

Towards a Curry-Howard Correspondence for Quantum Computation

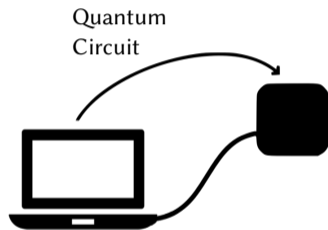
Kostia Chardonnet

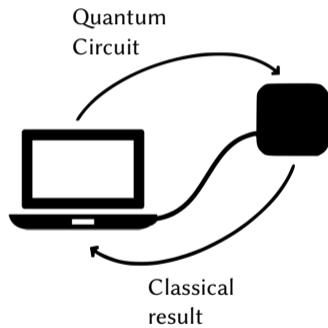
Univ. Paris Saclay, LMF, Quacs
Univ. Paris Centre, IRIF

PhD Defense, 09/01/2023

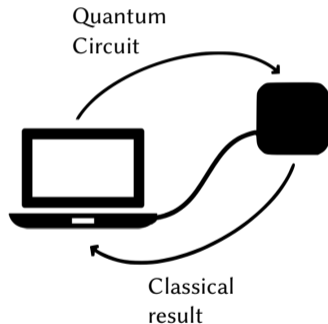
Supervised by: Pablo Arrighi, Alexis Saurin, Benoît Valiron



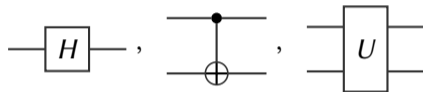




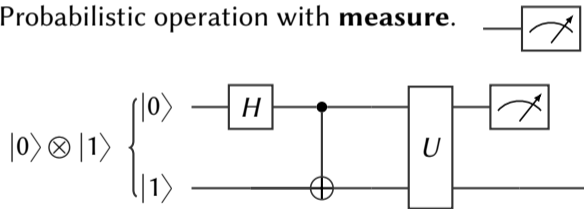
Two operations



- Unitary operations **inside** the coprocessor.

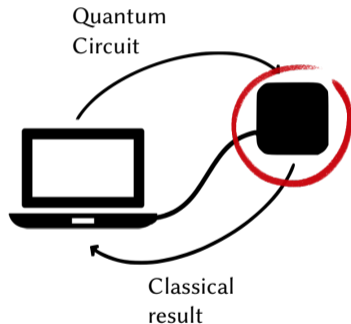


- Probabilistic operation with **measure**.

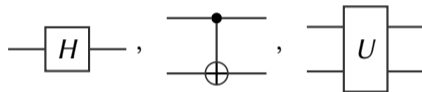


- Available data **inside coprocessor** : \otimes^n qubit.

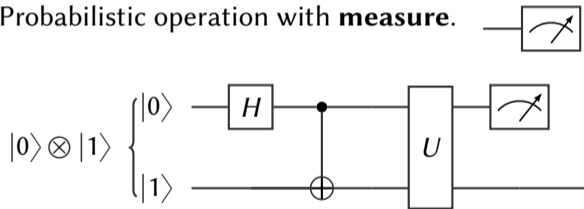
Two operations



- Unitary operations **inside** the coprocessor.



- Probabilistic operation with **measure**.



- Available data **inside coprocessor** : \otimes^n qubit.

Classical	Quantum
0	$ 0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
1	$ 1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
(0, 1)	$ 0\rangle \otimes 1\rangle$

$$\alpha |0\rangle + \beta |1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

Classical	Quantum
0	$ 0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
1	$ 1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
(0, 1)	$ 0\rangle \otimes 1\rangle$

$$\alpha |0\rangle + \beta |1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

- **Unitary Operations:** can be reversed.

- CNOT:

$$\begin{array}{c} \bullet \\ \text{---} \\ | \\ \oplus \\ \text{---} \end{array} = \begin{cases} |0\rangle \otimes |x\rangle \mapsto |0\rangle \otimes |x\rangle \\ |1\rangle \otimes |x\rangle \mapsto |1\rangle \otimes |\neg x\rangle \end{cases}$$

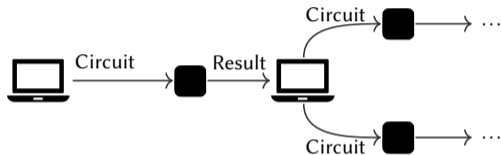
- Hadamard:

$$\text{---} \boxed{H} \text{---} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \begin{cases} |0\rangle \mapsto \frac{|0\rangle + |1\rangle}{\sqrt{2}} \\ |1\rangle \mapsto \frac{|0\rangle - |1\rangle}{\sqrt{2}} \end{cases}$$

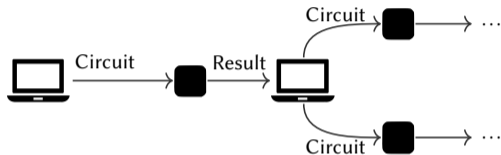
- **Non-Cloning Principle.** ~~$|x\rangle \mapsto |x\rangle \otimes |x\rangle$~~

From Classical to Quantum Control Flow

Classical Control Flow



Classical Control Flow



Quantum Control Flow

$$\text{QSwitch}(x, y, U, V) = \begin{cases} VU(y) & \text{if } x = |0\rangle \\ UV(y) & \text{if } x = |1\rangle \end{cases}$$

$$(\alpha |0\rangle + \beta |1\rangle) \otimes |y\rangle \mapsto \alpha |0\rangle \otimes (UV|y\rangle) + \beta |1\rangle \otimes (VU|y\rangle).$$

Physically implementation but **not in co-processor**.

Classical

- Bit = $1 \oplus 1$.

QRAM

- Qubit is **opaque**.

Classical

- Bit = $\mathbb{1} \oplus \mathbb{1}$.
- Rich type system.

QRAM

- Qubit is **opaque**.
- Only **tensors**.

Classical

- Bit = $1 \oplus 1$.
- Rich type system.
- Classical control flow.

QRAM

- Qubit is **opaque**.
- Only **tensors**.
- No **quantum control flow**.

Classical

- Bit = $\mathbb{1} \oplus \mathbb{1}$.
- Rich type system.
- Classical control flow.

QRAM

- Qubit is **opaque**.
- Only **tensors**.
- No **quantum control flow**.

This thesis

Develop a new model of quantum computation featuring

- **A richer type system** (inductive);
- **with quantum control flow**.

Classical

- Bit = $\mathbb{1} \oplus \mathbb{1}$.
- Rich type system.
- Classical control flow.

QRAM

- Qubit is **opaque**.
- Only **tensors**.
- No **quantum control flow**.

This thesis

Develop a new model of quantum computation featuring

- **A richer type system** (inductive);
- **with quantum control flow**.

Approach : Curry-Howard Correspondence.

Types

- Type = Description of a data.
product, choice, fonctions, ...
- Invariant on the structure of computation.
- Ensure **safety properties**.

The Curry-Howard Correspondence

Types

- Type = Description of a data.
product, choice, functions, ...
- Invariant on the structure of computation.
- Ensure **safety properties**.

Logic

- Formulas = Mathematical statements
AND, OR, IMP, ...
- Study of **mathematical reasoning**.
- Focus on **propositions and their proofs**.

The Curry-Howard Correspondence

Types

- Type = Description of a data.
product, choice, fonctions, ...
- Invariant on the structure of computation.
- Ensure **safety properties**.

Logic

- Formulas = Mathematical statements
AND, OR, IMP, ...
- Study of **mathematical reasoning**.
- Focus on **propositions and their proofs**.

