracket M_{nat} \rrbracket = \left\{ (0)^{\frac{1}{2}}, (S \ 0)^{\frac{1}{4}}, (S \ S \ 0)^{\frac{1}{8}}, \ldots \right\} \]

summing to 1.
First idea: extend the sized type system with:

\[
\frac{\Gamma \vdash M : \sigma \quad \Gamma \vdash N : \sigma}{\Gamma \vdash M \oplus_p N : \sigma}
\]

and “unify” types of \( M \) and \( N \) by subtyping.

Kind of product interpretation of \( \oplus \): we can’t capture more than SN…
Beyond SN terms, towards distribution types

First idea: extend the sized type system with:

\[ \Gamma \vdash M : \sigma \quad \Gamma \vdash N : \sigma \]

Choice

\[ \Gamma \vdash M \oplus_p N : \sigma \]

and “unify” types of \( M \) and \( N \) by subtyping.

We can’t type \( M_{bias} \) nor \( M_{unb} \) in a way decreasing the size (essential for letrec): we get at best

\[ f : \text{Nat}^i \rightarrow \text{Nat}^\infty \vdash \lambda y. f(y) \oplus_{\frac{2}{3}} (f(S S y))) : \text{Nat}^i \rightarrow \text{Nat}^\infty \]

and can’t use a variation of the letrec rule on that.
Beyond SN terms, towards distribution types

We will use distribution types, built as follows:

\[
\frac{
\Gamma \mid \Theta \vdash M : \mu \\
\Gamma \mid \Psi \vdash N : \nu \\
\{\mu\} = \{\nu\}
}{
\Gamma \mid \Theta \oplus_p \Psi \vdash M \oplus_p N : \mu \oplus_p \nu
}
\]

Now

\[
f : \left\{ \left( \text{Nat}^i \to \text{Nat}^{\infty} \right)^{\frac{2}{3}}, \left( \text{Nat}^{\hat{i}} \to \text{Nat}^{\infty} \right)^{\frac{1}{3}} \right\}
\]

\[
\vdash \\
\lambda y. f(y) \oplus^{\frac{2}{3}} (f(SSy))) : \text{Nat}^{\hat{i}} \to \text{Nat}^{\infty}
\]
Beyond SN terms, towards distribution types

We will use distribution types, built as follows:

\[
\frac{\Gamma \vdash M : \mu \quad \Gamma \vdash N : \nu}{\Gamma \vdash M \oplus p N : \mu \oplus_p \nu}
\]

Choice

Similarly:

\[
f : \left\{ \left( \text{Nat}^i \to \text{Nat}^\infty \right)^{1/2}, \left( \text{Nat}^i \to \text{Nat}^\infty \right)^{1/2} \right\}
\]

\[
\vdash \lambda y. f(y) \oplus_{1/2} (f(SSy))) : \text{Nat}^i \to \text{Nat}^\infty
\]
Designing the fixpoint rule

\[ \{ \Gamma \} = \text{Nat} \]

\[ i \notin \Gamma \text{ and } i \text{ positive in } \nu \]

\[ \Gamma \vdash f : \{ (\text{Nat}^{s_j} \rightarrow \nu[s_j/i])^{p_j} \mid j \in J \} \vdash V : \text{Nat}^\hat{i} \rightarrow \nu[^{\hat{i}}/i] \]

\[ \Gamma \vdash \text{letrec } f = V : \text{Nat}^r \rightarrow \nu[^{r}/i] \]
Designing the fixpoint rule

\[
\{ \Gamma \} = \text{Nat} \\
i \not\in \Gamma \text{ and } i \text{ positive in } \nu \\
\sum_{j \in J} p_j(\nu_j) < 1
\]

\[
\text{LetRec} \quad \frac{\Gamma \vdash f : \left\{ (\text{Nat}^{\nu_j} \rightarrow \nu[\nu_j/i])^{p_j} \mid j \in J \right\} \vdash V : \text{Nat}^\nu \rightarrow \nu[\nu/i]}{\Gamma \vdash \text{letrec } f = V : \text{Nat}^\rho \rightarrow \nu[\rho/i]}
\]

would allow to type

\[
M_{bias} = \left(\text{letrec } f = \lambda x.\text{case } x \text{ of } \left\{ S \rightarrow \lambda y.f(y) \oplus_\frac{2}{3} (f(SS\ y)) \mid 0 \rightarrow 0 \right\}\right) \ n
\]
Designing the fixpoint rule

\{ \Gamma \} = \text{Nat}

i \not\in \Gamma \text{ and } i \text{ positive in } \nu

\sum_{j \in J} p_j(\|s_j\|) < 1 \text{ or } \sum_{j \in J} p_j < 1

\text{LetRec}

\Gamma \mid f : \{ (\text{Nat}^{s_j} \rightarrow \nu[\hat{s}_j/i])^{p_j} \mid j \in J \} \vdash V : \text{Nat}^{\hat{i}} \rightarrow \nu[\hat{i}/i]

