A rewriting theory for quantum lambda-calculus

CSL 2025

Claudia FAGGIAN , Benoît VALIRON , Gaëtan LOPEZ

Université Paris Cité , Université Paris Saclay

Febuary 11, 2025

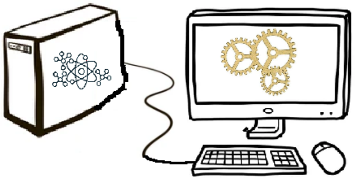
Quantum λ-calculus = **quantum** functional programming language

The quantum λ-calculus by **Selinger and Valiron** lies arguably at the **fundation of the development of higher-order quantum programming languages**.

A critical point :

- **non-duplicability** of quantum data, which is addressed by tools from **linear logic**

Selinger-Valiron's language used the QRAM model : quantum data + classical control



Lambda calculus

Lambda calculus has a rich, powerful notion of reduction, whose properties are studied by a **vast amount of literature**.

Rewriting theory provides a sound framework for reasoning about :

- **programs transformations,** such as compiler optimizations or parallel implementations
- **program equivalence**

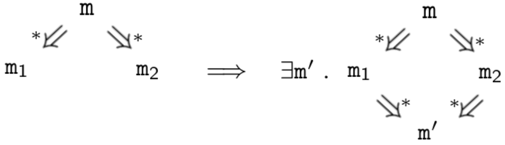
λ

λx . x

```
fn(x) -> x end
```

The pillars of the operational theory of λ-calculus

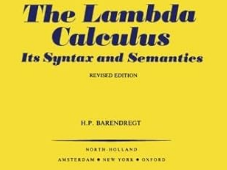
- **Confluence** guarantees that **normal forms are unique** :



- **Standardization (factorization):** computational steps can be reorganized.

$$M \to_{\beta}^{*} N \Rightarrow M \to_{h}^{*} \cdot \to_{\neg h}^{*} N$$

Hence **normalization** :
if a normal form exists, there is a **strategy** which is *guaranteed* to find it.



CALL-BY-NAME, CALL-BY-VALUE AND THE λ-CALCULUS

G. D. PLOTKIN

Department of Machine Intelligence, School of Artificial Intelligence, University of Edinburgh, Edinburgh, United Kingdom

Communicated by R. Milner
Received 1 August 1974

Call-by-name λ-calculus "à la Barendregt"	**Call-by-value λ_v-calculus "à la Plotkin"**
General reduction: $(\to_\beta)$ Evaluation: head $(\to_h)$	General reduction: $(\to_{\beta_v})$ Evaluation: weak-left $(\to_l)$
1. Head factorization (standardization): $M \to_\beta^* N$ then $M \to_h^* \cdot \to_{\neg h}^* N$ *2. Head normalization*: $M \to_\beta^* H$ then $M \to_h^* H'$	*1. Weak-left factorization (standardization)*: $M \to_{\beta_v}^* N$ then $M \to_l^* \cdot \to_{\neg l}^* N$ *2. Convergence to a value*: $M \to_{\beta_v}^* V$ then $M \to_l^* V'$

As pioneered by Plotkin (74), **standardization** allows to bridge between

- the **general reduction,** where **programs transformations** can be studied
- an **evaluation strategy**, implementing **execution of (idealized) programming language**.

What about these results in the quantum setting ?

We move from classical to quantum computation

Classical bit :

Quantum bit (**qubits**), **superposition** of data :

Qubits are

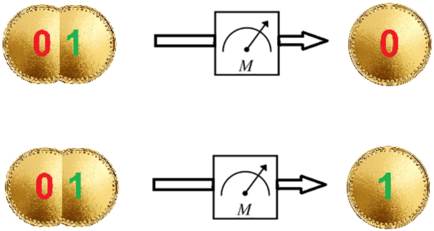
- **non-duplicable**
- **non-erasable**

Prime example: Shor Algorithm, finding the prime factors of an integer in polynomial time.

