Towards Term-based Verification of Diagrammatic Equivalence

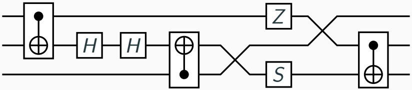
IJCAR 2026

Julie Cailler[1], Noé Delorme[1], Simon Perdrix[1], and Sophie Tourret[1,2]

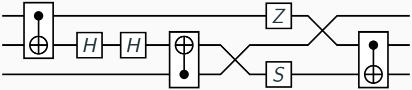
[1]Université de Lorraine, CNRS, INRIA, LORIA, Nancy, France

[2]Max Plank Institute for Informatics, Saarbrücken, Germany

Quantum circuits

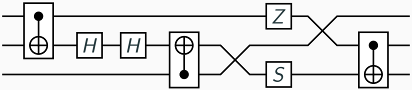


Quantum circuits



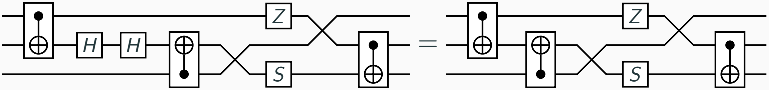
Similar to boolean circuits.

Quantum circuits



Similar to boolean circuits.
Used to describe quantum algorithms.

Quantum circuits

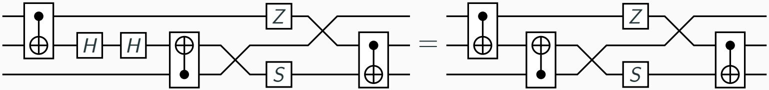


Similar to boolean circuits.

Used to describe quantum algorithms.

There are multiple quantum circuits for a single algorithm.

Quantum circuits



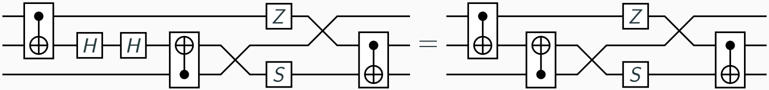
Similar to boolean circuits.

Used to describe quantum algorithms.

There are multiple quantum circuits for a single algorithm.

We can optimize quantum circuits by rewriting them into equivalent ones.

Quantum circuits



Similar to boolean circuits.

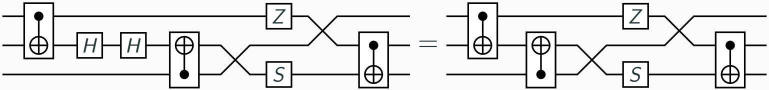
Used to describe quantum algorithms.

There are multiple quantum circuits for a single algorithm.

We can optimize quantum circuits by rewriting them into equivalent ones.

Crucial task in quantum compilation pipelines.

Quantum circuits



Similar to boolean circuits.

Used to describe quantum algorithms.

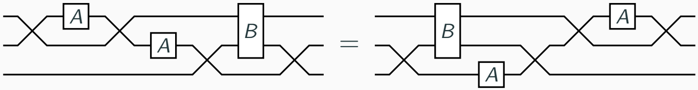
There are multiple quantum circuits for a single algorithm.

We can optimize quantum circuits by rewriting them into equivalent ones.

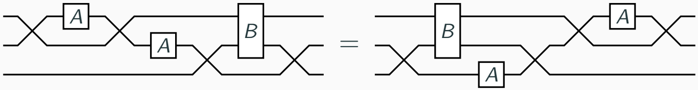
Crucial task in quantum compilation pipelines.

Our long-term goal: Certify quantum circuit equivalences.

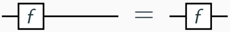
String diagrams

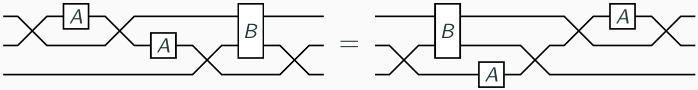


String diagrams

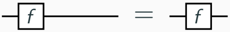


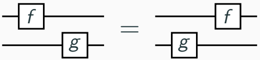
The string diagram formalism comes with a bunch of coherence laws governing the intuitive behavior of wires and boxes.