Curry-Howard

Types	↔	Propositions
Terms	↔	Proofs
Evaluation	↔	Cut-Elimination

The Curry-Howard Correspondence

Types

- Type = Description of a data.
product, choice, functions, ...
- Invariant on the structure of computation.
- Ensure **safety properties**.

Logic

- Formulas = Mathematical statements
AND, OR, IMP, ...
- Study of **mathematical reasoning**.
- Focus on **propositions and their proofs**.

Curry-Howard

Types	↔	Propositions
Terms	↔	Proofs
Evaluation	↔	Cut-Elimination

$$\frac{f : A \rightarrow B \quad x : A}{f(x) : B}$$

The Curry-Howard Correspondence

Types

- Type = Description of a data.
product, choice, functions, ...
- Invariant on the structure of computation.
- Ensure **safety properties**.

Logic

- Formulas = Mathematical statements
AND, OR, IMP, ...
- Study of **mathematical reasoning**.
- Focus on **propositions and their proofs**.

Curry-Howard

Types	↔	Propositions
Terms	↔	Proofs
Evaluation	↔	Cut-Elimination

$$\frac{f : A \rightarrow B \quad x : A}{f(x) : B}$$

$$\frac{\begin{array}{c} \pi \\ \vdots \\ A \rightarrow B \end{array} \quad \begin{array}{c} \pi' \\ \vdots \\ A \end{array}}{B} \text{ Modus Ponens}$$

Formulas

$$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$$

Formulas

$$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

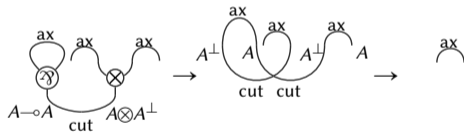
Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

Proof Nets



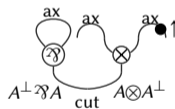
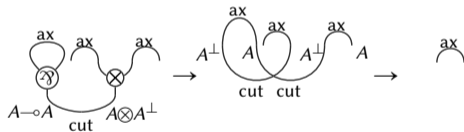
Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

Proof Nets



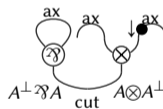
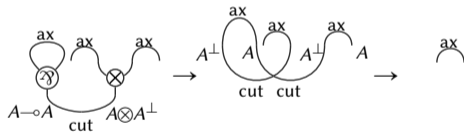
Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

Proof Nets



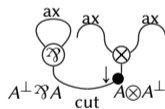
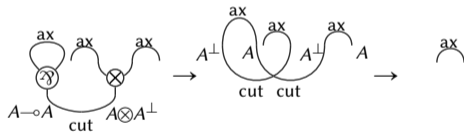
Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

Proof Nets



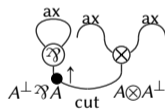
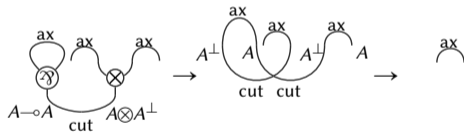
Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

Proof Nets



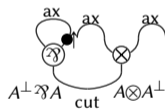
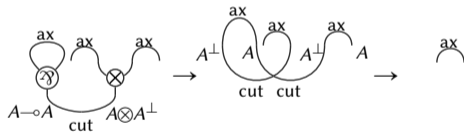
Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

Proof Nets



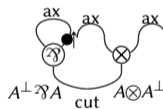
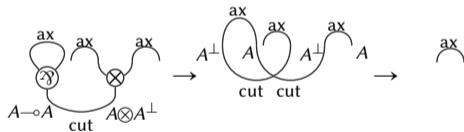
Formulas

$A, B ::= a \mid a^\perp \mid A \otimes B \mid A \wp B \mid \mathbb{1}$

$$(\lambda x.x) y \rightsquigarrow \frac{\frac{A \vdash A}{\vdash A^\perp \wp A} \wp \quad \frac{A \vdash A \quad A \vdash A}{A \vdash A, A \otimes A^\perp} \otimes}{A \vdash A} \text{cut}$$

No duplication or erasure.

Proof Nets



Additives

$A, B ::= \dots \mid A \oplus B$

\oplus represent the **action of a choice**.

$\text{Bool} = \mathbb{1} \oplus \mathbb{1}$.

Two routes for quantum types.

Classical Control

- bit = $\mathbb{1} \oplus \mathbb{1}$
- Allow duplication in a controlled way.
- Quantum λ -calculus [Selinger, Valiron'04].
- Classical control.

Quantum Control

- qubit = $\mathbb{1} \oplus \mathbb{1}$
- Inductive types
list(A) = $\mu X. \mathbb{1} \oplus (A \otimes X)$.
- Quantum Switch.

Our proposal: logic μ MALL, MALL + least and greatest fixed-point.

Towards a Curry-Howard Correspondence for Quantum Computation

I.f.p operator
(Ch. 4, CSL'23)

Pairing
(Ch. 5, MFCS'21)

$$[A] = \mu X. \mathbb{1} \oplus (A \otimes X)$$

Quantum Control
(Ch. 6, Draft)

Towards a Curry-Howard Correspondence for Quantum Computation

l.f.p operator
(Ch. 4, CSL'23)

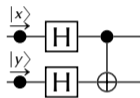
Pairing
(Ch. 5, MFCS'21)

$$[A] = \mu X. \mathbb{1} \oplus (A \otimes X)$$

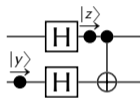
Quantum Control
(Ch. 6, Draft)

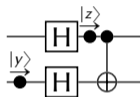
The diagram illustrates the equation $[A] = \mu X. \mathbb{1} \oplus (A \otimes X)$. Annotations include: 'l.f.p operator (Ch. 4, CSL'23)' pointing to the μ operator; 'Pairing (Ch. 5, MFCS'21)' pointing to the \otimes symbol; and 'Quantum Control (Ch. 6, Draft)' pointing to the \oplus symbol.

Token Machines for Quantum Computation



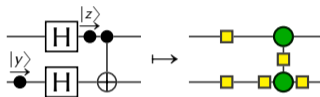
Token Machines for Quantum Computation





- MELL + Circuits [Dal Lago. et al'16].
- Require **synchronisation**.
- No superposition of **position**.
- Classical Control.

Token Machines for Quantum Computation



- MELL + Circuits [Dal Lago. et al'16].
- Require **synchronisation**.
- No superposition of **position**.
- Classical Control.
- Consider **Token Machine**.
- **Asynchronicity**.
- **Quantum Control**.