Quantum operations

- unitary operation

- measurement



Rewriting theory for quantum λ-calculus: state of the art

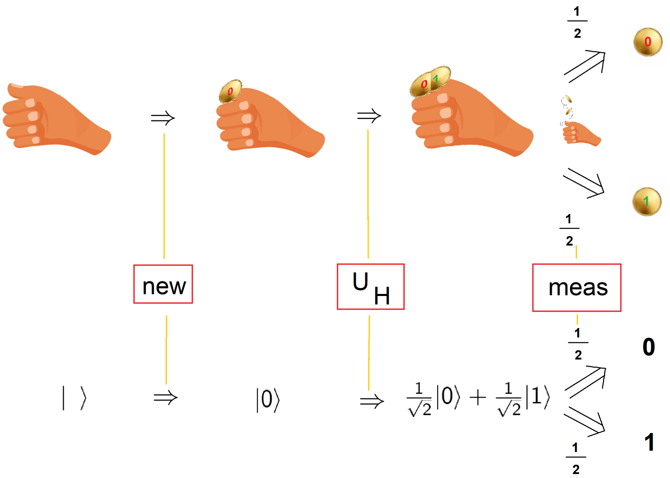
- **Confluence**:
  - without measurement : studied, since early work [Dal Lago-Masini-Zorzi 2009, Arrighi-Dowek 2017]
  - **with measurement** : studied with a **sophisticated approach** [Dal Lago-Masini-Zorzi 2011]

⚠ When dealing with measurement, the analysis is far more challenging

- **Standardization/normalization**:
  - **No results** (like the classical ones) in the literature for the **quantum setting**.

⚠ Challenge : dealing with probabilistic and asymptotic behavior

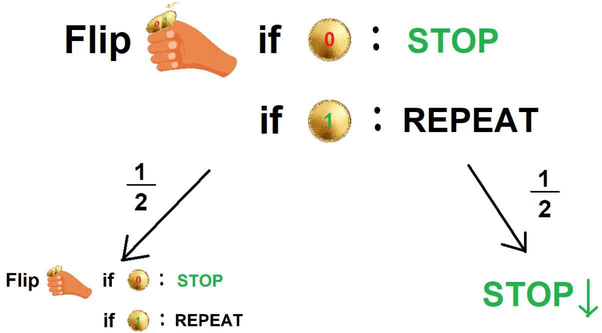
Let us flip a (fair) **quantum coin**



Measurement introduces probabilities.

Even termination becomes probabilistic !

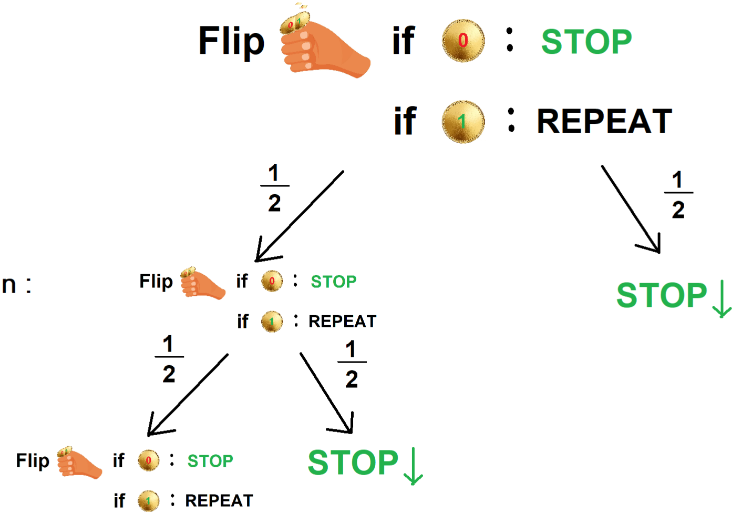
Even termination becomes probabilistic !



After, 1 step of computation :

$P_1(\text{STOP}) = \frac{1}{2}$

Even termination becomes probabilistic !



After 2 step of computation :