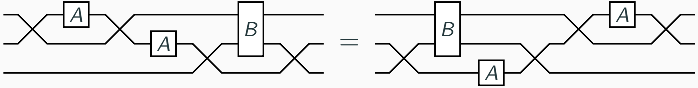


The string diagram formalism comes with a bunch of coherence laws governing the intuitive behavior of wires and boxes.

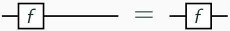


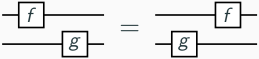


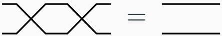
String diagrams

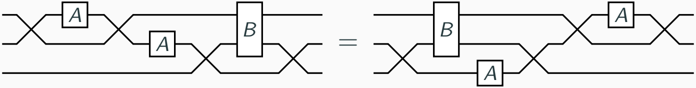


The string diagram formalism comes with a bunch of coherence laws governing the intuitive behavior of wires and boxes.

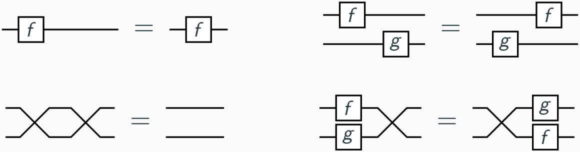






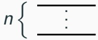


The string diagram formalism comes with a bunch of coherence laws governing the intuitive behavior of wires and boxes.

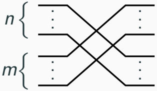


...

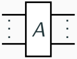
From diagrams to terms



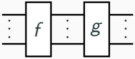
$\longrightarrow \mathrm{id}_n$



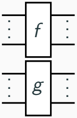
$\longrightarrow \sigma_{n,m}$



$\longrightarrow A$

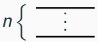


$\longrightarrow f \fatsemi g$

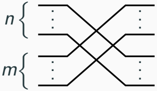


$\longrightarrow f \otimes g$

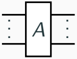
From diagrams to terms



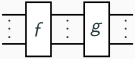
$\longrightarrow \mathrm{id}_n$



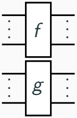
$\longrightarrow \sigma_{n,m}$



$\longrightarrow A$

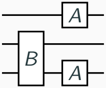


$\longrightarrow f \mathbin{;} g$



$\longrightarrow f \otimes g$

But then, there are multiple terms representing the same diagram.



$$\longrightarrow \begin{cases} A \otimes (B \mathbin{;} (\mathrm{id}_1 \otimes A)) \\ (\mathrm{id}_1 \otimes B) \mathbin{;} (A \otimes (\mathrm{id}_1 \otimes A)) \\ (A \otimes B) \mathbin{;} (\mathrm{id}_2 \otimes A) \\ \ldots \end{cases}$$

Coherence laws as terms

The coherence laws yield rewrite equations between terms.

$$f \fatsemi \mathrm{id}_n = f$$

$$(f \otimes \mathrm{id}_n) \fatsemi (\mathrm{id}_m \otimes g) = (\mathrm{id}_m \otimes g) \fatsemi (f \otimes \mathrm{id}_n)$$

$$\sigma_{n,m} \fatsemi \sigma_{m,n} = \mathrm{id}_{n+m}$$

$$(f \otimes g) \fatsemi \sigma_{m,q} = \sigma_{n,p}(g \otimes f)$$

Diagrammatic equivalence

Given two diagrammatically equivalent terms t_1 and t_2, compute a rewrite sequence from t_1 to t_2.

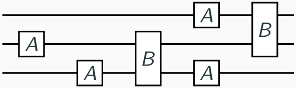
Contributions

General problem seems difficult. We solve two sub-problems.

Contributions

General problem seems difficult. We solve two sub-problems.

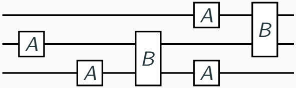
(1) Diagrams without swaps.



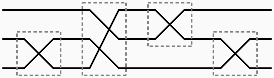
Contributions

General problem seems difficult. We solve two sub-problems.