Generators



(Empty)



(Id)



(Swap)



(Cap)



(Cup)

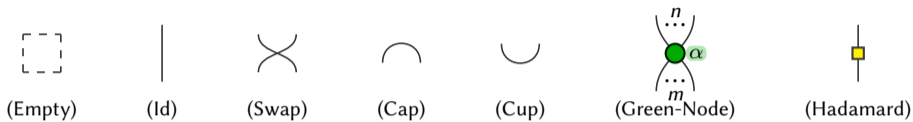


(Green-Node)



(Hadamard)

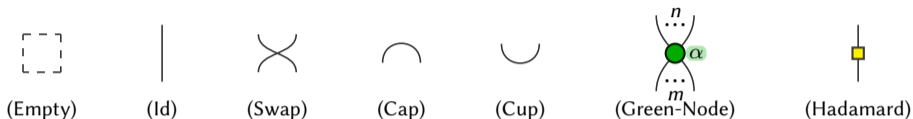
Generators



Compositions



Generators

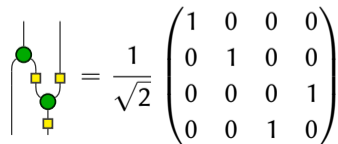


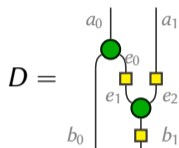
Compositions



Standard Interpretation

Linear Maps : $\mathbf{ZX} \rightarrow \mathcal{M}(\mathbb{C})$





Token

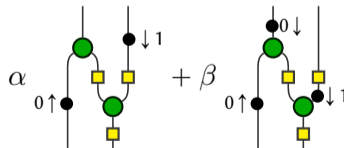
3-tuple $(e, d, b) \in \mathcal{E}(D) \times \{\uparrow, \downarrow\} \times \{0, 1\}$ where:

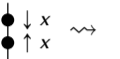
- e is an edge of the ZX-Diagram D .
- d is a direction.
- b is the state of the token.

Token State

A *token state* is a **sum** of **products** of tokens with complex coefficients.

$$\langle t | t' \rangle = \begin{cases} 1 & \text{if } t = t' \\ 0 & \text{if } t \neq t' \end{cases}$$



- Collisions : 

- Collisions : $\begin{array}{c} \bullet \\ \downarrow x \\ \bullet \\ \uparrow x \end{array} \rightsquigarrow \mid \quad \begin{array}{c} \bullet \\ \downarrow x \\ \bullet \\ \uparrow \neg x \end{array} \rightsquigarrow 0$

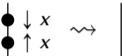
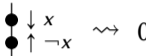
• Collisions : $\begin{array}{c} \bullet \\ \downarrow x \\ \bullet \\ \uparrow x \end{array} \rightsquigarrow \left| \begin{array}{c} \bullet \\ \downarrow x \\ \bullet \\ \uparrow -x \end{array} \rightsquigarrow 0 \right.$

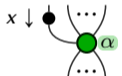
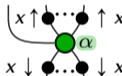
• Diffusions : $\begin{array}{c} x \downarrow \bullet \\ \curvearrowright \\ \bullet \\ \curvearrowleft \\ \dots \\ \dots \end{array} \rightsquigarrow e^{ix\alpha} \begin{array}{c} x \uparrow \bullet \dots \bullet \uparrow x \\ \curvearrowright \\ \bullet \\ \curvearrowleft \\ x \downarrow \bullet \dots \bullet \downarrow x \end{array}$



• Collisions : $\begin{array}{c} \bullet \\ \downarrow x \\ \bullet \\ \uparrow x \end{array} \rightsquigarrow \left| \begin{array}{c} \bullet \\ \downarrow x \\ \bullet \\ \uparrow -x \end{array} \right. \rightsquigarrow 0$


• Diffusions : $\begin{array}{c} x \downarrow \bullet \\ \vdots \\ \bullet \end{array} \rightsquigarrow e^{ix\alpha} \begin{array}{c} x \uparrow \bullet \dots \bullet \uparrow x \\ \vdots \\ x \downarrow \bullet \dots \bullet \downarrow x \end{array}$

$$\begin{array}{c} \bullet \\ \downarrow x \\ \square \end{array} \rightsquigarrow \frac{1}{\sqrt{2}} \left(\begin{array}{c} \square \\ \bullet \\ \downarrow 0 \end{array} + (-1)^x \begin{array}{c} \square \\ \bullet \\ \downarrow 1 \end{array} \right)$$

• Collisions :  |  $\rightsquigarrow 0$

• Diffusions :  $\rightsquigarrow e^{ix\alpha}$ 

 \rightsquigarrow 

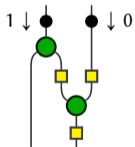
 $\rightsquigarrow \frac{1}{\sqrt{2}} \left(\begin{matrix} \text{yellow square} \\ \text{black dot} \end{matrix} \downarrow 0 + (-1)^x \begin{matrix} \text{yellow square} \\ \text{black dot} \end{matrix} \downarrow 1 \right)$

 \rightsquigarrow  ...

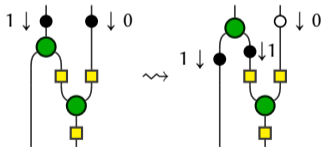
$$\frac{1}{\sqrt{2}} \text{CNOT} = \begin{array}{c} \text{---} \\ | \\ \bullet \\ \text{---} \\ \square \\ \text{---} \\ \bullet \\ \text{---} \\ \square \\ \text{---} \end{array} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \begin{array}{c} \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



$$\frac{1}{\sqrt{2}} \text{CNOT} = \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left[\text{Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{\sqrt{2}} \left(\left[\text{Circuit Diagram 1} \right] \downarrow 0 + \left[\text{Circuit Diagram 2} \right] \downarrow 1 \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left(\text{Circuit Diagram} \right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left(\text{Circuit Diagram} \right) \rightsquigarrow^* \frac{1}{\sqrt{2}} \left(\left(\text{Circuit Diagram 1} \right) + \left(\text{Circuit Diagram 2} \right) \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Quantum Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left[\text{Quantum Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{\sqrt{2}} \left(\left[\text{Quantum Circuit Diagram 1} \right] + \left[\text{Quantum Circuit Diagram 2} \right] \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left(\text{Circuit Diagram} \right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left(\text{Circuit Diagram} \right) \rightsquigarrow^* \frac{1}{\sqrt{2}} \left(\frac{1}{\sqrt{2}} \left(\text{Circuit Diagram 1} - \text{Circuit Diagram 2} \right) + \text{Circuit Diagram 3} \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left[\text{Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{2} \left(\left[\text{Circuit Diagram 1} \right] - \left[\text{Circuit Diagram 2} \right] + \left[\text{Circuit Diagram 3} \right] - \left[\text{Circuit Diagram 4} \right] \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left[\text{Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{2} \left(\left[\text{Circuit Diagram 1} \right] - \left[\text{Circuit Diagram 2} \right] + \left[\text{Circuit Diagram 3} \right] - \left[\text{Circuit Diagram 4} \right] \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left[\text{Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{2} \left(\left[\text{Circuit Diagram 1} \right] + \left[\text{Circuit Diagram 2} \right] - \left[\text{Circuit Diagram 3} \right] \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left(\text{CNOT circuit} \right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\begin{pmatrix} \text{CNOT circuit} \\ \text{CNOT circuit} \end{pmatrix} \rightsquigarrow^* \frac{1}{2} \left(\begin{pmatrix} \text{CNOT circuit} \\ \text{CNOT circuit} \end{pmatrix} - \begin{pmatrix} \text{CNOT circuit} \\ \text{CNOT circuit} \end{pmatrix} \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left[\text{Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{2} \left(\left[\text{Circuit Diagram 1} \right] - \left[\text{Circuit Diagram 2} \right] \right)$$

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

The circuit diagram shows two vertical lines. The top line has a green circle (CNOT control) and a yellow square (CNOT target). The bottom line has a green circle (CNOT control) and a yellow square (CNOT target). The lines are connected by a vertical line between the two green circles.

$$\left[\text{Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{2\sqrt{2}} \left(\left[\text{Circuit Diagram 1} \right] + \left[\text{Circuit Diagram 2} \right] - \left[\text{Circuit Diagram 3} \right] + \left[\text{Circuit Diagram 4} \right] \right)$$

The circuit diagram on the left has two input qubits, each with a black dot and a downward arrow. The first qubit has a '1' above it, and the second has a '0' above it. The circuit consists of a CNOT gate with the first qubit as control and the second as target, followed by another CNOT gate with the second qubit as control and the first as target.