$$P_2(\text{STOP}) = \frac{1}{2} + \frac{1}{2} \times \frac{1}{2}$$

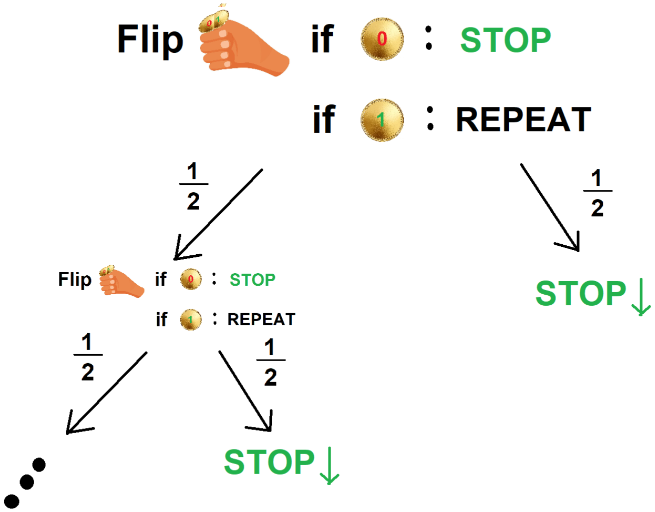
$$= \frac{3}{4}$$

Even termination becomes probabilistic !

After n step of computation :

$$P_n(\text{STOP}) = \sum_{k=1}^{n} \left(\frac{1}{2}\right)^k$$

$$= 1 - \left(\frac{1}{2}\right)^n$$



Asymptotic and quantum behaviour

Notice that:

- $\forall n,\ P_n(STOP) \neq 1$ (does not terminate in finite time)
- $\lim_{n \to +\infty} P_n(STOP) = 1$ (almost sure termination)

More challenges, induced by quantum computation :

- **manage the states of the quantum memory**
- **control the duplication/erasability inside λ-terms**

New tools in probabilistic rewriting

Recent advances in the theory of **probabilistic rewriting** give us a way to tackle this task with tools unavailable a decade ago **[Dal Lago-Faggian-Valiron-Yoshimizu 2017, DíazCaro-Guido Martinez 2018, Avanzini-Dal Lago-Yamada 2020, Faggian 2022]**.

- **It is time to develop such theory!**

Standardization and Asymptotic Normalization: the proof technique

Quantum λ-calculus [Selinger-Valiron 2004]

- **Confluence**: large body of early work
- [Dal Lago et al. MSCS 2009] (measure-free): reordering of some steps to put in a "**standard**" form

No "standard" *standardization* results

Probabilistic λ-calculus

- Probabilistic **rewriting**
- Probabilistic **λ**-calculus [Faggian-Ronchi, Lics 2019]: confluence and standardization

Proof techniques for modularizing standardization:
[Accattoli-Guerrieri-Faggian, CSL 2021]

Proof techniques for Asymptotic Normalization:
[Faggian, LMCS 2022]
Faggian-Guerrieri, FSCD 2022]

A rewriting theory for quantum **λ**-calculus (CSL 25)

Main results

In this work, we studied not only confluence but also standardization and normalization results for a quantum λ-calculus featuring measurement.

Contributions :

- an untyped quantum lambda-calculus, designed to allow for a more general reduction
- a rich operational semantics
- confluence
- standardization
- normalization result that scales to the asymptotic case

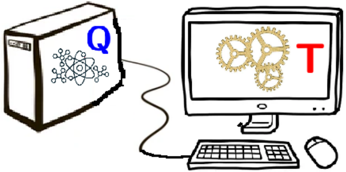
Quantum programs

Selinger and Valiron follows the paradigm :

- quantum data
- classical control

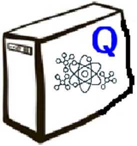
A quantum program is :

- a **Q**uantum memory (data)
- a **T**erm (control)



[Q ; T]

Quantum memory



[**Q** ; **T**]

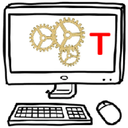
Quantum memory

The *state* Q of a quantum memory, consisting of n qubits, is a normalized vector $\in (\mathbb{C}^2)^{\otimes n}$

Example

- $|+\rangle := \frac{\sqrt{2}}{2}(|0\rangle + |1\rangle)$
- $|\phi^+\rangle := \frac{\sqrt{2}}{2}|00\rangle + \frac{\sqrt{2}}{2}|11\rangle$

Term



[Q ; T]

Terms (built on Simpson Calculus)

$$M, N, P ::= x \mid !M \mid \lambda^! x.M \mid MN \qquad \text{(Call-by-Push-Value } \lambda\text{-calculus)}$$

$$\mid \lambda x.M \quad \textbf{linear} \text{ abstraction (allows us to deal with qubits)}$$

$$\mid r_i \mid U_A \mid \texttt{new} \mid \texttt{meas}(P, M, N) \quad \text{(quantum)}$$

The thunk **!M** freezes a computation, so **can be duplicated** and **erased**.