(1) Diagrams without swaps.



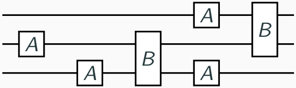
(2) Diagrams with only swaps.



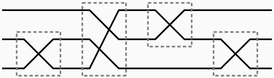
Contributions

General problem seems difficult. We solve two sub-problems.

(1) Diagrams without swaps.



(2) Diagrams with only swaps.

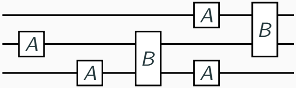


We tried some automated tools to encode the problem (egglog, Vampire, E, etc.). But it was not successful, especially because of the inherent presence of arithmetic in the problem.

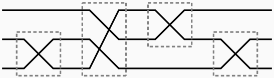
Contributions

General problem seems difficult. We solve two sub-problems.

(1) Diagrams without swaps.



(2) Diagrams with only swaps.



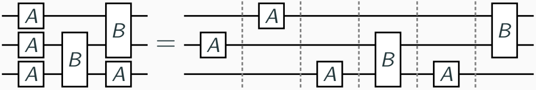
We tried some automated tools to encode the problem (egglog, Vampire, E, etc.). But it was not successful, especially because of the inherent presence of arithmetic in the problem.

Instead, we design terminating and confluent term rewriting systems.

Diagrams without swaps

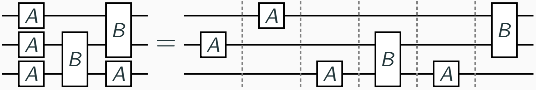
Sequence of layers

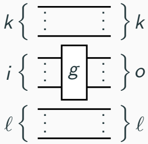
Idea: Transform diagrams as sequences of layers.



Sequence of layers

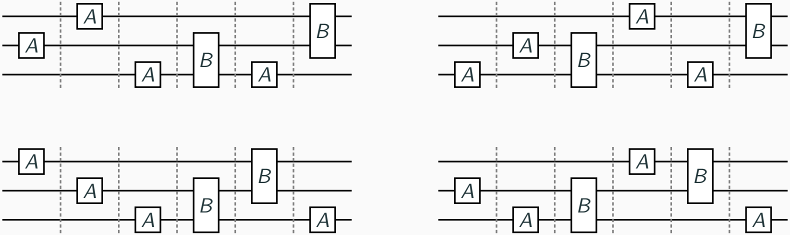
Idea: Transform diagrams as sequences of layers.



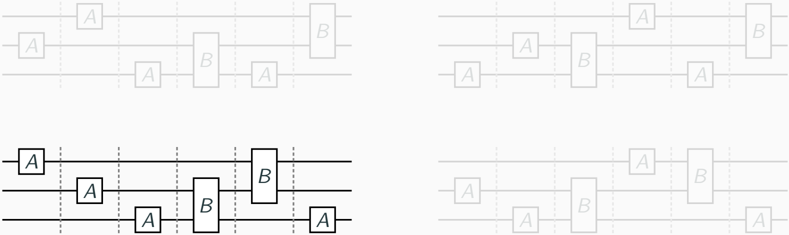
$L^k_\ell(g) := \mathrm{id}_k \otimes (g \otimes \mathrm{id}_\ell) =$ 

Ordering the layers

Issue: How to order the layers?

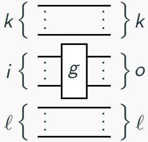


Issue: How to order the layers?



Normal form

$$L_\ell^k(g) \coloneqq \mathrm{id}_k \otimes (g \otimes \mathrm{id}_\ell) =$$



Definition

A normal form is a sequence of layers

$$L_{\ell_1}^{k_1}(g_1) \fatsemi \left(L_{\ell_2}^{k_2}(g_2) \fatsemi \left(L_{\ell_3}^{k_3}(g_3) \fatsemi \left(\dots \right) \right) \right)$$

such that

$$k_1 < k_2 + i_2 \quad , \quad k_2 < k_3 + i_3 \quad , \quad k_3 < k_4 + i_4 \quad , \quad \dots$$