The four circuit diagrams in the parentheses represent the decomposition of the CNOT gate into a sum of four terms. Each term has two input qubits with black dots and downward arrows. The first qubit has a '1' above it, and the second has either a '0' or a '1' above it. The terms are:

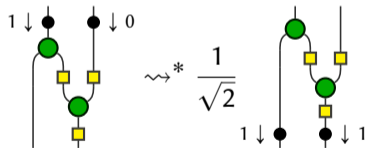
- Term 1: CNOT (1,0) with '+' sign
- Term 2: CNOT (1,1) with '+' sign
- Term 3: CNOT (1,0) with '-' sign
- Term 4: CNOT (1,1) with '+' sign

$$\frac{1}{\sqrt{2}} \text{CNOT} = \left[\text{Quantum Circuit Diagram} \right] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

$$\left[\text{Quantum Circuit Diagram} \right] \rightsquigarrow^* \frac{1}{2\sqrt{2}} \left(\begin{array}{c} \left[\text{Circuit 1} \right] + \left[\text{Circuit 2} \right] - \left[\text{Circuit 3} \right] + \left[\text{Circuit 4} \right] \\ \left(\begin{array}{c} 1 \downarrow \bullet \quad \bullet \downarrow 0 \\ 1 \downarrow \bullet \quad \bullet \downarrow 1 \\ 1 \downarrow \bullet \quad \bullet \downarrow 0 \\ 1 \downarrow \bullet \quad \bullet \downarrow 1 \end{array} \right) \end{array} \right)$$

↑ ↑

$$\frac{1}{\sqrt{2}} \text{CNOT} = \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



Rewriting System

We define \rightsquigarrow as *exactly* one **diffusion rule** followed by all possible **collision** rules until none applies.

Want to avoid:

- Having multiple tokens on the same edge that don't collide:
- Non-termination.



Rewriting System

We define \rightsquigarrow as *exactly* one **diffusion rule** followed by all possible **collision** rules until none applies.

Want to avoid:

- Having multiple tokens on the same edge that don't collide:
- Non-termination.



Two invariants:

- **Well-Formedness:** Avoid two tokens going in the same direction on a path.
- **Cycle-Balancedness:** Avoid tokens alone in cycles.

Polarity in a Path

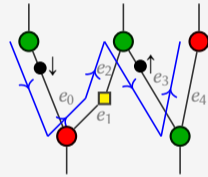
$p = (e_0, e_1, e_2, e_3, e_4)$ is an oriented path.

- If a token follows the path +1
- If it goes against it -1
- If it is not on the path 0

Example:

- Here, polarity

$$P(p, (e_0 \downarrow x)(e_3 \uparrow y)) = P(p, (e_0 \downarrow x)) + P(p, (e_3 \uparrow y)) = 0$$



Well-Formed Token State

Given a ZX-Diagram and a Token State, it is **Well-Formed** if for every path p its Polarity $\in \{-1, 0, 1\}$.

Well-Formed Token State

Given a ZX-Diagram and a Token State, it is **Well-Formed** if for every path p its Polarity $\in \{-1, 0, 1\}$.

- Thm 5.3.11: Well-Formedness preserved under \rightsquigarrow .
- Thm 5.3.12: Well-formed states cannot reach “bad configurations”.

Well-Formed Token State

Given a ZX-Diagram and a Token State, it is **Well-Formed** if for every path p its Polarity $\in \{-1, 0, 1\}$.

- Thm 5.3.11: Well-Formedness preserved under \rightsquigarrow .
- Thm 5.3.12: Well-formed states cannot reach “bad configurations”.

Cycle-Balanced Token State

Given a ZX-Diagram and a Token State, it is **Cycle-Balanced** if for every cycle c its Polarity = 0.

Well-Formed Token State

Given a ZX-Diagram and a Token State, it is **Well-Formed** if for every path p its Polarity $\in \{-1, 0, 1\}$.

- Thm 5.3.11: Well-Formedness preserved under \rightsquigarrow .
- Thm 5.3.12: Well-formed states cannot reach “bad configurations”.

Cycle-Balanced Token State

Given a ZX-Diagram and a Token State, it is **Cycle-Balanced** if for every cycle c its Polarity = 0.

- Thm 5.3.16: Termination of well-formed, cycle-balanced token state.
- Prop 5.3.18: Local confluence of well-formed, cycle-balanced token state.

Thm 5.3.25 (Simulation of Standard Interpretation)

Let D a ZX-Diagram such that $\left[\begin{array}{c} a_1 \quad a_n \\ \dots \\ D \\ \dots \\ b_1 \quad b_m \end{array} \right] = \sum_{q=1}^{2^{m+n}} \lambda_q |y_{1,q} \dots y_{m,q}\rangle \langle x_{1,q} \dots x_{n,q}|$

Let $D = \begin{array}{c} \dots \\ \text{---} \\ D' \\ \text{---} \\ \dots \end{array}$, consider $t = \begin{array}{c} \dots \\ \text{---} \\ D' \\ \begin{array}{c} \bullet \uparrow 0 \\ \bullet \downarrow 0 \end{array} \\ \text{---} \\ \dots \end{array} + \begin{array}{c} \dots \\ \text{---} \\ D' \\ \begin{array}{c} \bullet \uparrow 1 \\ \bullet \downarrow 1 \end{array} \\ \text{---} \\ \dots \end{array}$

Thm 5.3.25 (Simulation of Standard Interpretation)

Let D a ZX-Diagram such that $\left[\begin{array}{c} a_1 \quad a_n \\ \dots \\ D \\ \dots \\ b_1 \quad b_m \end{array} \right] = \sum_{q=1}^{2^{m+n}} \lambda_q |y_{1,q} \dots y_{m,q}\rangle \langle x_{1,q} \dots x_{n,q}|$

Let $D = \begin{array}{c} \dots \\ \square \\ \dots \end{array}$, consider $t = \begin{array}{c} \dots \\ \square \\ \bullet \quad 0 \\ \bullet \quad 0 \\ \dots \end{array} + \begin{array}{c} \dots \\ \square \\ \bullet \quad 1 \\ \bullet \quad 1 \\ \dots \end{array}$

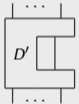
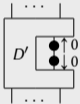
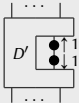
Then

$$t \rightsquigarrow^* \sum_{q=1}^{2^{m+n}} \lambda_q \begin{array}{c} x_{1,q} \uparrow \quad \dots \quad \uparrow x_{n,q} \\ \square \\ y_{1,q} \downarrow \quad \dots \quad \downarrow y_{m,q} \end{array}$$

Thm 5.3.25 (Simulation of Standard Interpretation)

Let D a ZX-Diagram such that

$$\left[\begin{array}{c} a_1 \quad a_n \\ \dots \\ D \\ \dots \\ b_1 \quad b_m \end{array} \right] = \sum_{q=1}^{2^{m+n}} \lambda_q |y_{1,q} \dots y_{m,q}\rangle \underbrace{\langle x_{1,q} \dots x_{n,q}|}$$

Let $D =$ , consider $t =$  $+$ 

Then

$$t \rightsquigarrow^* \sum_{q=1}^{2^{m+n}} \lambda_q \left[\begin{array}{c} x_{1,q} \quad \dots \quad x_{n,q} \\ \uparrow \quad \dots \quad \uparrow \\ D' \\ \downarrow \quad \dots \quad \downarrow \\ y_{1,q} \quad \dots \quad y_{m,q} \end{array} \right]$$

