Improvement Theory for Probabilistic Call-by-Need

David Sabel and Manfred Schmidt-Schauß

WPTE 2026, Lisbon, 19 July 2026

Introduction

Motivation

Probabilistic Programming	+	Call-by-Need Functional Programming Languages

- programs represent stochastic models
- program execution is a probabilistic experiment
- apply PL-techniques to reason on the models

- declarative, higher-order, equational reasoning
- efficient implementation of lazy evaluation
- semantics differs from call-by-name and call-by-value

$\longrightarrow$ **Investigate probabilistic call-by-need functional languages**

Previous Work

Probabilistic call-by-need lambda-calculi with binary choice $s \oplus t$

- correctness of program transformations w.r.t. contextual equivalence in an untyped calculus with recursive `let` [PPDP 2022]
- a PCF-like calculus, distribution equivalence coincides with contextual equivalence on programs of type nat [WPTE 2022, JLAMP 2023]
- encoding unfair, biased choice $\overset{p}{\oplus}$ with real-valued probability p by fair choice $\overset{1/2}{\oplus}$ [WPTE 2025]

Improvement Theory

Improvement theory was invented by Dave Sands in the 1990s, e.g.

- D.Sands: Total Correctness by Local Improvement in Program Transformation, POPL 95
- A.Moran & D.Sands, Improvement in a Lazy Context: An Operational Theory for Call-by-Need, POPL 99
- J.Gustavsson & D.Sands: Possibilities and Limitations of Call-by-Need Space Improvement, ICFP 01

Improvement $p_1 \preceq p_2$

Program p_2 **improves** program p_1 iff

$$p_1 \sim p_2 \qquad \text{and} \qquad \forall C : Cost(C[p_1]) \geq Cost(C[p_2])$$

p_1 and p_2 are semantically equal

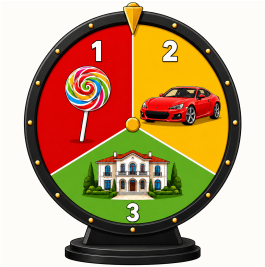
replacing p_1 with p_2 in any context does not increase the cost (runtime, space, etc.)

Our Goal: Develop an **improvement theory for probabilistic call-by-need**

Improvement Theory & Probabilism: Challenges

$$p_2 \text{ improves } p_1 \text{ iff } p_1 \sim p_2 \text{ and } \forall C : Cost(C[p_1]) \geq Cost(C[p_2])$$

- since programs represent stochastic models,
  an improvement computes the same model more efficiently.
- in deterministic PLs programs converge or diverge, and converging programs have a unique evaluation length,
  but in probabilistic PLs: expectation values have to be used (expected convergence, expected evaluation length), computed as limits
- find an alternative characterisation of the improvement-property
  to avoid the universal quantification over all contexts



Stochastic model:
uniform distribution over the outcomes $1, 2, 3$.

One stochastic model, two probabilistic programs

- Program 1: uniformly draw from $\{0,1,2,3\}$, restart on 0:

$$\begin{tikzcd}[column sep=small]
 & & {\color{red}0} \arrow[ddll, bend right, red, "\textit{restart}"'] \\
 & (0\oplus 1) \arrow[ur, red, "1/2"] \arrow[r, "1/2"'] & 1 \\
(0\oplus 1)\oplus(2\oplus 3) \arrow[ur, "1/2"] \arrow[dr, "1/2"'] & & \\
 & (2\oplus 3) \arrow[r, "1/2"] \arrow[dr, "1/2"'] & 2 \\
 & & 3
\end{tikzcd}$$

- Program 2: use biased choice:

$$\begin{tikzcd}[column sep=small]
 & 1 & \\
1 \overset{1/3}{\oplus} (2\oplus 3) \arrow[ur, "1/3"] \arrow[dr, "2/3"'] & & \\
 & (2\oplus 3) \arrow[r, "1/2"] \arrow[dr, "1/2"'] & 2 \\
 & & 3
\end{tikzcd}$$