Rewriting system

$$(t \mathbin{⨟} u) \mathbin{⨟} v \longrightarrow t \mathbin{⨟} (u \mathbin{⨟} v) \quad \text{(R1)} \qquad (t \otimes u) \otimes v \longrightarrow t \otimes (u \otimes v) \quad \text{(R2)}$$

$$\mathrm{id}_n \mathbin{⨟} t \longrightarrow t \quad \text{(R3)} \qquad t \mathbin{⨟} \mathrm{id}_n \longrightarrow t \quad \text{(R4)}$$

$$\mathrm{id}_n \otimes \mathrm{id}_m \longrightarrow \mathrm{id}_{n+m} \quad \text{(R5)} \qquad \mathrm{id}_n \otimes (\mathrm{id}_m \otimes t) \longrightarrow \mathrm{id}_{n+m} \otimes t \quad \text{(R6)}$$

$$\begin{array}{c} u \otimes v \longrightarrow (u \otimes \mathrm{id}_{\mathrm{dom}(v)}) \mathbin{⨟} (\mathrm{id}_{\mathrm{cod}(u)} \otimes v) \\ \text{if } \; u \neq \mathrm{id}_n \wedge v \neq \mathrm{id}_m \end{array} \quad \text{(R7)}$$

$$\mathrm{id}_n \otimes (u \mathbin{⨟} v) \longrightarrow (\mathrm{id}_n \otimes u) \mathbin{⨟} (\mathrm{id}_n \otimes v) \quad \text{(R8)}$$

$$(u \mathbin{⨟} v) \otimes \mathrm{id}_n \longrightarrow (u \otimes \mathrm{id}_n) \mathbin{⨟} (v \otimes \mathrm{id}_n) \quad \text{(R9)}$$

$$\begin{array}{c} L^{k_1}_{\ell_1}(g_1) \mathbin{⨟} L^{k_2}_{\ell_2}(g_2) \longrightarrow L^{k_2}_{\ell_2+i_1-o_1}(g_2) \mathbin{⨟} L^{k_1-i_2+o_2}_{\ell_1}(g_1) \\ \text{if } \; k_1 \geq k_2 + i_2 \end{array} \quad \text{(R10)}^*$$

$*$ There are actually two instances of this rule, allowing to apply it in a bigger sequence of layers.

Result

Theorem

The rewriting system is terminating and confluent.

Theorem

The rewriting system is terminating and confluent.

$$
\begin{tikzcd}
(A \otimes B) \fatsemi (\mathrm{id}_2 \otimes A) \arrow[dr] & \simeq & A \otimes (B \fatsemi (\mathrm{id}_1 \otimes A)) \arrow[dl] \\
& (\mathrm{id}_0 \fatsemi (A \otimes \mathrm{id}_2)) \fatsemi ((\mathrm{id}_1 \fatsemi (B \otimes \mathrm{id}_0)) \fatsemi (\mathrm{id}_2 \fatsemi (A \otimes \mathrm{id}_0))) &
\end{tikzcd}
$$

$$
= L_2^0(A) \fatsemi (L_0^1(B) \fatsemi L_0^2(A))
$$

Result

Theorem

The rewriting system is terminating and confluent.

$$
\begin{tikzcd}[column sep=small]
(A \otimes B) \fatsemi (\mathrm{id}_2 \otimes A) \arrow[dr] & \simeq & A \otimes (B \fatsemi (\mathrm{id}_1 \otimes A)) \arrow[dl] \\
& (\mathrm{id}_0 \fatsemi (A \otimes \mathrm{id}_2)) \fatsemi ((\mathrm{id}_1 \fatsemi (B \otimes \mathrm{id}_0)) \fatsemi (\mathrm{id}_2 \fatsemi (A \otimes \mathrm{id}_0))) &
\end{tikzcd}
$$

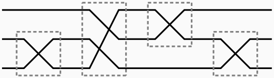
$$= L^0_2(A) \fatsemi (L^1_0(B) \fatsemi L^2_0(A))$$

The key steps of the proofs have been verified using the proof assistant Isabelle/HOL.