Thm 5.3.25 (Simulation of Standard Interpretation)

Let D a ZX-Diagram such that

$$\left[\begin{array}{c} a_1 \quad a_n \\ \dots \\ D \\ \dots \\ b_1 \quad b_m \end{array} \right] = \sum_{q=1}^{2^{m+n}} \lambda_q \underbrace{|y_{1,q} \dots y_{m,q}\rangle}_{\text{state}} \langle x_{1,q} \dots x_{n,q}|$$

Let $D = \begin{array}{c} \dots \\ \square \\ \dots \end{array}$, consider $t = \begin{array}{c} \dots \\ \square \\ \bullet \uparrow 0 \\ \bullet \downarrow 0 \\ \dots \end{array} + \begin{array}{c} \dots \\ \square \\ \bullet \uparrow 1 \\ \bullet \downarrow 1 \\ \dots \end{array}$

Then

$$t \rightsquigarrow^* \sum_{q=1}^{2^{m+n}} \lambda_q \begin{array}{c} x_{1,q} \uparrow \quad \dots \quad \uparrow x_{n,q} \\ \square \\ \bullet \downarrow y_{1,q} \quad \dots \quad \downarrow y_{m,q} \end{array}$$

Towards a Curry-Howard Correspondence for Quantum Computation

I.f.p operator
(Ch. 4, CSL'23)

Pairing
(Ch. 5, MFCS'21)

$$[A] = \mu X. \mathbb{1} \oplus (A \otimes X)$$

Quantum Control
(Ch. 6, Draft)

Towards a Curry-Howard Correspondence for Quantum Computation

I.f.p operator
(Ch. 4, CSL'23)

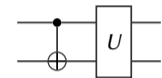
Pairing
(Ch. 5, MFCS'21)

$$[A] = \mu X. \mathbb{1} \oplus (A \otimes X)$$

Quantum Control
(Ch. 6, Draft)

The Many-Worlds Calculus : When \otimes meets \oplus

\otimes -based languages

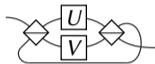


Quantum Circuits

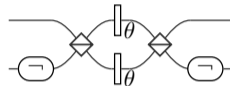


ZX-Calculus

\oplus -based languages



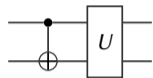
PBS-Calculus



LOv-Calculus

The Many-Worlds Calculus : When \otimes meets \oplus

\otimes -based languages

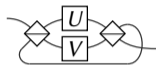


Quantum Circuits

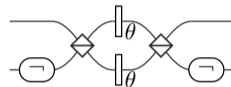


ZX-Calculus

\oplus -based languages

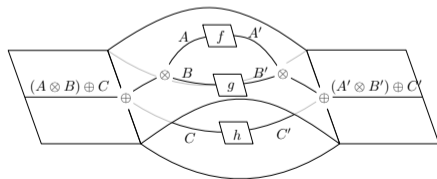


PBS-Calculus



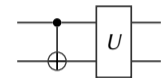
LOv-Calculus

The Many-Worlds Calculus



The Many-Worlds Calculus : When \otimes meets \oplus

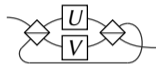
\otimes -based languages



Quantum Circuits

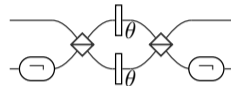


ZX-Calculus



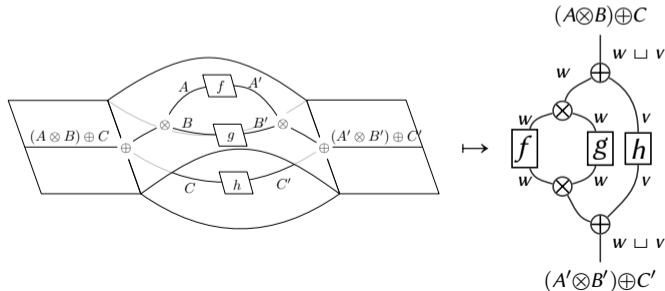
PBS-Calculus

\oplus -based languages



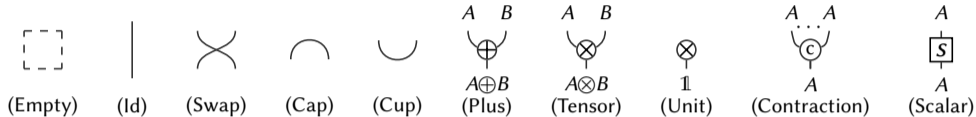
LOv-Calculus

The Many-Worlds Calculus

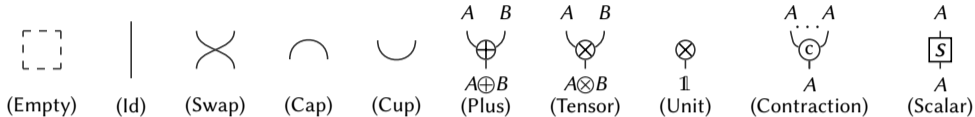


- Label **wires** with **worlds**.
- **Worlds** for naming **slices**.

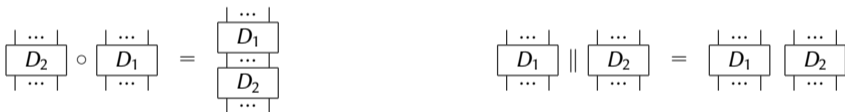
Generators



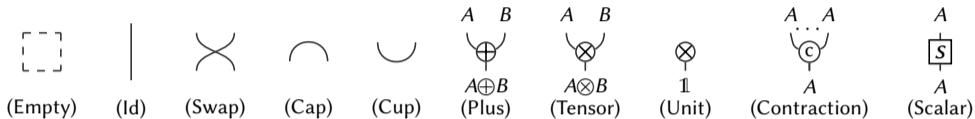
Generators



Compositions



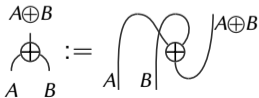
Generators



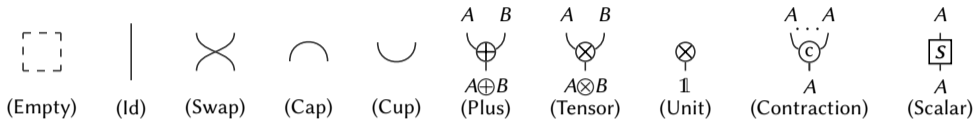
Compositions



Derived Constructors & Tokens



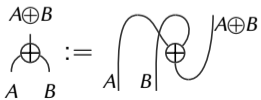
Generators



Compositions

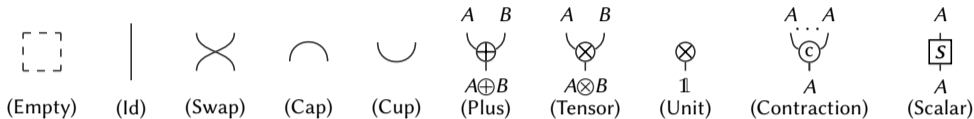


Derived Constructors & Tokens



$v ::= () \mid \langle s_1, s_2 \rangle \mid \text{inj}_\ell s \mid \text{inj}_r s$
 $s ::= v \mid \text{!}$