Both programs express the **same** stochastic model, but program 2 is **more efficient**

$\rightarrow$ **replacing program 1 by program 2 should be an improvement**

Probabilistic Lazy PCF

The Language Probabilistic Lazy PCF

Expressions and Types

Expressions:

$$s, t \in \mathit{Exp} \quad ::= x \mid \lambda x.s \mid (s\ t) \mid \texttt{fix}\ s \mid \texttt{if}\ s\ \texttt{then}\ t_1\ \texttt{else}\ t_2 \mid \texttt{let}\ x = s\ \texttt{in}\ t$$

$$n \mid \texttt{pred}\ s \mid \texttt{succ}\ s \mid (s \overset{p}{\oplus} t) \text{ where } n \in \mathbb{N} \text{ and } p \in (0,1)$$

Types: $\tau, \sigma \in \mathit{Typ} ::= \mathsf{nat} \mid \tau \to \sigma$

- natural numbers $n : \mathsf{nat}$, `succ`, `pred` as operations
- branching via `if-then-else`
- call-by-need via let-sharing
- `fix` for recursion
- probabilistic choice with real-valued probability p

Call-by-Need Standard Reduction: Excerpt

Selected rules

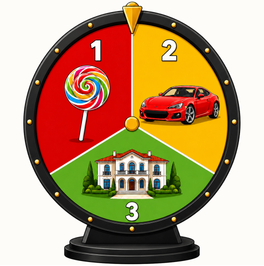
(*sr,lbeta*)	$R[(\lambda x.s)\ t] \xrightarrow{sr} R[\mathtt{let}\ x = t\ \mathtt{in}\ s]$	
(*sr,if-0*)	$R[\mathtt{if}\ 0\ \mathtt{then}\ s\ \mathtt{else}\ t] \xrightarrow{sr} R[s]$	
(*sr,if-not-0*)	$R[\mathtt{if}\ n\ \mathtt{then}\ s\ \mathtt{else}\ t] \xrightarrow{sr} R[t]$ if $n > 0$	probability
(*sr,probl*)	$R[s \overset{p}{\oplus} t] \xrightarrow{sr} R[s]$	p
(*sr,probr*)	$R[s \overset{p}{\oplus} t] \xrightarrow{sr} R[t]$	$1 - p$
...	...	

Call-by-need-strategy implemented by reduction contexts:

$$R ::= \mathrm{LR}[A] \mid \mathrm{LR}[\mathtt{let}\ x = A\ \mathtt{in}\ R[x]] \qquad \mathrm{LR} ::= [\cdot] \mid \mathtt{let}\ x = s\ \mathtt{in}\ \mathrm{LR}$$

$$A ::= [\cdot] \mid (A\ s) \mid \mathtt{if}\ A\ \mathtt{then}\ s\ \mathtt{else}\ t \mid \mathtt{pred}\ A \mid \mathtt{succ}\ A \mid \mathtt{fix}\ A$$

Wheel of Fortune, Programmed in Probabilistic Lazy PCF



- Program 1: uniformly draw from $\{0,1,2,3\}$, restart on 0:

$$s_1 := \texttt{fix}\ (\lambda f.\ \texttt{let}\ r = (0 \oplus 1) \oplus (2 \oplus 3)\ \texttt{in}\ \texttt{if}\ r\ \texttt{then}\ f\ \texttt{else}\ r)$$

- Program 2: use biased choice:

$$s_2 := 1 \overset{1/3}{\oplus} (2 \overset{1/2}{\oplus} 3)$$

Two Notions of Improvement

Contextual Cost-Improvement

Contextual Cost Improvement

$$s \preceq_{ci} t \quad \text{iff} \quad \underbrace{\forall C[\cdot]{:}\mathrm{nat} : \mathbb{P}(C[s]{\downarrow}) = \mathbb{P}(C[t]{\downarrow})}_{s \sim_c t \text{ (contextual equivalence)}} \quad \text{and} \quad \forall C[\cdot]{:}\mathrm{nat} : \underbrace{\mathbb{E}[\mathrm{L}(C[s])] \geq \mathbb{E}[\mathrm{L}(C[t])]}_{\mathit{Cost}(C[s]) \geq \mathit{Cost}(C[t])}$$