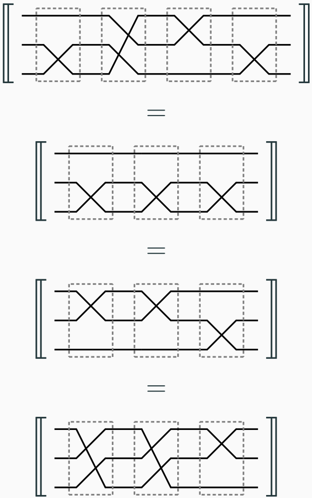
Diagrams with only swaps

Permutations

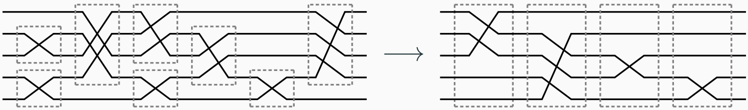
$$[\![-]\!] : \text{terms} \to \text{permutations}$$

$[\![$  $]\!] = (1, 3, 2)$

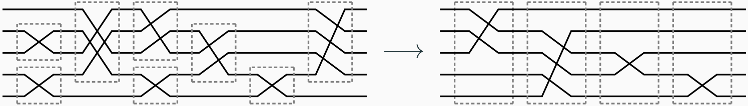
The normal form is no longer unique



Canonical form



Canonical form



Definition

A canonical form is a normal form

$$L^{k_1}_{\ell_1}(\sigma_{n_1,m_1}) \fatsemi \left(L^{k_2}_{\ell_2}(\sigma_{n_2,m_2}) \fatsemi \left(L^{k_3}_{\ell_3}(\sigma_{n_3,m_3}) \fatsemi \left(\dots \right)\right)\right)$$

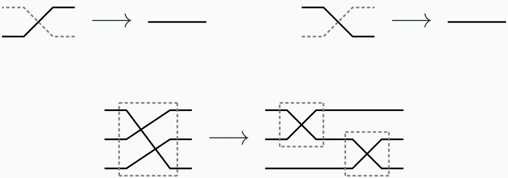
such that $m_1 = m_2 = m_3 = \cdots = 1$ and

$$k_1 < k_2 \quad , \quad k_2 < k_3 \quad , \quad k_3 < k_4 \quad , \quad \dots$$

New rewrite rules

$$L^{k_1}_{\ell_1}(\sigma_{n_1,m_1}) \fatsemi \left(L^{k_2}_{\ell_2}(\sigma_{n_2,m_2}) \fatsemi \left(L^{k_3}_{\ell_3}(\sigma_{n_3,m_3}) \fatsemi \left(\dots \right)\right)\right)$$

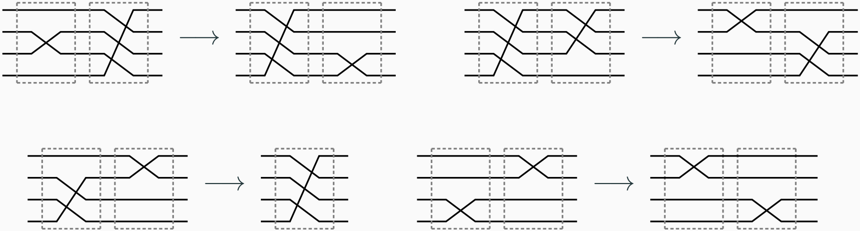
The conditions $m_i = 1$ are satisfied with the following new rewrite rules.



New rewrite rules

$$L^{k_1}_{\ell_1}(\sigma_{n_1,m_1}) \fatsemi \left(L^{k_2}_{\ell_2}(\sigma_{n_2,m_2}) \fatsemi \left(L^{k_3}_{\ell_3}(\sigma_{n_3,m_3}) \fatsemi \left(\ldots \right)\right)\right)$$

The other conditions $k_i < k_{i+1}$ are satisfied with the following new rewrite rules.