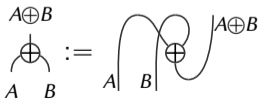
Generators



Compositions



Derived Constructors & Tokens

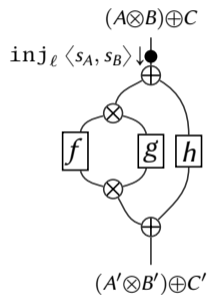


$v ::= () \mid \langle s_1, s_2 \rangle \mid \text{inj}_\ell s \mid \text{inj}_r s$

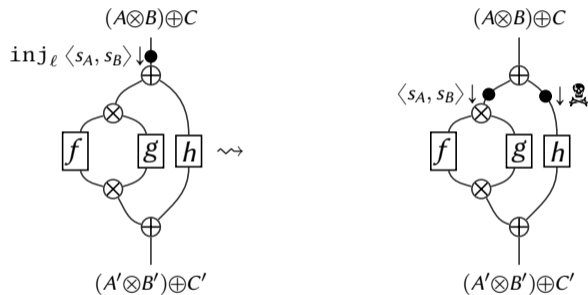
$s ::= v \mid \text{skull}$

Token = $(e, d, b) \in \mathcal{E}(D) \times \{\uparrow, \downarrow\} \times s$

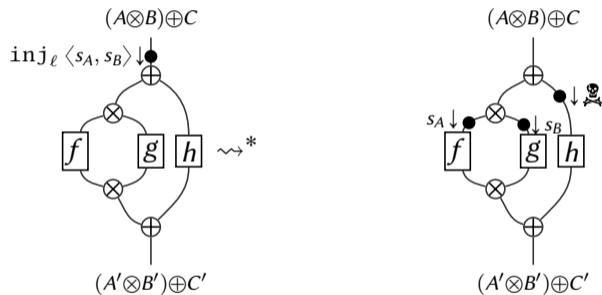
Assume s_A, s_B values of types A and B , then:



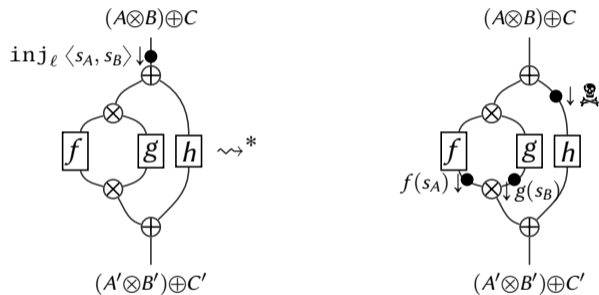
Assume s_A, s_B values of types A and B , then:



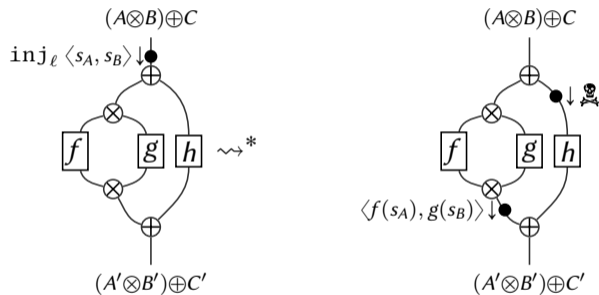
Assume s_A, s_B values of types A and B , then:



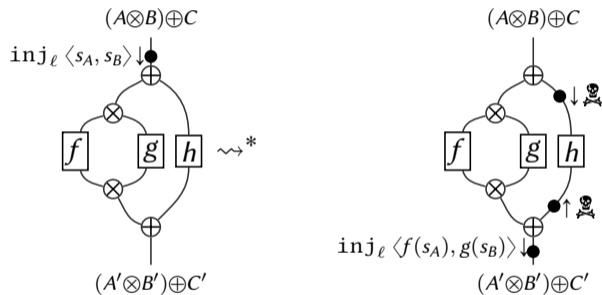
Assume s_A, s_B values of types A and B , then:



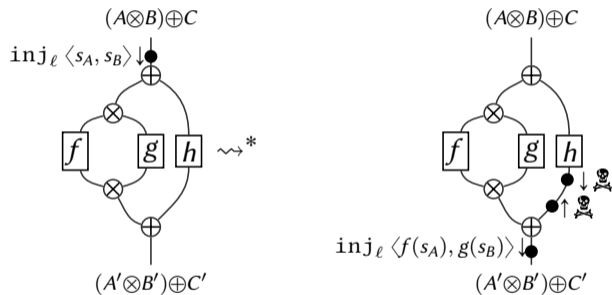
Assume s_A, s_B values of types A and B , then:



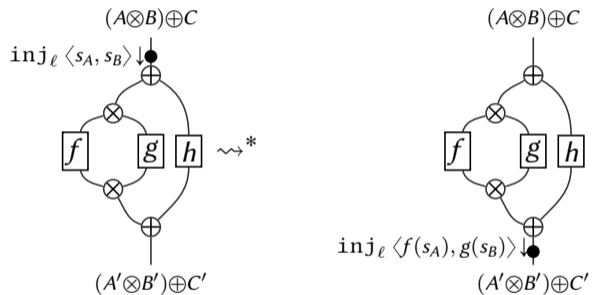
Assume s_A, s_B values of types A and B , then:



Assume s_A, s_B values of types A and B , then:

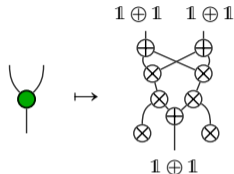


Assume s_A, s_B values of types A and B , then:

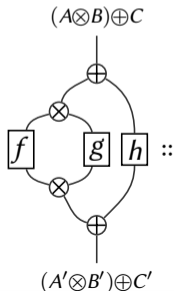


Results

Well-Formedness, Cycle-Balancedness still holds.
 \Rightarrow Confluence, Termination, Avoid bad configurations.

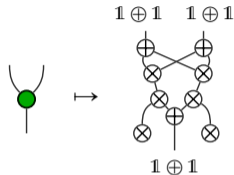


Quantum Control

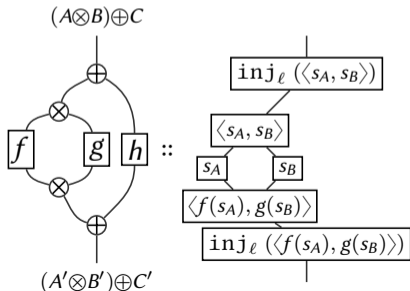


Results

Well-Formedness, Cycle-Balancedness still holds.
 \Rightarrow Confluence, Termination, Avoid bad configurations.

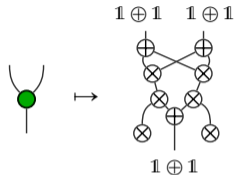


Quantum Control

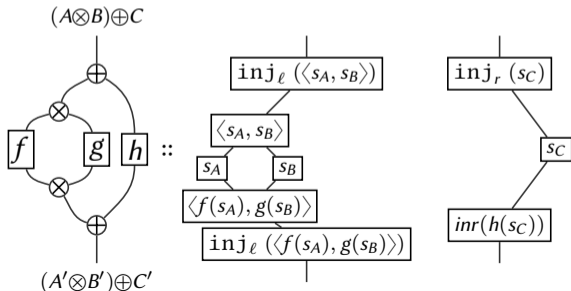


Results

Well-Formedness, Cycle-Balancedness still holds.
 \Rightarrow Confluence, Termination, Avoid bad configurations.