\Gamma \mid \emptyset \vdash \text{letrec } f = V : \text{Nat}^{r} \rightarrow \nu[r/i]

would allow to type \( M_{\text{nat}} \) too:

\[ M_{\text{nat}} = \left( \text{letrec } f = \lambda x. x \oplus \frac{1}{2} S (f \ x) \right) 0 \]
Designing the fixpoint rule

\[ \{ \Gamma \} = \text{Nat} \]

\[ i \notin \Gamma \text{ and } i \text{ positive in } \nu \]

\[ \sum_{j \in J} p_j(|s_j|) < 1 \text{ or } \sum_{j \in J} p_j < 1 \]

\[ \text{LetRec} \]

\[ \frac{\Gamma | f : \left\{ \left( \text{Nat}^{s_j} \rightarrow \nu[s_j/i] \right)p_j \mid j \in J \right\} \vdash V : \text{Nat}^{\hat{i}} \rightarrow \nu[\hat{i}/i]}{\Gamma | \emptyset \vdash \text{letrec } f = V : \text{Nat}^r \rightarrow \nu[r/i]} \]

But how to cope with

\[ M_{\text{unb}} = \left( \text{letrec } f = \lambda x.\text{case } x \text{ of } \left\{ S \rightarrow \lambda y.f(y) + \frac{1}{2} (f(S S y)) \mid 0 \rightarrow 0 \right\} \right)^n \]
Designing the fixpoint rule

\{ \Gamma \} = \text{Nat} \\
i \not\in \Gamma \text{ and } i \text{ positive in } \nu \\
\{ (\text{Nat}^j \rightarrow \nu[s_j/i])^{p_j} \mid j \in J \} \text{ induces an AST sized walk} \\

\text{LetRec} \\
\Gamma \vdash f : \{ (\text{Nat}^j \rightarrow \nu[s_j/i])^{p_j} \mid j \in J \} \vdash V : \text{Nat}^{\hat{i}} \rightarrow \nu[\hat{i}/i] \\
\Gamma \mid \emptyset \vdash \text{letrec } f = V : \text{Nat}^r \rightarrow \nu[r/i]

solves the problem for

\( M_{unb} = \left( \text{letrec } f = \lambda x. \text{case } x \text{ of } \{ \text{S } \rightarrow \lambda y. f(y) \oplus \frac{1}{2} (f(S\ S\ y)) \mid 0 \rightarrow 0 \} \right) n \)

by deferring to an external \text{PTIME} procedure the convergence checking.
Generalized random walks and the necessity of affinity

A crucial feature: our type system is affine. Higher-order symbols occur at most once. Why? Consider:

\[ M_{naff} = \text{letrec } f = \lambda x.\text{case } x \text{ of } \left\{ \begin{array}{l} S \to \lambda y.f(y) \oplus_{2} (f(S S y); f(S S y)) \mid 0 \to 0 \end{array} \right\} \]

and recall that its affine version was AST. Some reductions:

\[
\begin{align*}
M_{naff}(S 0) & \to^* 0 \\
M_{naff}(S 0) & \to^* M_{naff}(S S 0); M_{naff}(S S 0) \\
& \to^* M_{naff}(S 0); M_{naff}(S S 0) \\
& \to^* M_{naff} 0; M_{naff}(S S 0) \\
& \to^* 0; M_{naff}(S S 0) \\
& \to^* M_{naff}(S S 0) \\
M_{naff}(S 0) & \to^* M_{naff}(S S S 0); M_{naff}(S S S 0); M_{naff}(S S 0) \\
& \to^* 0
\end{align*}
\]
Generalized random walks and the necessity of affinity

Tree of recursive calls:

```
[0]          [2 2]
          /     |
[1]       [2 1]   [2 3 3]
     /   
[2]   [2 2 2]
  /   
[3 3]  
```

Leftmost edges have probability $\frac{2}{3}$; rightmost ones $\frac{1}{3}$. 
Generalized random walks and the necessity of affinity

Local shape

\[
\begin{align*}
[i_1 \cdots i_k] & \\
[i_1 \cdots i_k - 1] & [i_1 \cdots i_k + 1 i_k + 1]
\end{align*}
\]

when \( i_k > 1 \), and

\[
\begin{align*}
[i_1 \cdots i_{k-1} 1] & \\
[i_1 \cdots i_{k-1}] & [i_1 \cdots i_{k-1} 2 2]
\end{align*}
\]

else. Leaves are all labeled with \([0]\).