Rewriting system

$$\sigma_{0,n} \longrightarrow \mathrm{id}_n \quad \text{(R11)} \qquad \sigma_{n,0} \longrightarrow \mathrm{id}_n \quad \text{(R12)}$$

$$\sigma_{n,m+1} \longrightarrow (\sigma_{n,m} \otimes \mathrm{id}_1) \mathbin{;} (\mathrm{id}_m \otimes \sigma_{n,1}) \quad \text{(R13)}$$

$$\begin{gathered} L^{k_1}_{\ell_1}(\tau_{d_1}) \mathbin{;} L^{k_2}_{\ell_2}(\tau_{d_2}) \longrightarrow L^{k_2}_{\ell_2}(\tau_{d_2}) \mathbin{;} L^{k_1+1}_{\ell_1-1}(\tau_{d_1}) \\ \text{if } \ d_1 \geq 1, d_2 \geq 1 \ \text{ and } \ k_2 \leq k_1 < k_2 + d_2 - d_1 \end{gathered} \qquad \text{(R14)}$$

$$\begin{gathered} L^{k_1}_{\ell_1}(\tau_{d_1}) \mathbin{;} L^{k_2}_{\ell_2}(\tau_{d_2}) \longrightarrow L^{k_2}_{\ell_2+1}(\tau_{d_2-1}) \mathbin{;} L^{k_1+1}_{\ell_1}(\tau_{d_1-1}) \\ \text{if } \ d_1 \geq 1, d_2 \geq 1 \ \text{ and } \ k_2 \leq k_1 \wedge k_2 + d_2 - d_1 \leq k_1 < k_2 + d_2 \end{gathered} \qquad \text{(R15)}$$

$$\begin{gathered} L^{k_1}_{\ell_1}(\tau_{d_1}) \mathbin{;} L^{k_2}_{\ell_2}(\tau_{d_2}) \longrightarrow L^{k_2}_{\ell_1}(\tau_{d_1+d_2}) \\ \text{if } \ d_1 \geq 1, d_2 \geq 1 \ \text{ and } \ k_1 = k_2 + d_2 \end{gathered} \qquad \text{(R16)}$$

$$\begin{gathered} L^{k_1}_{\ell_1}(\tau_{d_1}) \mathbin{;} L^{k_2}_{\ell_2}(\tau_{d_2}) \longrightarrow L^{k_2}_{\ell_2}(\tau_{d_2}) \mathbin{;} L^{k_1}_{\ell_1}(\tau_{d_1}) \\ \text{if } \ d_1 \geq 1, d_2 \geq 1 \ \text{ and } \ k_1 > k_2 + d_2 \end{gathered} \qquad \text{(R17)}$$

Result

Theorem

The rewriting system is terminating and confluent.

Result

Theorem

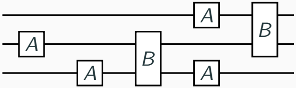
The rewriting system is terminating and confluent.

The key steps of the proofs have been verified using the proof assistant Isabelle/HOL.

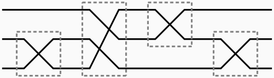
Future work

We solved two sub-problems.

(1) Diagrams without swaps.



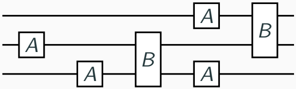
(2) Diagrams with only swaps.



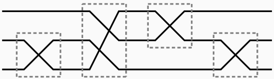
Future work

We solved two sub-problems.

(1) Diagrams without swaps.

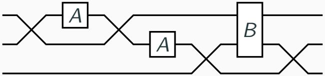


(2) Diagrams with only swaps.



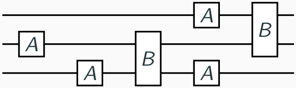
The next step is to combine the two solutions to solve the general problem.

(3) Diagrams with swaps and boxes.

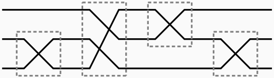


We solved two sub-problems.

(1) Diagrams without swaps.

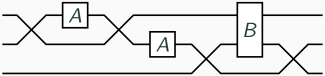


(2) Diagrams with only swaps.



The next step is to combine the two solutions to solve the general problem.

(3) Diagrams with swaps and boxes.



Thanks!