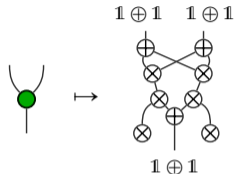


Quantum Control

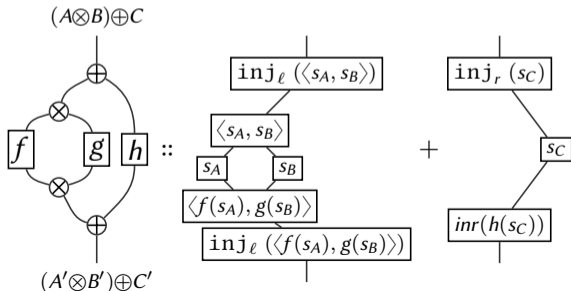


Results

Well-Formedness, Cycle-Balancedness still holds.
 \Rightarrow Confluence, Termination, Avoid bad configurations.

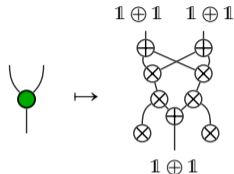


Quantum Control

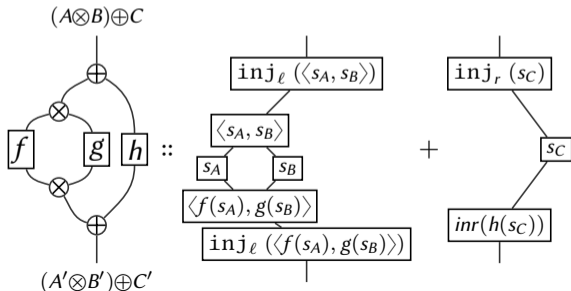


Results

Well-Formedness, Cycle-Balancedness still holds.
 \Rightarrow Confluence, Termination, Avoid bad configurations.



Quantum Control



Missing: Inductive types.

Towards a Curry-Howard Correspondence for Quantum Computation

I.f.p operator
(Ch. 4, CSL'23)

Pairing
(Ch. 5, MFCS'21)

$$[A] = \mu X. \mathbb{1} \oplus (A \otimes X)$$

Quantum Control
(Ch. 6, Draft)

Towards a Curry-Howard Correspondence for Quantum Computation

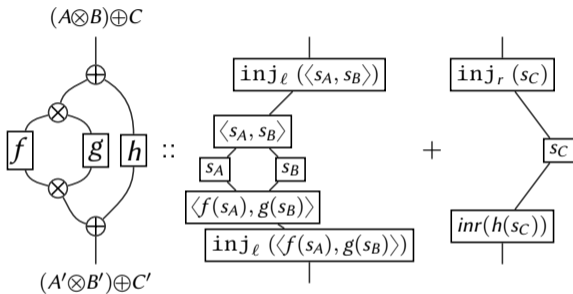
I.f.p operator
(Ch. 4, CSL'23)

Pairing
(Ch. 5, MFCS'21)

$$[A] = \mu X. \mathbb{1} \oplus (A \otimes X)$$

Quantum Control
(Ch. 6, Draft)

A Syntax Term Language for the Many-Worlds Calculus



$$\left\{ \begin{array}{l} \text{inj}_l(\langle\langle x, y \rangle\rangle) \leftrightarrow \text{inj}_l(\langle\langle f x, g y \rangle\rangle) \\ \text{inj}_r(z) \leftrightarrow \text{inj}_r(h z) \end{array} \right\} \quad \text{Function from } (A \otimes B) \oplus C \leftrightarrow (A' \otimes B') \oplus C'$$

(Base types) $A, B ::= \mathbb{1} \mid A \oplus B \mid A \otimes B$

(Isos, first-order) $T ::= A \leftrightarrow B$

(Values) $v ::= x \mid () \mid \langle v_1, v_2 \rangle \mid \text{inj}_\ell v \mid \text{inj}_r v$

(Expressions) $e ::= v \mid \text{let } x = \omega \text{ in } e$

(Isos) $\omega ::= \{v_1 \leftrightarrow e_1 \mid \dots \mid v_n \leftrightarrow e_n\}$

(Base types) $A, B ::= \mathbb{1} \mid A \oplus B \mid A \otimes B \mid \mu X. A \mid X$

(Isos, first-order) $T ::= A \leftrightarrow B$

(Values) $v ::= x \mid () \mid \langle v_1, v_2 \rangle \mid \text{inj}_\ell v \mid \text{inj}_r v \mid \text{fold } e$

(Expressions) $e ::= v \mid \text{let } x = \omega y \text{ in } e$

(Isos) $\omega ::= \{v_1 \leftrightarrow e_1 \mid \dots \mid v_n \leftrightarrow e_n\} \mid \text{fix } f. \omega \mid f$

$$\text{map}(\omega) = \text{fix } f. \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega h) :: (f t) \end{array} \right\} : [A] \leftrightarrow [B]$$

$$[] = \text{fold} (\text{inj}_\ell (())) \quad h :: t = \text{fold} (\text{inj}_r (\langle h, t \rangle))$$

$\mu\text{MALL}^\infty = \text{MALL} + \mu.$

$$\frac{\Delta, A[X \leftarrow \mu X.A] \vdash B}{\Delta, \mu X.A \vdash B} \mu_L \qquad \frac{\Delta \vdash A[X \leftarrow \mu X.A]}{\Delta \vdash \mu X.A} \mu_R$$

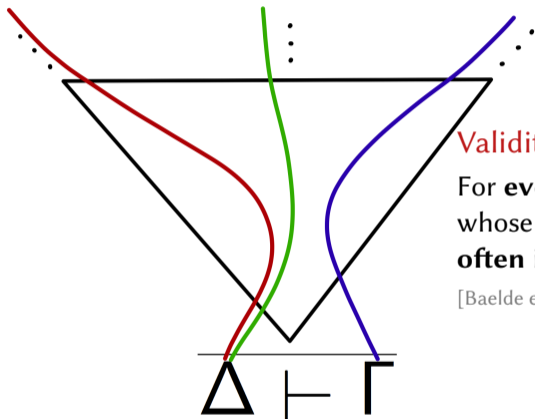
$\mu\text{MALL}^\infty = \text{MALL} + \mu$.

$$\frac{\Delta, A[X \leftarrow \mu X.A] \vdash B}{\Delta, \mu X.A \vdash B} \mu_L \qquad \frac{\Delta \vdash A[X \leftarrow \mu X.A]}{\Delta \vdash \mu X.A} \mu_R$$

Non-wellfounded proofs:

$$\frac{\begin{array}{c} \vdots \\ \mu X.X \vdash F \end{array} \mu \quad \begin{array}{c} \vdots \\ \vdash \mu X.X \end{array} \mu}{\vdash F} \text{cut}$$

There is a need for a **validity criterion** on derivations.



Validity Condition

For **every infinite branch**, there is a **thread** whose **smallest** formula that occurs **infinitely often** is a μ formula on the **left**.