The rightmost branch always increases the sum \( i_1 + \cdots + i_k \) by at least 3 → non AST random walk.
Key points

- **Affine** type system
- **Distribution** types
- **Sized walks** induced by the letrec rule and solved by an external PTIME procedure
Key properties

**Theorem (Subject reduction)**

Let $n \in \mathbb{N}$, and $\left\{ (M_i : \mu_i)^{p_i} \mid i \in I \right\}$ be a closed typed distribution. Suppose that $\left\{ (M_i)^{p_i} \mid i \in I \right\} \xrightarrow{n \, \nu} \left\{ (N_j)^{p'_j} \mid j \in J \right\}$ then there exists a closed typed distribution $\left\{ (L_k : \nu_k)^{p''_k} \mid k \in K \right\}$ such that

- $E\left((M_i : \mu_i)^{p_i}\right) = E\left((L_k : \nu_k)^{p''_k}\right)$,

- and that $\left[ (L_k)^{p''_k} \mid k \in K \right]$ is a pseudo-representation of $\left\{ (N_j)^{p'_j} \mid j \in J \right\}$. 
Theorem (Subject reduction)

Let \( n \in \mathbb{N} \), and \( \{ (M_i : \mu_i)^{p_i} \mid i \in I \} \) be a closed typed distribution. Suppose that \( \{ (M_i)^{p_i} \mid i \in I \} \rightarrow_n^v \{ (N_j)^{p'_j} \mid j \in J \} \) then there exists a closed typed distribution \( \{ (L_k : \nu_k)^{p''_k} \mid k \in K \} \) such that

- \( E((M_i : \mu_i)^{p_i}) = E((L_k : \nu_k)^{p''_k}) \),
- and that \( \{ (L_k)^{p''_k} \mid k \in K \} \) is a pseudo-representation of \( \{ (N_j)^{p'_j} \mid j \in J \} \).
Key properties

Theorem (Subject reduction)

Let $n \in \mathbb{N}$, and $\{ (M_i : \mu_i)^{p_i} \mid i \in I \}$ be a closed typed distribution. Suppose that $\{ (M_i)^{p_i} \mid i \in I \} \rightarrow^n v \{ (N_j)^{p_j'} \mid j \in J \}$ then there exists a closed typed distribution $\{ (L_k : \nu_k)^{p_k''} \mid k \in K \}$ such that

- $E((M_i : \mu_i)^{p_i}) = E((L_k : \nu_k)^{p_k''})$,
- and that $\{ (L_k)^{p_k''} \mid k \in K \}$ is a pseudo-representation of $\{ (N_j)^{p_j'} \mid j \in J \}$.
Theorem (Subject reduction)

Let \( n \in \mathbb{N} \), and \( \{ (M_i : \mu_i)^{p_i} \mid i \in I \} \) be a closed typed distribution. Suppose that \( \{ (M_i)^{p_i} \mid i \in I \} \rightarrow^n \{ (N_j)^{p'_j} \mid j \in J \} \) then there exists a closed typed distribution \( \{ (L_k : \nu_k)^{p''_k} \mid k \in K \} \) such that

1. \( \mathbb{E}((M_i : \mu_i)^{p_i}) = \mathbb{E}((L_k : \nu_k)^{p''_k}) \),

2. and that \( \{ (L_k)^{p''_k} \mid k \in K \} \) is a pseudo-representation of \( \{ (N_j)^{p'_j} \mid j \in J \} \).
Theorem (Subject reduction)

Let \( n \in \mathbb{N} \), and \( \{ (M_i : \mu_i)^{p_i} \mid i \in I \} \) be a closed typed distribution. Suppose that \( \{ (M_i)^{p_i} \mid i \in I \} \rightarrow_n^\nu \{ (N_j)^{p_j'} \mid j \in J \} \) then there exists a closed typed distribution \( \{ (L_k : \nu_k)^{p_k''} \mid k \in K \} \) such that

- \( \mathbb{E} ((M_i : \mu_i)^{p_i}) = \mathbb{E} ((L_k : \nu_k)^{p_k''}) \),
- and that \( \{ (L_k)^{p_k''} \mid k \in K \} \) is a pseudo-representation of \( \{ (N_j)^{p_j'} \mid j \in J \} \).
Key properties

Theorem (Typing soundness)

\[ \text{If } \Gamma | \Theta \vdash M : \mu, \text{ then } M \text{ is AST.} \]

Proof by reducibility, using set of candidates parametrized by probabilities.

Next step: look for the type inference (decidable again??)

Thank you for your attention!
Key properties

Theorem (Typing soundness)

If $\Gamma \vdash \Theta \vdash M : \mu$, then $M$ is AST.

Proof by reducibility, using set of candidates parametrized by probabilities.

Next step: look for the type inference (decidable again??)

Thank you for your attention!
**Theorem (Typing soundness)**

If $\Gamma \vdash_\Theta M : \mu$, then $M$ is AST.

Proof by reducibility, using set of candidates parametrized by probabilities.

**Next step:** look for the type inference (decidable again??)

Thank you for your attention!