quantum information flow


FORTHCOMING: textbook with Aleks Kissinger on a purely diagrammatic presentation of basic quantum information
Overview / general idea:
Overview / general idea:

- Quantum informatic systems and operations and computations thereon admit diagrammatic calculus.
Overview / general idea:

- Quantum informatic systems and operations and computations thereon admit *diagrammatic calculus*.

- Pictures expose ‘*information flows*’.
Overview / general idea:

- Quantum informatic systems and operations and computations thereon admit diagrammatic calculus.

- Pictures expose ‘information flows’.

- Important fragments are ‘complete’:
Overview / general idea:

- Quantum informatic systems and operations and computations thereon admit **diagrammatic calculus**.

- Pictures expose ‘**information flows**’.

- Important fragments are ‘**complete**’:
  - Bipartite entanglement generated (Selinger’08)
Overview / general idea:

- Quantum informatic systems and operations and computations thereon admit **diagrammatic calculus**.

- Pictures expose ‘**information flows**’.

- Important fragments are ‘**complete**’:
  - Bipartite entanglement generated (Selinger’08)
  - The qubit stabiliser fragment (Backens’12)
Overview / general idea:

- Quantum informatic systems and operations and computations thereon admit diagrammatic calculus.

- Pictures expose ‘information flows’.

- Important fragments are ‘complete’:
  - Bipartite entanglement generated (Selinger’08)
  - The qubit stabiliser fragment (Backens’12)

- The mathematical backbone is category theory.
Category Theory, . . .
Category Theory, . . .

. . . isn’t it just tedious abstract nonsense?
Category Theory, . . .

. . . isn’t it just tedious abstract nonsense? No!
Category Theory, . . .

. . . isn’t it just tedious abstract nonsense? No!

Symmetric Monoidal Categories are everywhere!
1. Let $A$ be a raw potato.
1. Let \( A \) be a **raw potato**.

\( A \) admits many **states** e.g. **dirty, clean, skinned, ...**
1. Let $A$ be a raw potato. 
$A$ admits many states e.g. dirty, clean, skinned, ...

2. We want to process $A$ into cooked potato $B$. 
$B$ admits many states e.g. boiled, fried, deep fried, baked with skin, baked without skin, ...
1. Let $A$ be a raw potato. $A$ admits many states e.g. dirty, clean, skinned, ...

2. We want to process $A$ into cooked potato $B$. $B$ admits many states e.g. boiled, fried, deep fried, baked with skin, baked without skin, ... Let

\[ A \xrightarrow{f} B \quad A \xrightarrow{f'} B \quad A \xrightarrow{f''} B \]

be boiling, frying, baking.
1. Let $A$ be a raw potato. $A$ admits many *states* e.g. dirty, clean, skinned, ...

2. We want to *process* $A$ into cooked potato $B$. $B$ admits many *states* e.g. boiled, fried, deep fried, baked with skin, baked without skin, ... Let

$$A \xrightarrow{f} B \quad A \xrightarrow{f'} B \quad A \xrightarrow{f''} B$$

be boiling, frying, baking. *States are processes*

$I := \text{unspecified} \xrightarrow{\psi} A$. 

3. Let

\[ A \xrightarrow{g \circ f} C \]

be the \textit{composite process} of first boiling \( A \xrightarrow{f} B \) and then salting \( B \xrightarrow{g} C \).
3. Let

\[ A \xrightarrow{g \circ f} C \]

be the \textit{composite process} of first boiling \[ A \xrightarrow{f} B \] and then salting \[ B \xrightarrow{g} C \]. Let

\[ X \xrightarrow{1_X} X \]

be doing nothing. We have \[ 1_Y \circ \xi = \xi \circ 1_X = \xi \].
4. Let $A \otimes D$ be potato $A$ and carrot $D$
4. Let $A \otimes D$ be potato $A$ and carrot $D$ and let

$$A \otimes D \xrightarrow{f \otimes h} B \otimes E$$

be boiling potato while frying carrot.
4. Let $A \otimes D$ be potato $A$ and carrot $D$ and let

$$A \otimes D \xrightarrow{f \otimes h} B \otimes E$$

be boiling potato while frying carrot. Let

$$C \otimes F \xrightarrow{x} M$$

be mashing spice-cook-potato and spice-cook-carrot.
5. Total process:

\[ A \otimes D \xrightarrow{f \otimes h} B \otimes E \xrightarrow{g \otimes k} C \otimes F \xrightarrow{x} M = A \otimes D \xrightarrow{x \circ (g \otimes k) \circ (f \otimes h)} M. \]
5. Total process:

$$A \otimes D \xrightarrow{f \otimes h} B \otimes E \xrightarrow{g \otimes k} C \otimes F \xrightarrow{x} M = A \otimes D \xrightarrow{x \circ (g \otimes k) \circ (f \otimes h)} M.$$ 