[Baelde et al., 2016, 2022]

Let us take the $\text{map}(\omega)$ functions on lists.

$$\mathbf{fix} \ f. \ \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega \ h) :: (f \ t) \end{array} \right\} : [A] \leftrightarrow [B]$$

Send it to a derivation proof($\text{map}(\omega)$) : $[A] \vdash [B]$.

$$\frac{\frac{\frac{\vdash [B]}{\mathbb{1} \vdash [B]} \mathbb{1}_L \quad \frac{A, [A] \vdash [B]}{A \otimes [A] \vdash [B]} \otimes_L}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \oplus_L}{[A] \vdash [B]} \mu_L$$

$$\mathbf{fix} \ f. \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega \ h) :: (f \ t) \end{array} \right\}$$

$$\begin{array}{c}
 \frac{\frac{\overline{\vdash \mathbb{1}} \quad \mathbb{1}_R}{\vdash \mathbb{1} \oplus (B \otimes [A])} \oplus_R^1}{\vdash [B]} \mu_R \\
 \frac{\frac{\vdash [B]}{\mathbb{1} \vdash [B]} \mathbb{1}_L}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \oplus_L \\
 \frac{A, [A] \vdash [B]}{A \otimes [A] \vdash [B]} \otimes_L \\
 \frac{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]}{[A] \vdash [B]} \mu_L
 \end{array}$$

$$\mathbf{fix} \ f. \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega \ h) :: (f \ t) \end{array} \right\}$$

$$\frac{\frac{\frac{\overline{\vdash \mathbb{1}} \quad \mathbb{1}_R}{\vdash \mathbb{1} \oplus (B \otimes [A])} \oplus_R^1}{\vdash [B]} \mu_R \quad \frac{\frac{A, [A] \vdash B \otimes [B]}{A, [A] \vdash \mathbb{1} \oplus (B \otimes [B])} \oplus_R^2}{A, [A] \vdash [B]} \mu_R}{\frac{\frac{\vdash [B]}{\mathbb{1} \vdash [B]} \mathbb{1}_L}{A \otimes [A] \vdash [B]} \otimes_L}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \oplus_L} \mu_L$$

$$\mathbf{fix} \ f. \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega \ h) :: (f \ t) \end{array} \right\}$$

From Derivations To Proofs - Example

$$\begin{array}{c}
 \frac{\frac{\frac{\overline{\vdash \mathbb{1}} \quad \mathbb{1}_R}{\vdash \mathbb{1} \oplus (B \otimes [A])} \oplus_R^1}{\vdash [B]} \quad \mathbb{1}_L}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \mu_R \\
 \\
 \frac{\frac{\frac{\overline{A \vdash B} \quad \omega}{A, [A] \vdash B \otimes [B]} \otimes_R}{A, [A] \vdash \mathbb{1} \oplus (B \otimes [B])} \oplus_R^2}{\frac{A, [A] \vdash [B]}{A \otimes [A] \vdash [B]} \otimes_L} \mu_R \\
 \\
 \frac{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]}{[A] \vdash [B]} \oplus_L \mu_L
 \end{array}$$

$$\mathbf{fix} \ f. \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega \ h) :: (f \ t) \end{array} \right\}$$

$$\frac{\frac{\frac{\overline{\vdash \mathbb{1}} \quad \mathbb{1}_R}{\vdash \mathbb{1} \oplus (B \otimes [A])} \oplus_R^1}{\vdash [B]} \quad \mathbb{1}_L}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \mu_R}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \mu_L$$

$$\frac{\frac{\frac{\frac{\omega}{A \vdash B} \quad \frac{\vdots}{[A] \vdash [B]}}{A, [A] \vdash B \otimes [B]} \otimes_R}{A, [A] \vdash \mathbb{1} \oplus (B \otimes [B])} \oplus_R^2}{A, [A] \vdash [B]} \mu_R}{A \otimes [A] \vdash [B]} \otimes_L}{A \otimes [A] \vdash [B]} \oplus_L$$

$$\mathbf{fix} \ f. \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega \ h) :: (f \ t) \end{array} \right\}$$

$$\begin{array}{c}
 \frac{\frac{\frac{\overline{\vdash \mathbb{1}} \quad \mathbb{1}_R}{\vdash \mathbb{1} \oplus (B \otimes [A])} \oplus_R^1}{\vdash [B]} \quad \mu_R}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \oplus_L^1 \\
 \frac{\frac{\frac{\overline{\vdash \mathbb{1}} \quad \mathbb{1}_R}{\vdash \mathbb{1} \oplus (B \otimes [A])} \oplus_R^1}{\vdash [B]} \quad \mu_R}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \oplus_L^1 \\
 \frac{\frac{\frac{\frac{\overline{\omega} \quad \vdots}{A \vdash B} \quad \frac{\overline{[A] \vdash [B]}}{[A] \vdash [B]} \oplus_R}{A, [A] \vdash B \otimes [B]} \otimes_R}{A, [A] \vdash \mathbb{1} \oplus (B \otimes [B])} \oplus_R^2}{A, [A] \vdash [B]} \quad \mu_R \\
 \frac{A, [A] \vdash [B]}{A \otimes [A] \vdash [B]} \otimes_L \\
 \frac{A \otimes [A] \vdash [B]}{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]} \oplus_L \\
 \frac{\mathbb{1} \oplus (A \otimes [A]) \vdash [B]}{[A] \vdash [B]} \mu_L
 \end{array}$$

$$\mathbf{fix} \ f. \left\{ \begin{array}{l} [] \leftrightarrow [] \\ h :: t \leftrightarrow (\omega h) :: (f t) \end{array} \right\}$$

Language

Iso have an well-typed inverse. Lemma 4.2.11: $\omega : A \leftrightarrow B$ then $\omega^{-1} : B \leftrightarrow A$.

Iso are isomorphisms. Thm 4.2.13: $\omega \circ \omega^{-1} = \text{Id}$.

Subject Reduction & Progress. Lemma 4.2.18 & 4.2.19.

Curry-Howard

Proof Validity. Thm 4.4.20: If $\omega : A \leftrightarrow B$ then $\text{proof}(\omega) : A \vdash B$ is a proof.

Cut-Elimination Simulation. Thm 4.4.29: If $t \rightarrow t'$ then $\text{proof}(t) \rightarrow \text{proof}(t')$.

ZX Token Machine: MFCS'21

- **With:** Benoît Valiron, Renaud Vilmart.
- **What:** Study of Token-Based semantics for the ZX-Calculus.
- **Why:** First approach for control quantum, led to the Many-Worlds Calculus.

ZX Token Machine: MFCS'21

- **With:** Benoît Valiron, Renaud Vilmart.
- **What:** Study of Token-Based semantics for the ZX-Calculus.
- **Why:** First approach for control quantum, led to the Many-Worlds Calculus.

Many-Worlds Calculus: Draft

- **With:** Marc de Visme, Benoît Valiron, Renaud Vilmart.
- **What:** Graphical language with \otimes and \oplus .
- **Why:** Explicit quantum control, quantum types.

ZX Token Machine: MFCS'21

- **With:** Benoît Valiron, Renaud Vilmart.
- **What:** Study of Token-Based semantics for the ZX-Calculus.
- **Why:** First approach for control quantum, led to the Many-Worlds Calculus.

Many-Worlds Calculus: Draft

- **With:** Marc de Visme, Benoît Valiron, Renaud Vilmart.
- **What:** Graphical language with \otimes and \oplus .
- **Why:** Explicit quantum control, quantum types.

Isomorphisms & μ MALL: CSL'23

- **With:** Alexis Saurin, Benoît Valiron.
- **What:** Linear, reversible programming language with inductive types.
- **Why:** First step towards quantum inductive types.

Future Work:

- Quantum Programming Language with Inductive Types.
- Many-Worlds with Inductive Types.
- Relation between the two ?

Future Work:

- Quantum Programming Language with Inductive Types.
- Many-Worlds with Inductive Types.
- Relation between the two ?

First step:

- Remove inductive types.
- Add superposition of expressions.
- Translate iso as Many-Worlds diagrams.
- Thm 6.9.14 : Soudness.