Operational semantics (aka reduction)

The **general reduction** is made of :

- **beta** reductions $\to_\beta$ **unconstrained**
- **quantum** reductions $\to_q$ **constrained to be surface** (*i.e. not in the scope of thunk !M*)
  to avoid duplication

q-reduction

Quantum-reductions :

- initializing a qubit in the memory

$$\overline{[\mathtt{Q}; \mathtt{new}] \mapsto_q \{\!\{ [\mathtt{Q}\otimes|0\rangle; r_n] \}\!\}}$$

- apply unitary to the memory

$$\overline{[\mathtt{Q}; U_A\ r_0] \mapsto_q \{\!\{ [(A \otimes \mathrm{Id})\mathtt{Q}; r_0] \}\!\}}$$

$$\overline{[\mathtt{Q}; (U_A\ \langle r_0, r_1\rangle] \mapsto_q \{\!\{ [(A \otimes \mathrm{Id})\mathtt{Q}; \langle r_0, r_1\rangle] \}\!\}}$$

- measure the n-th qubit

$$\overline{[\mathtt{Q}; \mathtt{meas}(r_n, M, N)] \mapsto_q \{\!\{ |\alpha_0|^2[\mathtt{Q}_0; M], |\alpha_1|^2[\mathtt{Q}_1; N] \}\!\}}$$

Measure is probabilistic:
a program reduces to a distribution of programs

Lift to rewriting on (multi)-distributions

Lifting [Avanzini Dal Lago Yamada 2020]

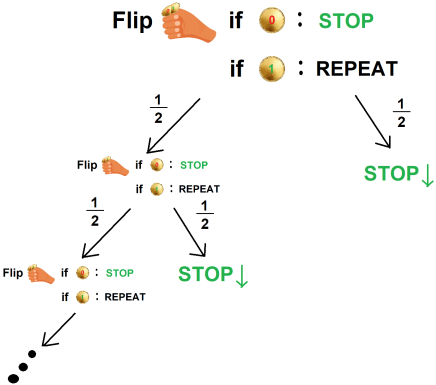
$$\frac{}{\{\!\{\, \mathbf{p} \,\}\!\} \Rightarrow \{\!\{\, \mathbf{p} \,\}\!\}} \qquad \frac{\mathbf{p} \to \mathtt{m}}{\{\!\{\, \mathbf{p} \,\}\!\} \Rightarrow \mathtt{m}} \qquad \frac{(\{\!\{\, \mathbf{p}_i \,\}\!\} \Rightarrow \mathtt{m}_i)_{i \in I}}{\{\!\{\, p_i \mathbf{p}_i \mid i \in I \,\}\!\} \Rightarrow \sum_{i \in I} p_i \cdot \mathtt{m}_i}$$

Expressivity

Repeatedly flippling a quantum coin can be encoded as :

$$\Delta!\Delta \text{ where } \Delta := \lambda^! x.\,\mathtt{meas}(U_H\,\mathtt{new}, I, x!x)$$

$$\{\!\{\,[\;|\rangle ; \underline{\Delta!\Delta}]\,\}\!\}$$



$\mathbb{P}$ - **termination**

$$\{\!\{ [\ |\rangle ; \Delta!\Delta] \}\!\} \qquad \mathbf{0}$$

$$\Downarrow \quad \vdots \quad \Downarrow$$

$$\{\!\{ \tfrac{1}{2}[|\rangle ; \Delta!\Delta],\ \tfrac{1}{2}[|\rangle ; I] \}\!\} \qquad \frac{1}{2}$$

$$\Downarrow \quad \vdots \quad \Downarrow$$

$$\{\!\{ \tfrac{1}{4}[|\rangle ; \Delta!\Delta],\ \tfrac{1}{4}[|\rangle ; I],\ \tfrac{1}{2}[|\rangle ; I] \}\!\} \qquad \frac{3}{4}$$

$$\Downarrow \quad \vdots$$

Evaluation: surface reduction

Surface reduction $\underset{s}{\Rightarrow}$:

both beta and quantum steps are constrained to be surface.