6. Recipe = composition structure on processes.
5. Total process:

\[ A \otimes D \xrightarrow{f \otimes h} B \otimes E \xrightarrow{g \otimes k} C \otimes F \xrightarrow{x} M = A \otimes D \xrightarrow{x \circ (g \otimes k) \circ (f \otimes h)} M. \]

6. Recipe = composition structure on processes.

7. Laws governing recipes:

\[ (1_B \otimes g) \circ (f \otimes 1_C) = (f \otimes 1_D) \circ (1_A \otimes g) \]
5. Total process:

\[ A \otimes D \xrightarrow{f \otimes h} B \otimes E \xrightarrow{g \otimes k} C \otimes F \xrightarrow{x} M = A \otimes D \xrightarrow{x \circ (g \otimes k) \circ (f \otimes h)} M. \]

6. Recipe = composition structure on processes.

7. Laws governing recipes:

\[ (1_B \otimes g) \circ (f \otimes 1_C) = (f \otimes 1_D) \circ (1_A \otimes g) \]

i.e.

boil potato then fry carrot = fry carrot then boil potato
5. Total process:

\[ A \otimes D \xrightarrow{f \otimes h} B \otimes E \xrightarrow{g \otimes k} C \otimes F \xrightarrow{x} M = A \otimes D \xrightarrow{x \circ (g \otimes k) \circ (f \otimes h)} M. \]

6. Recipe = composition structure on processes.

7. Laws governing recipes:

\[ (1_B \otimes g) \circ (f \otimes 1_C) = (f \otimes 1_D) \circ (1_A \otimes g) \]

i.e.

boil potato then fry carrot = fry carrot then boil potato

\[ \Rightarrow \text{Symmetric Monoidal Category} \]
Why does a tiger have stripes and a lion doesn’t?
Why does a tiger have stripes and a lion doesn’t?

prey \otimes predator \otimes environment

\begin{align*}
\text{hunt} \\
\text{dead prey} \otimes \text{eating predator}
\end{align*}


wire and box language

Box :=

\[ f \]

input wire(s)  \[ \rightarrow \]  output wire(s)
— wire and box language —

\[
\text{Box} := \begin{array}{c}
\begin{array}{c}
\text{output wire(s)}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\text{input wire(s)}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
f
\end{array}
\end{array}
\end{array}
\]

**Interpretation:** wire := system ; box := process
wire and box language

Box :=

\[ \begin{array}{c}
\text{output wire(s)} \\
\text{input wire(s)}
\end{array} \]

Interpretation: wire := system ; box := process

one system: \hspace{1cm} n subsystems: \hspace{1cm} no system:

\[ \begin{array}{c}
1 \\
\{ \text{n} \} \\
0
\end{array} \]
wire and box games
wire and box games

sequential or causal or connected composition:

\[ g \circ f \equiv \]

\[ g \]

\[ f \]
— wire and box games —

sequential or causal or connected composition:

\[ g \circ f \equiv \begin{array}{c} g \end{array} \begin{array}{c} f \end{array} \]

parallel or acausal or disconnected composition:

\[ f \otimes g \equiv \begin{array}{c} f \end{array} \begin{array}{c} g \end{array} \]
merely a new notation?
merely a new notation?

\[(g \circ f) \otimes (k \circ h) = (g \otimes k) \circ (f \otimes h)\]
merely a new notation?

\[(g \circ f) \otimes (k \circ h) = (g \otimes k) \circ (f \otimes h)\]

peel potato and then fry it, peel potato while clean carrot, and then, while, clean carrot and then boil it fry potato while boil carrot
MINIMAL QUANTUM PROCESS LANGUAGE


[von Neumann 1932] Formalized quantum mechanics in “Mathematische Grundlagen der Quantenmechanik”
[von Neumann 1932] Formalized quantum mechanics in “Mathematische Grundlagen der Quantenmechanik”

[von Neumann to Birkhoff 1935] “I would like to make a confession which may seem immoral: I do not believe absolutely in Hilbert space no more.” (sic)
[von Neumann 1932] Formalized quantum mechanics in “Mathematische Grundlagen der Quantenmechanik”