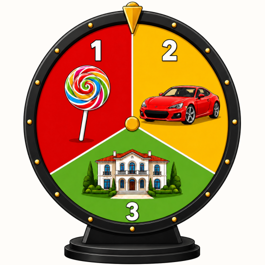
expected convergence $\mathbb{P}(r{\downarrow}) \in [0, 1]$

is the probability that r evaluates to a WHNF

expected evaluation length $\mathbb{E}[\mathrm{L}(r)] \in [0, \infty]$

is the expected number of essential reduction steps of the evaluations $r \xrightarrow{sr,*}$ WHNF

Wheel of Fortune: Expected Convergence and Expected Evaluation Length



$$s_1 := \texttt{fix}\ (\lambda f.\ \texttt{let}\ r = (0 \oplus 1) \oplus (2 \oplus 3)\ \texttt{in if}\ r\ \texttt{then}\ f\ \texttt{else}\ r)$$

$$s_2 := 1 \overset{1/3}{\oplus} (2 \overset{1/2}{\oplus} 3)$$

expected convergence

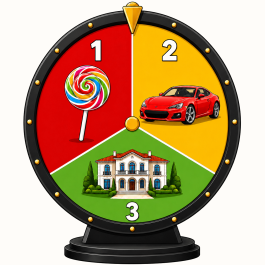
$$\mathbb{P}(s_1{\downarrow}) = \mathbb{P}(s_2{\downarrow}) = 1$$

expected evaluation length

$$\mathbb{E}[\mathrm{L}(s_1)] \geq 3 \sum_{k \geq 1} \frac{2k}{4^k} = \frac{8}{3} \text{ and } \mathbb{E}[\mathrm{L}(s_2)] = \frac{5}{3}$$

This shows $s_2 \not\preceq_{ci} s_1$, but for $s_1 \preceq_{ci} s_2$ we have to reason about $C[s_1]$ and $C[s_2]$ **for all** C.

Wheel of Fortune: Expected Convergence and Expected Evaluation Length



$$s_1 := \texttt{fix}\ (\lambda f.\ \texttt{let}\ r = (0 \oplus 1) \oplus (2 \oplus 3)\ \texttt{in}\ \texttt{if}\ r\ \texttt{then}\ f\ \texttt{else}\ r)$$

$$s_2 := 1 \overset{1/3}{\oplus} (2 \overset{1/2}{\oplus} 3)$$

expected convergence

$$\mathbb{P}(s_1\!\downarrow) = \mathbb{P}(s_2\!\downarrow) = 1$$

expected evaluation length

$$\mathbb{E}[\mathrm{L}(s_1)] \geq 3 \sum_{k \geq 1} \frac{2k}{4^k} = \frac{8}{3} \text{ and } \mathbb{E}[\mathrm{L}(s_2)] = \frac{5}{3}$$

This shows $s_2 \not\preceq_{ci} s_1$, but for $s_1 \preceq_{ci} s_2$ we have to reason about $C[s_1]$ and $C[s_2]$ **for all** C.