Surface reduction plays similar role to head reduction (in CbN) or left-weak reduction (in CbV)

Call-by-name λ-calculus "à la Barendregt"	**Call-by-value λ_v-calculus "à la Plotkin"**
General reduction: $(\rightarrow_\beta)$ Evaluation: head $(\rightarrow_h)$	General reduction: $(\rightarrow_{\beta_v})$ Evaluation: weak-left $(\rightarrow_l)$
1. *Head factorization (standardization)*: $M \rightarrow_\beta^* N$ then $M \rightarrow_h^* \cdot \rightarrow_{\neg h}^* N$ 2. *Head normalization*: $M \rightarrow_\beta^* H$ then $M \rightarrow_h^* H'$	1. *Weak-left factorization (standardization)*: $M \rightarrow_{\beta_v}^* N$ then $M \rightarrow_l^* \cdot \rightarrow_{\neg l}^* N$ 2. *Convergence to a value*: $M \rightarrow_{\beta_v}^* V$ then $M \rightarrow_l^* V'$

First key results: **Finitary** properties

Confluence of the general reduction

$$\begin{array}{ccc} & \mathtt{m} & \\ {}^{*}\swarrow & & \searrow^{*} \\ \mathtt{m}_1 & & \mathtt{m}_2 \end{array} \quad \Longrightarrow \quad \exists \mathtt{m}' \,.\; \begin{array}{ccc} & \mathtt{m} & \\ {}^{*}\swarrow & & \searrow^{*} \\ \mathtt{m}_1 & & \mathtt{m}_2 \\ \searrow^{*} & & {}^{*}\swarrow \\ & \mathtt{m}' & \end{array}$$

Surface Standardization

$$\mathtt{m} \Rightarrow^{*} \mathtt{n} \quad \Longleftrightarrow \quad \exists \mathtt{m}' \;.\; \mathtt{m} \underset{\mathtt{s}}{\Rightarrow}^{*} \mathtt{m}' \underset{\neg\mathtt{s}}{\Rightarrow}^{*} \mathtt{n}$$

Standardization allows us to prove a normalization result which scales to probabilistic termination.
Let first see what is the challenge.

Probabilistic Termination of a reduction sequence

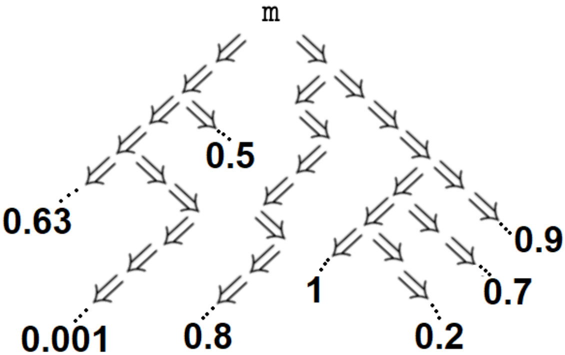
The **probability** that a program **reaches surface normal form** along a reduction sequence is **a limit**.

	$\mathbb{P}$ - termination
$\{\!\{ [\ \rvert\rangle ; \Delta!\Delta] \}\!\}$	**0**
$\Downarrow$ $\vdots$ $\Downarrow$	
$\{\!\{ \frac{1}{2}[\rvert\rangle ; \Delta!\Delta], \frac{1}{2}[\rvert\rangle ; I] \}\!\}$	$\frac{1}{2}$
$\Downarrow$ $\vdots$ $\Downarrow$	
$\{\!\{ \frac{1}{4}[\rvert\rangle ; \Delta!\Delta], \frac{1}{4}[\rvert\rangle ; I], \frac{1}{2}[\rvert\rangle ; I] \}\!\}$	$\frac{3}{4}$
$\Downarrow$ $\vdots$	

$$\lim_{n \to +\infty} P_n = \sum_{n=0}^{\infty} \left(\frac{1}{2}\right)^n = 1$$

Challenge

Since reduction is general, there are **many reduction sequences**, thus **many limits** !



Main result in the paper:
Surface Evaluation is a strategy for asymptotic normalization

Theorem (Asymptotic normalization)

There is a strategy $\underset{s}{\rightrightarrows}$ that is **guaranteed to converge** with the **greatest probability**.

These are the first steps, much more work to do!

Work in progress: a finer notion of asymptotic result

If quantum program computes a quantum register...
at the limit the (asymptotic) result is a distribution of quantum registers: a density matrix.

We expect the normalization results to scale.