[von Neumann to Birkhoff 1935] “I would like to make a confession which may seem immoral: I do not believe absolutely in Hilbert space no more.” (sic)

**genesis**

[von Neummann 1932] Formalized quantum mechanics in “Mathematische Grundlagen der Quantenmechanik”

[von Neumann to Birkhoff 1935] “I would like to make a confession which may seem immoral: I do not believe absolutely in Hilbert space no more.” (sic)


[1936 – 2000] many followed them, ...
[von Neumann 1932] Formalized quantum mechanics in “Mathematische Grundlagen der Quantenmechanik”

[von Neumann to Birkhoff 1935] “I would like to make a confession which may seem immoral: I do not believe absolutely in Hilbert space no more.” (sic)


[1936 – 2000] many followed them, ... and FAILED.
the mathematics of it
— the mathematics of it —

**Hilber space stuff**: continuum, field structure of complex numbers, vector space over it, inner-product, etc.
— the mathematics of it —

**Hilber space stuff**: continuum, field structure of complex numbers, vector space over it, inner-product, etc.

**WHY?**
Hilber space stuff: continuum, field structure of complex numbers, vector space over it, inner-product, etc.

WHY?

von Neumann: only used it since it was ‘available’.
the physics of it
von Neumann crafted Birkhoff-von Neumann Quantum ‘Logic’ to capture the concept of superposition.
von Neumann crafted Birkhoff-von Neumann Quantum ‘Logic’ to capture the concept of superposition.

Schrödinger (1935): the stuff which is the true soul of quantum theory is how quantum systems compose.
von Neumann crafted Birkhoff-von Neumann Quantum ‘Logic’ to capture the concept of superposition.

Schrödinger (1935): the stuff which is the true soul of quantum theory is how quantum systems compose.

Quantum Computer Scientists: Schrödinger is right!
the game plan
Task 0. Solve:
\[
\frac{\text{tensor product structure}}{\text{the other stuff}} = ???
\]
Task 0. Solve:

\[
\text{tensor product structure} = \text{the other stuff} = ???
\]

i.e. *axiomatize “⊗” without reference to spaces.*
Task 0. Solve:

\[
\frac{\text{tensor product structure}}{\text{the other stuff}} = ???
\]

i.e. **axiomatize “⊗” without reference to spaces.**

Task 1. Investigate which assumptions (i.e. which structure) on \(⊗\) is needed to deduce **physical phenomena.**
Task 0. Solve:

\[
\frac{\text{tensor product structure}}{\text{the other stuff}} = ???
\]

i.e. **axiomatize “⊗” without reference to spaces.**

**Task 1.** Investigate which assumptions (i.e. which structure) on ⊗ is needed to deduce **physical phenomena**.

**Task 2.** Investigate whether such an “interaction structure” appear elsewhere in **“our classical reality”**.
Outcome 1a:
Outcome 1a: “Sheer ratio of results to assumptions”

Outcome 1a: “Sheer ratio of results to assumptions” confirms that we are probing something very essential.

Outcome 1a: “Sheer ratio of results to assumptions” confirms that we are probing something very essential.

Outcome 1b: Exposing this structure has already helped to solve open problems elsewhere. (e.g. 2× ICALP’10)

Outcome 1a: “Sheer ratio of results to assumptions” confirms that we are probing something very essential.

Outcome 1b: Exposing this structure has already helped to solve open problems elsewhere. (e.g. $2 \times$ ICALP’10)

Outcome 1c: Framework is a simple intuitive (but rigorous) diagrammatic language,

Outcome 1a: “Sheer ratio of results to assumptions” confirms that we are probing something very essential.

Outcome 1b: Exposing this structure has already helped to solve open problems elsewhere. (e.g. 2× ICALP’10)

Outcome 1c: Framework is a simple intuitive (but rigorous) diagrammatic language, meanwhile adopted by others e.g. Lucien Hardy in arXiv:1005.5164:

“... we join the *quantum picturalism* revolution [1]”

Outcome 2a:

Behaviors of matter *(Abramsky-C; LiCS’04, quant-ph/0402130)*:

Meaning in language *(Clark-C-Sadrzadeh; Linguistic Analysis, arXiv:1003.4394)*:

Knowledge updating *(C-Spekkens; Synthese, arXiv:1102.2368)*:
quantitative metric

\[ f : A \rightarrow B \]
quantitative metric

\[ f^\dagger : B \rightarrow A \]
asserting (pure) entanglement

\[
\frac{\text{quantum}}{\text{classical}} = \begin{array}{c} \text{red triangles} \\ \text{black bars} \end{array} \neq \begin{array}{c} \text{red triangles} \\ \text{black bars} \end{array} = \begin{array}{c} \text{red triangles} \\ \text{black bars} \end{array}
\]
asserting (pure) entanglement

\[
\frac{\text{quantum}}{\text{classical}} = \begin{array}{c}
\begin{array}{ccc}
\text{\textcircled{1}} & \text{\textcircled{2}} & \text{\textcircled{3}} \\
\end{array}
\end{array} = \begin{array}{c}
\begin{array}{cccc}
\text{\textcircled{1}} & \text{\textcircled{2}} & \text{\textcircled{3}} \\
\end{array}
\end{array}
\]

\Rightarrow \text{introduce ‘parallel wire’ between systems:}

\[\bigcup\]

subject to: only topology matters!
E.g.
Transpose:

Conjugate:
classical data flow?

\[ f = f \]

\[ = \]

\[ \text{Diagram showing flow of data} \]
classical data flow?

ALICE

BOB

⇒ quantum teleportation
Applying "decorated" normalization \( \Rightarrow \) Entanglement swapping
$\Rightarrow$ gate teleportation MBQC
Thm. [Kelly-Laplaza ’80; Selinger ’05] An equational statement between expressions in dagger compact categorical language holds if and only if it is derivable in the graphical notation via homotopy.

Thm. [Hasegawa-Hofmann-Plotkin; Selinger ’08] An equational statement between expressions in dagger compact categorical language holds if and only if it is derivable in the dagger compact category of finite dimensional Hilbert spaces, linear maps, tensor product and adjoints.
In words: Any equation involving:

- states, operations, effects
- unitarity, adjoints (e.g. self-adjoint), projections
- Bell-states/effects, transpose, conjugation
- inner-product, trace, Hilbert-Schmidt norm
- positivity, completely positive maps, ...

holds in quantum theory if and only if it can be derived in the graphical language via homotopy.
What are these diagrams in ordinary QM?

- Step 1: only maps (no vectors, numbers)
- Step 2: cast Dirac in 2D
- Step 3: summations $\sum_i |ii\rangle \mapsto$ topological entity
  - E.g. $\sum_i |ii\rangle \mapsto \bigcup$

... glass board tutorial
A SLIGHTLY DIFFERENT LANGUAGE FOR NATURAL LANGUAGE MEANING


— the from-words-to-a-sentence process —
Consider meanings of **words**, e.g. as vectors (cf. Google):

--- the from-words-to-a-sentence process ---
What is the meaning the sentence made up of these?

--- the from-words-to-a-sentence process ---
I.e. how do we/machines produce meanings of *sentences*?
I.e. how do we/machines produce meanings of sentences?

Information flow within a verb:

- **object**
- **subject**
Information flow within a verb:

\[ \text{subject} \rightarrow \text{verb} \rightarrow \text{object} \]

Again we have:

\[ \neq \]
Alice ⊗ does ⊗ not ⊗ like ⊗ Bob

meaning vectors of words
Alice $\otimes$ does $\otimes$ not $\otimes$ like $\otimes$ Bob

meaning vectors of words

grammar
Alice $\otimes$ does $\otimes$ not $\otimes$ like $\otimes$ Bob

meaning vectors of words

grammar
Alice \otimes \text{does} \otimes \text{not} \otimes \text{like} \otimes Bob \quad \text{grammar} \\

\text{meaning vectors of words} \\

= \\
\text{Alice} \quad \text{not} \quad \text{Bob} \\
\text{like} \\

— analogy: quantizing grammar! —

Topological quantum field theory:

\[ F : n\text{Cob} \rightarrow \text{FVect}_\mathbb{C} :: \quad \mapsto \quad V \]

Grammatical quantum field theory:

\[ F : P\text{pregroup} \rightarrow \text{FVect}_{\mathbb{R}^+} :: \quad \mapsto \quad V \]

Louis Crane was the first one to notice this analogy.
UNIVERSAL QUANTUM REASONING
WIRES?
not expressive enough
not expressive enough

what is expressive enough?
SPIDERS!
such that, for $k > 0$: 

\[
\begin{array}{c}
m + m' - k \\
n + n' - k
\end{array}
\]
such that, for $k > 0$: 

\[
\begin{array}{c}
m + m' - k \\
n + n' - k \\
\end{array} = \begin{array}{c}
m \\
n \\
\end{array}
\]
\textbf{slopes} \\