Inspecting the Distribution: Expectations per Number

Distribution-Cost Improvement

For closed expressions s and t of type nat:

$$s \preceq_{di} t \quad \text{iff} \quad \underbrace{\forall n \in \mathbb{N} : \mathbb{P}(s{\downarrow}n) = \mathbb{P}(t{\downarrow}n)}_{s \sim_d t \text{ (distribution equivalence)}} \text{ and } \underbrace{\forall n \in \mathbb{N} : \mathbb{E}[\mathrm{L}(s) \mid s{\downarrow}n] \geq \mathbb{E}[\mathrm{L}(t) \mid t{\downarrow}n]}_{\forall n \in \mathbb{N} : \mathrm{Cost}(s,n) \geq \mathrm{Cost}(t,n)}$$

expected value convergence $\mathbb{P}(r{\downarrow}n)$ is the probability that r converges with number n

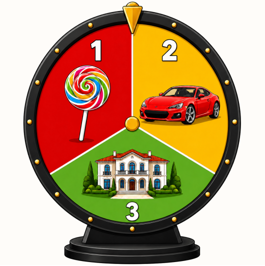
"says how likely result n is"

conditional expected length, $\mathbb{E}[\mathrm{L}(r) \mid r{\downarrow}n]$ is the expected number of evaluation steps to value n given that r evaluates to n

"says how expensive it is, conditional on obtaining n."

Advantage: $\preceq_{di}$ is defined without context quantification!

The Example with Distribution-Cost



$$s_1 := \texttt{fix}\ (\lambda f.\ \texttt{let}\ r = (0 \oplus 1) \oplus (2 \oplus 3)\ \texttt{in}\ \texttt{if}\ r\ \texttt{then}\ f\ \texttt{else}\ r)$$

$$s_2 := 1 \overset{1/3}{\oplus} (2 \overset{1/2}{\oplus} 3)$$

$\mathbb{P}(s_1{\downarrow}n) = \frac{1}{3}$ for $n \in \{1,2,3\}$

$\mathbb{P}(s_2{\downarrow}n) = \frac{1}{3}$ for $n \in \{1,2,3\}$

Thus: $s_1 \sim_d s_2$

$\mathbb{E}[\mathrm{L}(s_1) \mid s_1{\downarrow}n] \geq 3 \sum_{k \geq 1} \frac{2k}{4^k} = \dfrac{8}{3}$ for $n \in \{1,2,3\}$

$\mathbb{E}[\mathrm{L}(s_2) \mid s_2{\downarrow}1] = 1$, $\mathbb{E}[\mathrm{L}(s_2) \mid s_2{\downarrow}n] = 2$ for $n \in \{2,3\}$

Thus: $s_1 \preceq_{di} s_2$

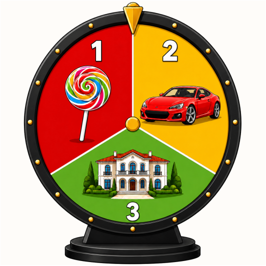
Main Result

Main Result

Main Theorem

For closed s, t : nat, the contextual cost improvement coincides with the distribution-cost improvement, i.e.

$$s \preceq_{ci} t \iff s \preceq_{di} t.$$



$s_1 := \texttt{fix}\ (\lambda f.\ \texttt{let}\ r = (0 \oplus 1) \oplus (2 \oplus 3)\ \texttt{in if}\ r\ \texttt{then}\ f\ \texttt{else}\ r)$

$s_2 := 1 \overset{1/3}{\oplus} (2 \overset{1/2}{\oplus} 3)$

Since $s_1 \preceq_{di} s_2$, the main theorem shows $s_1 \preceq_{ci} s_2$

Notes on the Proof of the Main Theorem

Claim: For closed s, t:nat : $s \preceq_{ci} t \iff s \preceq_{di} t$.