\textit{\'(co-)mult.'} = \begin{cases} m \\ n \end{cases}

such that, for $k > 0$:

\[ m + m' - k \quad \begin{cases} \text{...} \\
\text{...} \\
\text{...} \end{cases} \]

\[ n + n' - k \quad \begin{cases} \text{...} \\
\text{...} \\
\text{...} \end{cases} \]

[Diagram of spiders]
such that, for $k > 0$: 

\[
\begin{align*}
\text{such that, for } k > 0: & \quad m + m' - k \\
\text{such that, for } k > 0: & \quad n + n' - k
\end{align*}
\]
Theorem. The following are the same:
Theorem. The following are the same:

• Linear maps \( \{ \mathcal{H}^{\otimes n} \to \mathcal{H}^{\otimes m} \}_{n,m} \) as spiders.
Theorem. The following are the same:

- Linear maps \( \{ \mathcal{H}^\otimes n \to \mathcal{H}^\otimes m \}_{n,m} \) as spiders.

- Orthonormal bases for \( \mathcal{H} \).
Theorem. The following are the same:

- Linear maps \( \{ \mathcal{H} \otimes n \rightarrow \mathcal{H} \otimes m \} \) as spiders.
- Orthonormal bases for \( \mathcal{H} \).
- Copying/erasing pairs

\[ \Delta : \mathcal{H} \rightarrow \mathcal{H} \otimes \mathcal{H} \quad \epsilon : \mathcal{H} \rightarrow \mathbb{C} \]

that are commutative, dagger Frobenius & special.
To proof:
1a) How does an ONB define a copying/deleting pair?
1b) ... and conversely? (CPV’08 thm)
2a) How does a copying/deleting pair define spiders?
2b) ... and conversely? (TQFT thm)

... glass board tutorial

... decorated spiders
... allergic spiders
UNBIASEDNESS:

“allergic spiders”
Thm. Unbiasedness ⇔
Thm. Unbiasedness $\iff$

$Z$-spin:

\[ \delta_Z : |i\rangle \leftrightarrow |ii\rangle \]

$X$-spin:

\[ \delta_X : |\pm\rangle \leftrightarrow |\pm\pm\rangle \]
MEASUREMENT & CLASSICAL CONTROL
QUANTUM CIRCUITS
\[
\delta^\dagger_Z \otimes 1 \circ (1 \otimes \delta_X) = \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0
\end{pmatrix} = CNOT
\]
\[
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0
\end{pmatrix}
\circ
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0
\end{pmatrix}
= ?
\]
STRONG UNBIASEDNESS
\[
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 \\
\end{pmatrix}
\circ \sigma \circ 
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 \\
\end{pmatrix}
\circ \sigma \circ 
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 \\
\end{pmatrix}
= ?
\]
Def. Strong unbiasedness $\Leftrightarrow$

+ comults copy units of other colour.

Strong complementarity $\Rightarrow$ complementarity
PHASES:

“decorated spiders”
Thm. ‘Unbiased states’ always form an Abelian group for spider multiplication with conjugate as inverse.

\[ \{ \begin{array}{c} m \\ n \end{array} \mid n, m \in \mathbb{N}_0, \alpha \in G \} \]
For qubits in $\mathbf{FHilb}$ with $\text{green} \equiv \{|0\rangle, |1\rangle\} \equiv Z$:

\[
\alpha = \begin{pmatrix} 1 \\ e^{i\alpha} \end{pmatrix}_Z \quad \alpha = Z\alpha = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}_Z
\]

These are relative phases for $Z$, hence in $X-Y$:

[Diagram showing a 3D representation of qubits with a relative phase $\alpha$]
For qubits in $\text{FHilb}$ with $\text{red} \equiv \{|+, |-\} \equiv X$:

\[
\begin{pmatrix}
1 \\
e^{i\alpha}
\end{pmatrix}_X \quad \text{and} \quad \begin{pmatrix}
1 & 0 \\
0 & e^{i\alpha}
\end{pmatrix}_X
\]

These are relative phases for $X$, hence in $Z-Y$:
— colour changer —

\[ \begin{array}{c}
\text{HH} \\
\text{HH} \\
\text{HH} \\
\text{HH} \\
\text{HH} \\
\text{HH} \\
\end{array} \quad \text{=} \quad \begin{array}{c}
\text{HH} \\
\text{HH} \\
\text{HH} \\
\text{HH} \\
\text{HH} \\
\text{HH} \\
\end{array} \]
--- one CZ gate ---

\[
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & -1
\end{pmatrix}
\]
--- MBQC is universal for qubit gates ---
— MBQC is universal for qubit gates —
— MBQC is universal for qubit gates —
— MBQC is universal for qubit gates —

⇒ Arbitrarily one-qubit unitary
Thm. (BC & Ross Duncan) The above described graphical language, is universal for computing with qubits.

Proof. CNOT-gate + arbitrary one-qubit unitaries via their Euler angle decomposition yield all unitaries:

$$\Lambda^Z(\gamma) \circ \Lambda^X(\beta) \circ \Lambda^Z(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} \gamma \\ \beta \\ \alpha \end{pmatrix}.$$
Thm. (Miriam Backens, about a year ago) The above described graphical calculus with phases restricted to $\frac{\pi}{2}$-multiples, is complete for qubit stabiliser fragment of quantum computing, provided that we add the Euler angle decomposition of the Hadamard gate.
Example 18. The ubiquitous CNOT operation can be computed by the pattern
\[ P = X_3^4 Z_2^4 Z_2^1 M_0^3 M_0^2 E_{13} E_{23} E_{34} N_3 N_4 \ldots \]
This yields the diagram,
\[ D_P = H H H \pi_3, \{3\} \pi_2, \{2\} \pi_2, \{2\} \pi_3, \{3\} \pi_2, \{2\} \pi_2, \{2\} \pi_2, \{2\} π \]
where each qubit is represented by a vertical "path" from top to bottom, with qubit 1 the leftmost, and qubit 4 the rightmost.

By virtue of the soundness of \( R \) and Proposition 10, if \( D_P \) can be rewritten to a circuit-like diagram without any conditional operations, then the rewrite sequence constitutes a proof that the pattern computes the same operation as the derived circuit.

Example 19. Returning to the CNOT pattern of Example 18, there is a rewrite sequence, the key steps of which are shown below, which reduces the \( D_P \) to the unconditional circuit-like pattern for CNOT introduced in Example 7. This proves two things: firstly that \( P \) indeed computes the CNOT unitary, and that the pattern \( P \) is deterministic.


Toy qubits vs. true quantum theory in one language:

\[
\frac{\text{Spekkens’ qubit QM}}{\text{stabilizer qubit QM}} = \frac{Z_2 \times Z_2}{Z_4} = \frac{\text{local}}{\text{non-local}}
\]

Toy qubits vs. true quantum theory in one language:

\[
\frac{\text{Spekkens’ qubit QM}}{\text{stabilizer qubit QM}} = \frac{\mathbb{Z}_2 \times \mathbb{Z}_2}{\mathbb{Z}_4} = \frac{\text{local}}{\text{non-local}}
\]


Generalized Mermin arg. ⇔ strong complementarity

— multipartite entanglement structure —

Tripartite SLOCC-classes as comm. Frobenius algs:

\[
GHZ = |000\rangle + |111\rangle \\
W = |001\rangle + |010\rangle + |100\rangle
\]

= ‘special’ CFAs

\[
= \begin{array}{c}
\text{‘anti-special’ CFAs}
\end{array}
\]

\[
\begin{array}{c}
\end{array}
\]
--- **GHZ-spiders** ---

**Data:**

$$\left\{ \begin{array}{c} m \\ n \end{array} \right\} \quad | \quad n, m \in \mathbb{N}$$

**Rules:**

$$m + m' - k = m + m' - k$$  
$$n + n' - k = n + n' - k$$
--- W-spiders ---

**Data:**

\[
\begin{cases}
m \\
n
\end{cases},
\quad n, m \in \mathbb{N}
\]

**Rules:**

\[
m + m' - 1 = n + n' - 1
\]
--- **W-spiders** ---

**Data:**

\[
\begin{cases}
  m \\
  n
\end{cases}, \quad \bullet, \quad \bullet \quad | \quad n, m \in \mathbb{N}
\]

**Rules:**

\[
\begin{align*}
  m + m' - 2 \\
  n + n' - 2
\end{align*}
\]

\[
= \\
\begin{align*}
  m + m' - 2 \\
  n + n' - 2
\end{align*}
\]