- From previous work $\sim_d = \sim_c$, so the proof mainly has to show the length parts
- $s \preceq_{ci} t \Rightarrow s \preceq_{di} t$ is the easier part
- $s \preceq_{di} t \Rightarrow s \preceq_{ci} t$ is the hard part.
  Show $s \preceq_{di} t \implies \forall C : C[s] \preceq_{di} C[t]$ as follows
  1. Show $s \preceq_{di} t \implies \forall R : R[s] \preceq_{di} R[t]$ using a so-called s-first strategy
  2. Use a context lemma to lift this to arbitrary contexts.
     Proof uses multi-contexts, bounded measures (bound on the number of prob-steps) and an asymmetric claim, i.e. one core inequation is

$$\mathbb{E}[\mathrm{L}(C[s, \dots, s]); C[s, \dots, s]{\downarrow}n] \geq \mathbb{E}^{\leq k}[\mathrm{L}(C[t, \dots, t]); C[t, \dots, t]{\downarrow}n] \quad \text{for all } n, k.$$

Consequences: Transformations

Further Transformations

For expressions of type nat:

Commutativity: $s \overset{p}{\oplus} t \asymp_{di} t \overset{1-p}{\oplus} s$

Idempotency: $s \overset{p}{\oplus} s \preceq_{di} s$

Distributivity: $(r \overset{p}{\oplus} s) \overset{q}{\oplus} (r \overset{p}{\oplus} t) \preceq_{di} r \overset{p}{\oplus} (s \overset{q}{\oplus} t)$

Garbage collection: $\texttt{let}\ x{=}s\ \texttt{in}\ t \asymp_{di} t$ if $x \notin FV(t)$

Let-float: $A^1[\texttt{let}\ x{=}s\ \texttt{in}\ t] \asymp_{di} \texttt{let}\ x{=}s\ \texttt{in}\ A^1[t]$

where $A^1 ::= ([\cdot]\ s) \mid \texttt{if}\ [\cdot]\ \texttt{then}\ s\ \texttt{else}\ t \mid \texttt{pred}\ [\cdot] \mid \texttt{succ}\ [\cdot] \mid \texttt{fix}\ [\cdot]$

Surface unique copying: $\texttt{let}\ x{=}t\ \texttt{in}\ C[x] \asymp_{di} C[t]$

if the hole of C is not in an abstraction and x is fresh for C, t

where $s \asymp_{di} t$ iff $s \preceq_{di} t$ and $t \preceq_{di} s$

Introducing Sharing can be Unsound

Sharing a choice changes the distribution

Independent sampling (two coin flips):

$$r_1 = \texttt{if } 0 \oplus 1 \texttt{ then } 0 \oplus 1 \texttt{ else } 2$$

$$\mathbb{P}(r_1{\downarrow}0) = \mathbb{P}(r_1{\downarrow}1) = \tfrac{1}{4}, \quad \mathbb{P}(r_1{\downarrow}2) = \tfrac{1}{2}$$

Shared sampling (one coin flip):

$$r_2 = \texttt{let } x = 0 \oplus 1 \texttt{ in if } x \texttt{ then } x \texttt{ else } 2$$

$$\mathbb{P}(r_2{\downarrow}0) = \tfrac{1}{2}, \quad \mathbb{P}(r_2{\downarrow}1) = 0, \quad \mathbb{P}(r_2{\downarrow}2) = \tfrac{1}{2}$$

Sharing forces x to be 0 in the `then`-branch:
result 1 becomes impossible, hence $r_1 \not\sim_d r_2$.

→ Common subexpression elimination is sound only under restrictions, e.g. for deterministic subexpressions

Conclusion

Conclusion

- Improvement theory captures the optimisation problem for stochastic models: preserve the distribution, reduce expected cost
- The main theorem gives an intrinsic characterisation, without context quantification:

$$s \preceq_{ci} t \iff s \preceq_{di} t$$

- It makes improvements checkable: comparing two distributions and, per result n, the conditional expected lengths suffices
  $\rightarrow$ correctness of commutativity, idempotency, distributivity, gc, let-float, ...
- Probabilistic call-by-need is a distinctive semantic setting because laziness and sharing interact non-trivially with random choice

Future Work

- Lift the results from expressions of number type to all types, including higher-order types
- Introduce work-decorations, to enable equational reasoning
- Explore further transformations
- Investigate common subexpression elimination for deterministic subterms
- Study further cost models, like e.g. space consumption
- Study different / extended languages

Thank You!